Demonstrating Fourier-Based Filtering Using the TMS320C5x EVM

Authors: M.A. Diaz-Vicente, A. Martin-Fraile, J. Lopez-Martinez, Dr. J.M. Giron-Sierra

EFRIE, France

December 1995
SPRA306

TEXAS INSTRUMENTS

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain application using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

TRADEMARKS

CONTACT INFORMATION

US TMS320 HOTLINE	(281) 274-2320
US TMS320 FAX	(281) 274-2324
US TMS320 BBS	(281) 274-2323
US TMS320 email	dsph@ti.com

Contents

Demonstrating Fourier-Based Filtering Using the TMS320C5x EVM

Abstract

This application report describes a experiment using the Texas Instruments (TI™) TMS320C5x Evaluation Module (EVM) and Fourier-based filtering to achieve real time response. The Discrete Fourier Transform (DFT) is employed to obtain a discreet spectrum of a source signal received by the EVM. The coefficients of the spectrum can then be modified using an MS-Windows-based user interface that mimics a mixing table for sound processing. DFT is again used to synthesize an output (filtered) signal.

This application report discusses the TI TMS320C5x EVM as a platform for academic research, derives the theory of the Fourier-based filter, and describes the user interface created for the implementation of the filter.

This document was an entry in the 1995 DSP Solutions Challenge, an annual contest organized by TI to encourage students from around the world to find innovative ways to use DSPs. For more information on the TI DSP Solutions Challenge, see TI's World Wide Web site at www.ti.com.

Product Support on the World Wide Web

Our World Wide Web site at www.ti.com contains the most up to date product information, revisions, and additions. Users can register with TI&ME to build custom information pages and receive new product updates automatically via email.

The Project

Main Idea

The Fourier Transform may be used to obtain signal decomposition, in terms of sinusoidal harmonics. We can draw the spectrum of the signal, plotting the values of the amplitude of each harmonic vs. its frequency. We can recover the original signal by adding these harmonics.

If we modify the amplitude of the harmonics, changing the spectrum profile (for instance, emphasizing high frequencies), the recovered signal is a filtered version of the original.

We want to devise an easy and friendly system to experiment with this filtering idea.

The filtering process must be achieved in real time. And the human interface should be inviting for the user.

Objective

The objective is to create a Fourier-based filter for real-time use.

The TMS320C5x DSP makes it possible to apply computational-intensive signal processing methods (as in our case) to derive instantaneous results. The TMS320C5x EVM provides an appropriate platform to reach our objective, with the perspective of its application in academic laboratory conditions.

The chief development effort is to design and code the algorithm for the TM5320C50, to accomplish the DFT of the input signal, interactively change the resulting spectrum, and synthesize the so-filtered output signal.

Theory for the Fourier-Based Filter

We selected the DFT as the simplest way to introduce ourselves to the programming and use of DSPs for Fourier-based applications.

Here is a formulation of the Discrete Fourier Transform:

N = number of signal samples (33 in our program)

L = number of harmonics = $\frac{(N-1)}{2}$ (16 in our program)

The input signal can be approximated by the following combination of harmonics:

$$u(t) = \frac{1}{2}\alpha_0 + \sum_{k=1}^{L} \alpha_k \cos\left(\frac{2\pi}{N}kt\right) + \beta_k \sin\left(\frac{2\pi}{N}kt\right)$$

In this expression, the harmonics cos() and sin() are present with an amplitude denoted by the coefficients α_k and β_k.

To calculate the value of these coefficients:

$$\alpha_k = \frac{2}{N}\sum_{n=0}^{N-1} y(nT)\cos\left(\frac{2\pi}{N}knT\right) \qquad k = 0,1,\ldots,L$$

$$\beta_k = \frac{2}{N}\sum_{n=0}^{N-1} y(nT)\sin\left(\frac{2\pi}{N}knT\right) \qquad k = 0,1,\ldots,L$$

In our application, we modify these values (each coefficient is multiplied by a filtering factor), obtaining new coefficients:

$$\alpha'_k = f_k\,\alpha_k$$

$$\beta'_k = f_k\,\beta_k$$

We then synthesize the following filtered output signal:

$$y(t) = \frac{1}{2}\alpha'_0 + \sum_{k=1}^{L} \alpha'_k \cos\left(\frac{2\pi}{N}kt\right) + \beta'_k \sin\left(\frac{2\pi}{N}kt\right)$$

The harmonics are ordered following a discrete set of increasing frequencies. Lowering or rising the amplitude of the ith harmonic has the effect of making more intense or moderate the ith frequency contents of the filtered signal.

Implementation Details of the Algorithm

During real time filtering we follow a cyclic procedure governed by the sampling period. We have to maintain a record of the last N input signal samples.

After a sample is taken, we compute the coefficients, α_k and β_k, for the $u(t)$ approximation. As the number of samples is fixed (N = 33), we can simplify calculations by managing a table of pre-calculated sinusoidal terms cos( ) and sin () for $t = nT$ (T, the sampling period).

Once the set of coefficients is determined, we apply the filtering factors, as established with the simulated knobs on-screen, and compute the synthesis of the filtered signal. The result appears at the output on the next sampling instant.

User Interface

We developed the user interface with Delphi for MS-Windows.

The screen displays the mimic of a sound preamplifier table with a set of knobs: using the mouse you can change the harmonics amplitudes. There are two screens: one for the sin() terms, and one for the cos() terms.

Once a desired filter profile is established, you use the *Transferir Coeficientes al DSP* button to actually modify the set of filtering factors applied by the EVM.

Execution

When the user pushes the *Transferir Coeficientes al DSP* button, an automated sequence of our environment operations starts as follows:

1) Two files are generated.
   - `cossen.asm`

     Table of pre-calculated sinusoidal terms (It is generated only if not present in the actual directory.)
   - `coeffs.asm`

     Table of new filtering factors
2) Then, the DSP assembler (`dsk5a.exe`) is launched to process our program (`sintesis.asm`), which must be present in the actual directory, obtaining `sintesis.dsk`.

3) Afterward, we launch the loader (`dsk5l.exe`) to download `sintesis.dsk` into the EVM to be executed there.

We included a simple menu bar with two options. Using this menu, you can change the communication port to be employed for the PC-EVM interaction.

System Requirements

We adhere to the MS- Windows standards. Any computer able to run the MS-Windows version 3.0 operating system can run our environment using either the COM 1 or 2 port to interact with the EVM.

Installation Procedure

1) Ensure that the EVM is running.
2) Create a working directory (hard disk or diskette).
3) Copy all files to this directory.
4) Go to this directory.
5) Type `Win sintesis` and press Enter.

(We included the file `DSP.ICO`: an icon to be employed by the user to launch our program.)

Utilization Experience

The topic of the Fourier Transform appears in several academic contexts: courses in applied mathematics or physics and engineering curricula (when dealing with signals). It is of pedagogical interest to show how it works in a lively fashion.

The experience we acquired in our laboratory of Digital Control is quite encouraging. The students enjoyed working with the user interface and the immediate results obtained with a microphone and a loudspeaker. Some asked to take the EVM home to try it with their own music chains. To learn more, we employed signal generators and oscilloscopes to see the effects of the approximation we are handling.

Diskette Contents

The diskette has two subdirectories. The TEXT subdirectory includes this project description in Microsoft Word format along with a program listing (`sintesis.lst`).

The other subdirectory, called PROGRAM, includes all files for our application: you can copy these files to hard disk for quick execution (the application can be directly executed from the diskette, but this means a delay while running).

Summary

We conceived a motivating experiment about the utilization of Fourier for filtering. The DSP makes it possible to achieve this experiment in real time. We developed an environment to use the TMS32OC5x EVM for such a purpose.

Within our environment, the user can easily change the frequency contents of a signal (connected to the EVM input), and immediately observe the results (the corresponding modified signal appears at the EVM output).

By means of the Discrete Fourier Transform (DFT), we obtained the spectrum of the input signal. With a special human interface, we could change the profile of this spectrum (for instance, rising or lowering any of the harmonics). Again using the DFT, we synthesized the resulting filtered signal.

We changed the frequency contents by moving the knobs of a simulated sound preamplifier table. Using our MS Windows-based human interface , you use the mouse to move the knobs and so modify the filter characteristics.

Our environment included the means to facilitate the use of the EVM for the experiment: automatic code generation, downloading, and execution.

We tested the system using a microphone and a loudspeaker with interesting pedagogical results. It is also good practice to employ a signal generator and an oscilloscope to study the details of the filtering process.

Both the experiment and the system to accomplish it are intended to be employed in the context of signal processing courses (engineering or applied mathematics and/or physics curricula).

References

F. Scheid. *Numerical Analysis.* Schaum. McGraw-Hill (Chapter 24).

The Authors

Team Members

Maria A. Diaz-Vicente. Born in Madrid, Spain, 1973. Student of Computer Engineering at the University Complutense of Madrid, Spain. Experience in distributed/concurrent programming, and machine-code development.

Alberto Martin-Fraile. Born in Madrid, Spain, 1973. Student of Computer Engineering at the University Complutense of Madrid, Spain. Experience in distributed/concurrent programming, and machine-code development.

Javier Lopez-Martinez. Born in Madrid, Spain, 1973. Student of Computer Engineering at the University Complutense of Madrid, Spain.Experience in distributed/concurrent programming, and machine-code development.

Advising Professor

Dr. Jose M. Giron-Sierra. Born in Valladolid, Spain, 1949. Ph. D. Physics, University Complutense of Madrid, 1978. Full professor of Digital Control with the Department of Computer Science and Automatic Control, University Complutense of Madrid, Spain. During the last nine years, he conducted seven Research Projects on Artificial Intelligence for Simulation and Control, applied to industrial processes and robotics. More than forty publications. Present research interests: neural nets, fuzzy, and genetic algorithms for experimental control.