Implementation of Hilbert-Huang Transform (HHT) Based on DSP

Li Hongsheng Li Zhengqin Chen Wenwu Zhao zhao
Wuhan University of Technology, Wuhan, China 430070

Abstract: Hilbert-Huang Transform (HHT) is a new method for non-stationary and nonlinear data analysis. It has been applied with great success soon after it was first introduced in 1998. Most researches at present are mainly focused on the algorithm itself and its software implementation. However higher processing speed is necessary for realtime signal processing or quasi-realtime signal processing. In our efforts, a hardware-accelerated approach of HHT method based on PC, DSP and PCI technologies is developed. With this approach, the processing time can be reduced to less than one tenth compared with "pure software" ones. In this paper, the basic principle of HHT is briefly reviewed, the hardware and software design considerations are discussed, and some preliminary experimental results are given.

Keywords: Hilbert-Huang Transform, DSP, EMD

1. Introduction

Hilbert-Huang Transform (HHT) refers to the data analysis algorithms proposed by Huang et al in 1998 [1], which comprises Empirical Mode Decomposition (EMD) and its related Hilbert spectrum. On the study of instantaneous frequency, Huang et al proposed a method called Empirical Mode Decomposition (EMD), with which any complicated data set can be decomposed into a finite and often small number of Intrinsic Mode Functions (IMF) that admit well-behaved Hilbert transform [1,2]. Thus the energy-frequency-time distribution of the signal is constructed. The major advantage of EMD is that the basis functions are derived from the signal itself. Hence the analysis is adaptive, in contrast to the traditional methods which basis functions are fixed. Though the theory of HHT has not yet been strictly proved mathematically, researches show that it's an effective method for non-stationary and nonlinear signal analysis, having great potential in application.

Most researches at present focus on the improvement of HHT algorithm and its software implementation. However experiments found that the processing time of HHT, depending on the complexity of the data, some criterions of the algorithm and the implementation method, may be less satisfactory than expected in some situations requiring realtime processing and quasi-realtime processing. NASA Goddard Space Flight Center has developed a system called HHT data processing system (HHT-DPS) [3], however it is still at the phase of "software only" version, with "hardware accelerated" version being their future work, hoping to achieve 10~50X speed-up.

In fact, we have been on the work since 2002. In our efforts, a hardware accelerated system for implementing HHT method has been developed. The system is composed of a PC and a PC plug-in card. The card, using DSP chip as the processor, incorporating with high speed A/D unit, specially designed control logic unit and PCI interface circuit, has high-speed data acquiring ability and can process the data in EMD method. While the upper PC provides a user-friendly interface, it controls the DSP system's operation, receives the processed data, computing Hilbert spectrum and display it on monitor. Experiments and preliminary tests show that the system works well for HHT implementation and can achieve about 10 times speed-up compared with the "software only" approach, provided that the data complexity and algorithm options are the same.

2. HHT BASICS[1,2,4]

Given a real signal x(t), we can build the corresponding analytic signal (or complex trace): X(t) = x(t) + jH[x(t)], where H[x(t)] is the quadrature signal corresponding to x(t) obtained using Hilbert transform: $H[x(t)] = \frac{1}{\pi} P \int_{-\infty}^{+\infty} \frac{x(\tau)}{t-\tau} d\tau$ with $P \int_{-\infty}^{+\infty}$

being the Cauchy principal value of the integral. The complex trace can be expressed as $X(t)=A(t)e^{j\theta(t)}$, where A(t) is the instantaneous amplitude (or reflection strength) and $\theta(t)$ is the instantaneous phase. We can then define the instantaneous frequency $\omega(t)$ as $d\theta(t)/dt$.

Admitting the instantaneous frequency could avoid using traditional frequency defined in global sense. Unfortunately, not all the signal will have meaningful $\omega(t)$ defined above, many non-physical negative frequency values would occur. So, restrictions should be added. Having carefully studied the instantaneous frequency, Huang et al found that there are a set of functions would get meaningful instantaneous frequency, the restrictive conditions are: (1) in the whole data set, the number of extrema and the number of zero crossings must either equal or differ at most by one; (2) at any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero. These functions are called 'intrinsic mode function' (IMF). EMD method has the ability of decomposing the data into a number of IMFs, this made instantaneous frequency of the signal meaningful and Hilbert spectrum available. In this sense, EMD method is the key part of HHT. Given a time series data x(t), the effective algorithm of EMD could be summarized as follows [4]:

(1) Initialize $r_0(t)=x(t)$, i=1

(2) Extract the i-th IMF

(a) Initialize $h_0(t)=r_i(t)$, j =1

(b) Identify all the extrema of $h_{j-1}(t)$

(c) Interpolate the local maxima and the local minima by a cubic spline to form upper and lower envelopes $e_{max}(t)$ and $e_{min}(t)$ of $h_{j-1}(t)$

(d) Compute $m_{j-1}(t)=(e_{max}(t)+e_{min}(t))/2$

(e) if $h_j(t)=h_{j-1}(t)-m_{j-1}(t)$ is a IMF or it's a IMF with a criterion, then a IMF is extracted, designated as $c_i(t)=h_j(t)$, go to 3; else go to (b) with j=j+1

(3) $r_i(t)=r_{i-1}(t)-c_i(t)$

(4) if stopping criterion is satisfied, that is, if $r_i(t)$ is small enough or $r_i(t)$ couldn't be decomposed anymore, the process stop, with $r_i(t)$ is the residual; else goto (2) with i=i+1 .

After EMD, we have: $x(t)=\sum_{i=1}^{n} c_i(t)+r_n(t)$.

Leave residual $r_n(t)$ for purpose, and apply Hilbert transform to each IMF $c_i(t)$, then we can calculate the instantaneous frequency $\omega_i(t)$ and instantaneous amplitude $A_i(t)$ of each IMF respectively. Finally, we can represent in the time-frequency plane the triplet $\{t, \omega_i(t), A_i(t)\}$ in color map or contour , which reveals the intrinsic feature of the original signal.

3 DSP system structure

The DSP system includes signal conditioning circuit, A/D, DSP, PCI interface circuit, control unit, power unit and level shifting circuit. Consider the price-performance ratio, TMS320VC5402, a fixed-point DSP from TI company, is used in our design, which features enhanced Harvard architecture, minimized power consumption and a high degree of parallelism.

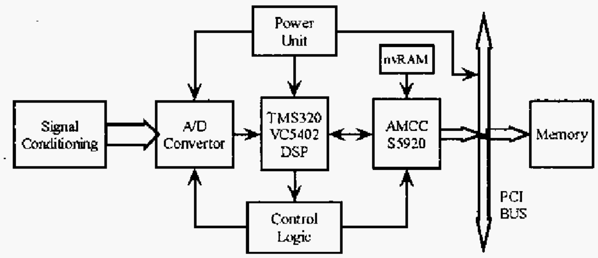

Fig.1 DSP System Architecture

3.1 Hardware design

(1) PCI interface design. PCI bus is a widely used popular local bus. General PCI interface IC S5920 is used in our design. S5920, produced by AMCC Company, is compliant with PCI 2.2 specification, featuring full 132Mbyte/sec transfer rate, bus operation to 33MHz, and 4 definable Pass-Thru regions.

(2) The interface between S5920 and DSP. In this system, S5920 works at master mode. Bi-directional data transfer is realized using Pass-Thru Region of S5920 and HPI port of VC5402. Since HPI port works at DMA, no other resources of VC5402 are occupied. Fig.2 shows the interface circuit.

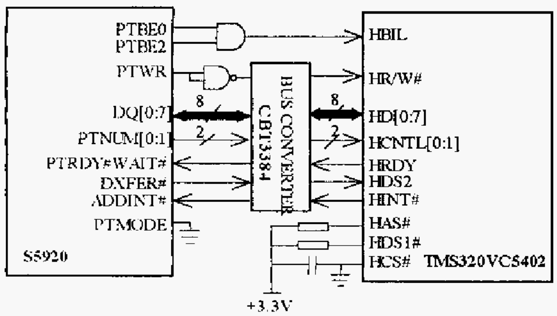

Fig.2 Interface Circuit of VC5402 and S5920

(3) A/D unit. AD7865 is a high speed (2.4us), low power consumption, 14 bit analog digital converter with the ability of 4 channels sampling synchronously. Fig.3 shows the circuit of this unit. XF from DSP timely triggers AD7865 for AD converting, when converting is finished, EOC# is activated, and this generate an interrupt (INT0#) in DSP for the notice of getting converted results.

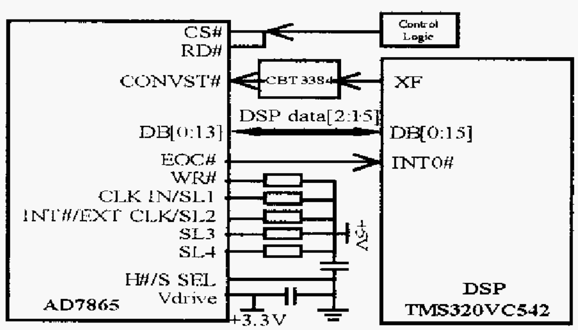

Fig.3 AD unit

3.2 Software design

The system software consists of three functional modules: DSP signal processing module, PCI card driver module and PC display module.

(1) DSP signal processing software. It's the core part of the system software, including configuration of DSP, A/D sampling and implementation of EMD algorithm. Fig. 4 shows the flow chart of this module. The development toolkit of C54X supports programming in assembly language and in C language. Assembly language, as a low-level language, could allow the programmer exploit the capabilities of DSP hardware, and gain high run speed, but has the disadvantage of low portability and readability; while programming in C language, though the size of code generated may be larger and need more run-time, it has the advantage of high portability, readability and maintainability. The third way is a combined way, which is adopted in our development, the control code is written in C and time-critical inner loops are written in assembly language. Fig.4 shows the flow chart of the DSP signal processing software.

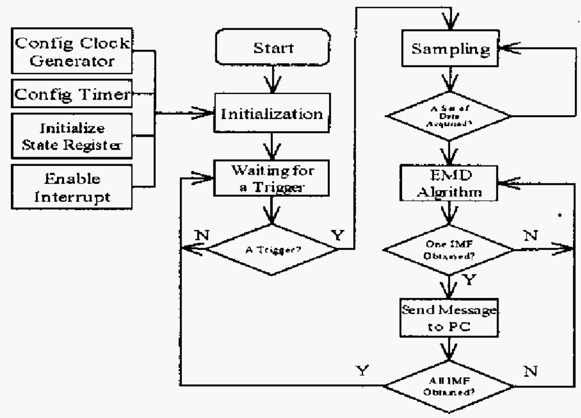

Fig.4 flow chart of DSP software

(2) PCI card driver. This software module, running at PC, provides for PC the ability to access PCI card resources. But since hardware can't be accessed from the application level (user mode or ring3) under Windows9X/NT, the software must be created in kennel mode (ring0). There are several tools for developing/debugging in kernel mode such as DDK, SDK, but utilizing these tools, the programmer have to learn the internals of the operating system he is working on. In this system, PCI card driver is developed utilizing WinDriver , a device driver development tool produced by KRFTech company, which allows the programmer to create device driver in the familiar user mode environment, using MSDEV, Visual C/C++, etc, and no need to have knowledge of operating system internals, kernel programming, thus reduce the development time to a great degree. Windriver offers two programming mode: User Mode and Kernel Plug In Mode. The later offers better kernel performance, it is used when there are critical run-time requirements. In this system, programming in User Mode is used at present stage.

(3) PC display module. It functions offering

user friendly GUI: setting EMD options, getting EMD results, performing Hilbert transform, and displaying the time-frequency-energy distribution on PC monitor. The module is programmed using VC++ and Matlab.

4. Experiments

(1) The first example is mathematically described as $X(t)=\cos(2\pi t/50)+0.6\cos(2\pi t/25)+0.5\sin(2\pi t/200)$, which is a non-stationary signal composed of three wave of different frequency. Data length is assumed to be 300, and sampling time is 300 second, sampling frequency is 1.0 Hz. The EMD results are shown in Fig.5

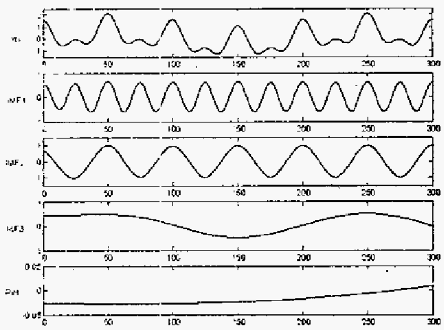

Fig.5 example 1

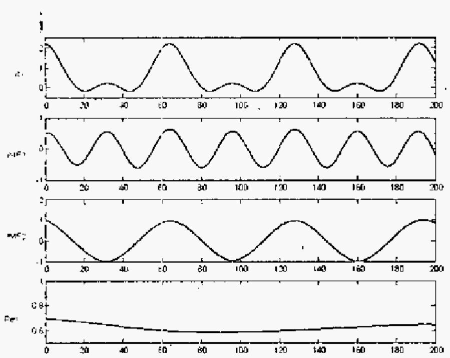

Fig.6 example 2

(2) The second example is Stokes Wave. It's currently a non-linear theoretical model commonly used of a limited amplitude wave. The 2 order Stocks wave could be approximately described as $X(t)=0.5a^2k+a\cos\omega t+0.5a^2k\cos 2\omega\cdot t$, where a is the wave amplitude, k is the wave number. For a=1, ak=1.2, $\omega=2\pi/64$, assuming sampling frequency is 0.5 Hz, sampling time is 200 second, then 400 data is obtained. The EMD results are shown in Fig.6

5. Conclusions

We have developed a system based on PC, DSP and PCI technologies, aiming at realizing high speed data acquiring and HHT processing. Preliminary experiments and tests show that the system works well, but the results reported here are merely of an experimental natre. Future work will include: (1) Improve hardware system, for example, using CPLD or FPGA instead of current simply designed control logic circuit. (2) Keep up with the new improvements of HHT, and update the software. (3) Optimize the software by evaluating software computational complexity, finding bottlenecks and try using DSP/BIOS Real time Kennel and Real-time Data Exchange (RTDX) technology. (4) Set up system performance measures.

References:

[1] Huang N E, Shen Z, Long S R, et al. The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-stationary Time Series Analysis. Proc R Soc Lond A, 1998, 454: 899-995

[2] Chen Wenwu. Research on the Implementation Technologies of EMD Method Based on DSP (in Chinese). Paper for Master's degree of Wuhan Universtity of Technology, 2002.

[3] Semion Kizner, Thomas P. Flatley, Dr. Norden E. Huang, Hilbert-Huang Transform Data Processing System (HHT-DPS) http://techtransfer.gsfc.nasa.gov/HHT/attachments/HHT_Presentation_032403.pdf

[4] G. Rilling, P. Flandrin, P. Goncalves. On Empirical Mode Decomposition and its Algorithms. IEEE-EURASIP Workshop on Nonlinear Signal and Image Processing NSIP-03, Grado (I). June 2003

[5] TMS320C54X DSP Reference Set, Volume 4: Applications Guide. Texas Instruments. 1996

[6] AMCC PCI Products Data Book. 1998