Design of Real-Time Adaptive Lattice Predictor Using a Digital Signal Processor

(DSP를 이용한 실시간 적응격자 예측기 설계)

金 成 煥*, 洪 基 龍*, 洪 玩 憙*

(Sung Hwan Kim, Ki Ryong Hong and Woan Hue Hong)

要 約

본 논문에서는 TMS32020 DSP를 사용하여 적응격자 예측기를 실시간으로 구현하였다. 실시간 적응격자 예측기 시스템은 중앙연산 제어부 및 입출력부로 설계 제작하였다. 본 연구에서 제작한 실시간 적응격자 예측기는 입력신호와 예측기에 의하여 얻어진 출력신호를 비교 검증 함으로써 그 성능을 평가하였으며, 4단 격자구조인 경우 전체 수행시간은 156μs로서 최대 수행주파수 6.41KHz를 얻었다.

Abstract

Real-time adaptive lattice predictor was implemented on the TMS32020 DSP chip for digital signal processing. The implemented system was composed of Input-Output units and central processing-control unit and its supporting assembly software. The performance of hardware realization was verified by comparing input signal and one-step prediction signal which are calculated by the real-time adaptive lattice predictor.

As a result, for 4 stage lattice structure, the maximum running frequency was obtained as 6.41 KHz in this experiment.

I. Introduction

Rapidly growing microelectronics technology has revolutionized modern digital signal processing. One of the most important breakthroughs in electronic technology is the high-speed digital signal processors (DSP). Over last dozen years, a great deal of digital signal processing algorithms have been successfully developed in speech, image, sonar, radar and other areas. More recently there has been a growing interest in the development of adaptive filtering algorithms. Until now, most of the digital signal processing was only the domain of minicomputers. Especially, adaptive lattice algorithm was not realized as a real-time hardware because of its complexity. Lattice forms are widely used in signal processing applications involving linear filtering and prediction after the study based on the two-multiplier lattice of Itakura [1].

In this paper, the adaptive lattice predictor (ALP) which is the basic element for digital signal processing was implemented on the TMS 32020 (Texas Instrument) DSP in real-time.

The implemented ALP is composed of input-output units and central processing-control unit with the DSP chip, and its support-

*正會員, 서울市立大學校 電子工學科
(Dept. of Elec. Eng., Seoul City Univ.)

接受日字 : 1987年 5月 30日

ing assembly software. Completed programs were burned into EPROM which were installed on the circuit board with the DSP chip to complete the system. The performance of hardware realization was verified by comparing input signal and one-step prediction signal which are calculated by the ALP on the DSP. Also, the maximum running frequency was determined.

II. Adaptive Lattice Predictor

The adaptive lattice one-step predictor model is given in Fig.1, where $e_i(n)$ and $w_i(n)$ are referred to the forward and backward prediction errors, respectively; $K_i(n)$ denotes the i-th reflection coefficient (lattice weight). The input to the N-stage lattice is x(n) and each Z^{-1} represents a unit delay.

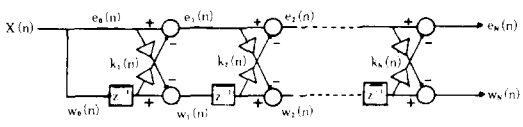

Fig.1. Adaptive lattice one-step predictor.

If x(n) is an estimate for predicting x(n) using the past samples x(n-1), x(n-2), ...x(n-N), then

$$e_N(n) = x(n) - \hat{x}(n) \tag{1}$$

is the forward prediction error. Again given x(n-1), x(n-2), ... x(n-N), if we wish to predict x(n-N-1), then

$$W_N(n) = x(n-N-1) - \hat{x}(n-N-1) \tag{2}$$

is the backward prediction error, where $\hat{x}(n-N-1)$ denotes an estimate of x(n-N-1). Several stratagies for updating the weights are discussed in [3] Here, in this experiments, the gradient adaptive lattice algorithm proposed by Griffiths [2] was implemented in real-time, and the simpler version of this algorithm was used when only a single reflection coefficient is considered. As such , the lattice weights are updated using the relation

$$K_i(n+1) = K_i(n) - \hat{u}\left[\frac{\partial \delta_i^2(n)}{\partial K_i(n)}\right], \quad 1 \leqslant i \leqslant N \tag{3}$$

where $\delta_i^2(n) = e_i^2(n) + w_i^2(n)$ is the total prediction error at lattice stage i and time n, $K_i(n)$ denotes the value of K_i at time n; an $\hat{u}$ is a convergence parameter.

The recursive equations that describe the lattice model with time-varying weight in Fig.1 are as follows:

$$x(n) = e_0(n) = W_0(n) \tag{4}$$

$$e_i(n) = e_{i-1}(n) - K_i(n)\, W_{i-1}(n-1) \tag{5}$$

$$W_i(n) = W_{i-1}(n-1) - K_i(n)\, e_{i-1}(n) \tag{6}$$

for $1 \leqslant i \leqslant N$

Using (3) and (5), (6) it can be shown that

$$K_i(n+1) = K_i(n) + 2\hat{u}\,[\, e_i(n)\, W_{i-1}(n-1) + W_i(n)\, e_{i-1}(n)\,] \tag{7}$$

A normalized convergence parameter has to be used in (7) in place of $2\hat{u}$. Thus equation (7) is modified to obtain

$$K_i(n+1) = K_i(n) + \frac{\alpha}{\sigma_i^2(n)}\,[\, e_i(n)\, W_{i-1}(n-1) + W_i(n)\, e_{i-1}(n)\,] \tag{8}$$

where $0 < \alpha < 1$ and $\sigma_i^2(n)$ is the input power estimate at the i-stage.
It can be computed recursively as

$$\sigma_i^2(n) = \beta\, \sigma_i^2(n-1) + (1-\beta)\,[e_{i-1}^2(n) + W_{i-1}^2(n-1)] \tag{9}$$

where $0 < \beta < 1$ is a smoothing parameter.
A convenient choice for α in (8) is $\alpha = 1-\beta$. Then we have the update equation

$$K_i(n+1) = K_i(n) + \frac{1-\beta}{\sigma_i^2(n)}\,[e_i(n)\, W_{i-1}(n-1) + W_i(n)\, e_{i-1}(n)\,] \tag{10}$$

for $1 \leqslant i \leqslant N$, which is referred to as gradient adaptive lattice equation for a one-step predictor, where $\sigma_i^2(n)$ is updated via (9).

III. System Design

1. Hardware

In a real-time implemention of the ALP, there are two principal features of the processor hardware. The first is a high speed external memory interface circuit. This circuitry provides two separate memory address generators which are operated under programmed control of the DSP chip. The second is an interface between the DSP and external I/O devices.

A functional block diagram of our designed ALP processor architecture is shown in Fig.2.

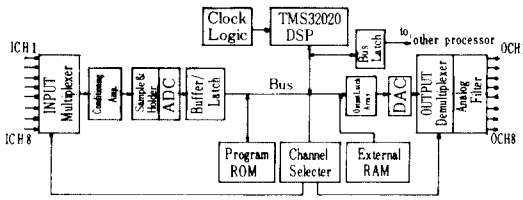

Fig.2. Architecture of the ALP.

The architecture permits access from external memory to the DSP under the gradient adaptive lattice algorithm using the TMS 32020 assembly language instructions. In order to retain maximum flexibility, we have made each 16K of RAM and EPROM accessible to the DSP chip by multiplexing the address bits input to the memory devices.

The input unit circuitry is shown in Fig.3. An input signal selected by channel selector (Q1) is low pass filtered (f_c=21.9 KHz; Q2) for high frequency noise rejection and amplified by noniverting amplifier Q3. The sample and holder Q4 is a SHC 5320 (Burr-Brown) and the analog to digital (A/D) converter Q6 is an ADC 80 AG (Burr-Brown). The A/D converter outputs (complementary offset binary code) are latched by Q10, Q11 and stored on the internal RAM by the I/O signal of DSP.

The central processing-control circuitry is shown in Fig.4. The master clock of TMS 32020 (Q2) is a 20-MHz, so the access time of all components must be within 50 nsec. But the access time of the EPROM and RAM used in this circuit is 200 nsec. Therefore two wait-state generator composed of Q4 and Q8 is used. The DSP chip is communicated with external devices by the control signals ($\overline{PS}$: ROM control signal, $\overline{IS}$: I/O port selection

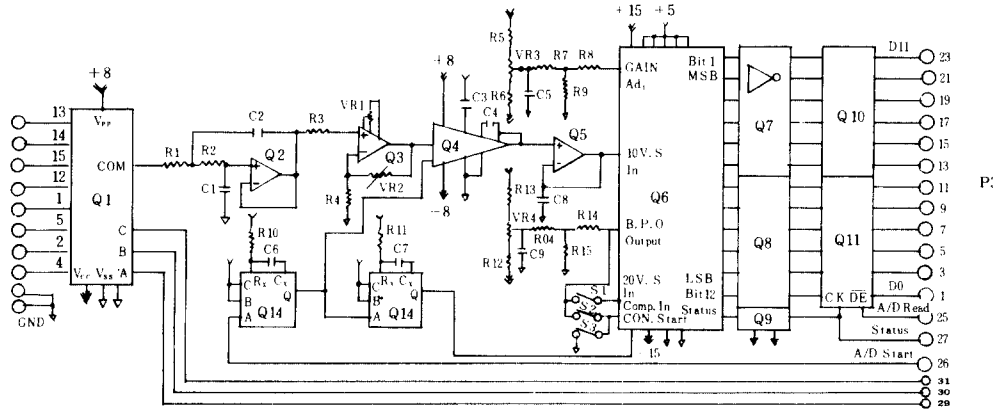

Fig.3. Schematic of input unit.

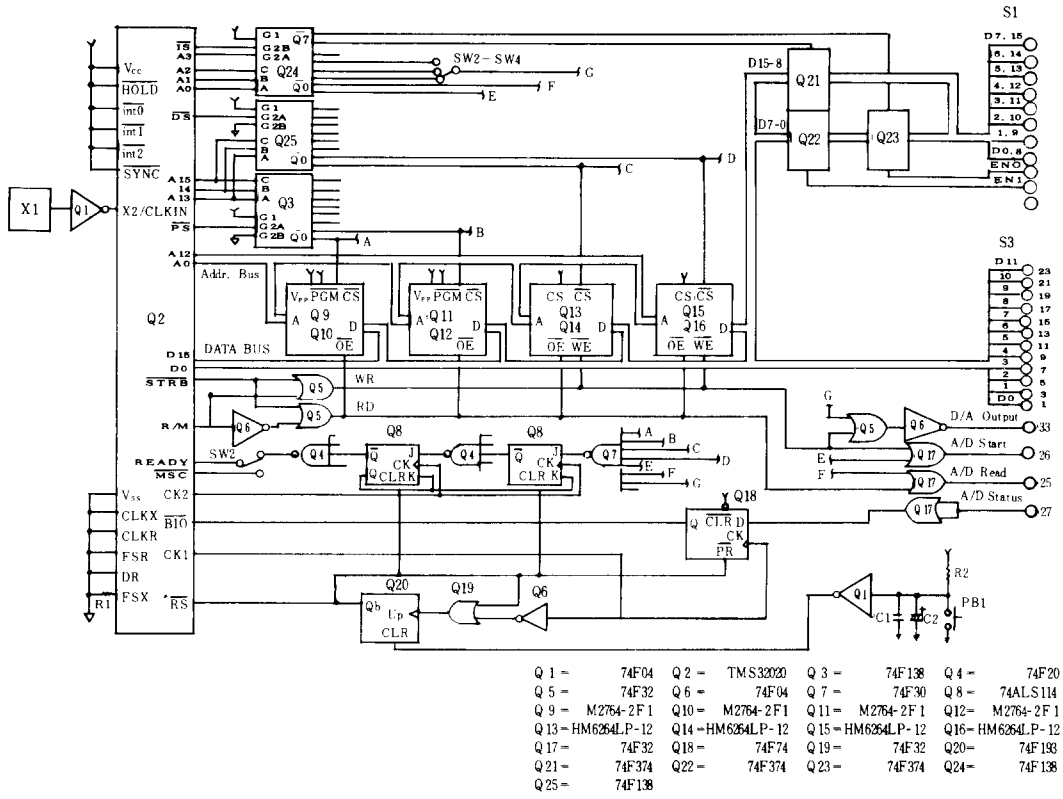

Fig.4. Schematic of central processing-control unit.

signal, $\overline{DS}$: RAM control signal) through Q3, Q24, Q25 respectively. By the selection of Q24, the conversion start ($\overline{CS}$) signal is activated and the status signal from the A/D coverter transmitted to $\overline{BIO}$ interrupter of the DSP. After the conversion is complete, the conversion complete status signal is passed on directly to a $\overline{BIO}$ of the DSP and the conversion data of ADC is transferred into the buffer/latch register. Then the contents of the buffer/latch register are read by the DSP. By the firm were written in EPROM, the adapting reflection coefficients and variance at each lattice stage of the ALP are calculated from these data. Next the calculated prediction value and forward error are sent to the output unit.

The output unit circuitry is shown in Fig.5. The output data from the DSP are latched by Q12, Q13 and passed to the digital to analog (D/A) converter Q15 (DAC-HK12BUC; DATEL-INTERSIL). The output channel is selected by Q16. The output data is conditioned by Q17, Q18, Q19.

2. Software

All of the instruction code was written in the TMS32020 DSP assembly language for maximum execution speed. The TMS32020 simulator was used to allow program verification. The simulator uses the TMS32020 Macro Assembler/Linker. Once program execution is suspended, the internal registers and both program and data memories were inspected and/or modified with this simulator. After the system initialization, the smoothing parameter and the number of lattice stages are set to the internal registers, and the order and time are updated by the ALP computational steps. In the 2's complement fixed-point arithmetic, the roundoff error noise variance depends on quantization step size, and the TMS32020 DSP perform 16*16 bits operation through the hardware multiplier within one cycle. The important property is that lattice structure is better behaved (less sensitive) under quantization of the multiplier coeffi-

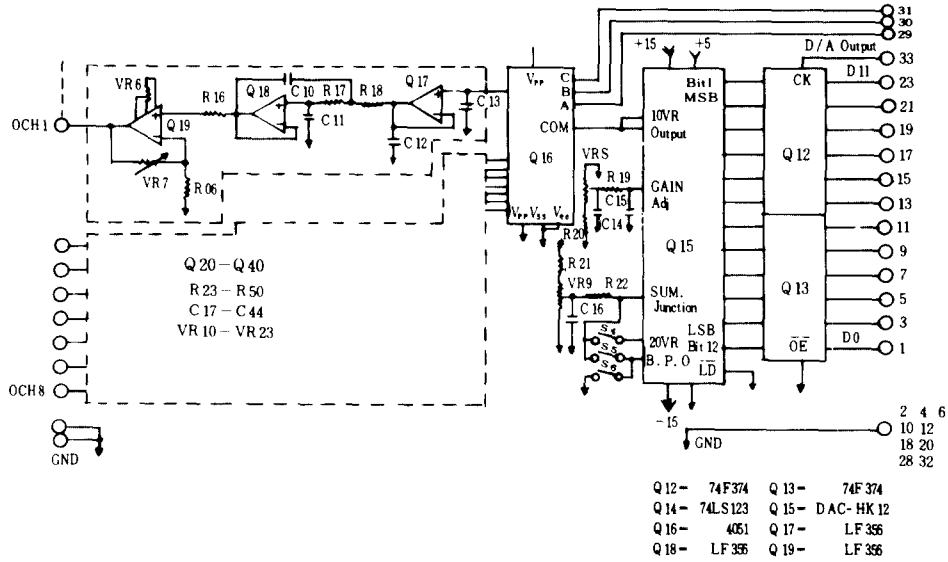

Fig.5. Schematic of output unit.

cients than the direct form realization [4].

IV. Experimental Results

The real-time ALP has been realized as illustrated in Fig.3-5. In this experiment, the smoothing parameter β was chosen as 0.999. Input waveforms to the ALP are 100 Hz sine wave and a pseudorandom wave synthesized by 100 Hz, 320 Hz and 600 Hz sinusoidal waves. The input and output waveforms of the ALP under 100 Hz and 1KHz sinewave excitation were shown in Fig. 6. The output is a steady state waveform after the adaption. This one-step prediction signal is calculated to evaluate the ALP performance. Comparing I/O waveform, it reveals that the high performance of the ALP's prediction capability. Also the output was predicted well under pseudorandom signal excitation as shown in Fig. 7. The transient behavior of the ALP under 100 Hz sine wave and pseudorandom excitation were shown in Fig.8. From this figure, it reaches the steady-state value of prediction error in about 15 time steps respectively. The gradient lattice algorithm is very similar but not identical to the recursive least-square lattice algorithm (RLSL). The computation time of gradient lattice was faster by 20% than RLSL in this study, but the initial convergence behavior of gradient lattice more slowly than RLSL about 10 time steps. Fig.9 shows the interrupt $\overline{BIO}$'s waveform of the DSP chip and output waveform of the ALP. When the $\overline{BIO}$ is low, it represents A/D conversion time duration (22 μsec), and the $\overline{BIO}$ is high, then it represents the operating time (134 μsec) of 4-stages adaptive lattice predictor. Therefore the total execution time of the algorithm was 156 μsec, so it means that the maximum running frequency of this designed real-time ALP is 6.41 KHz.

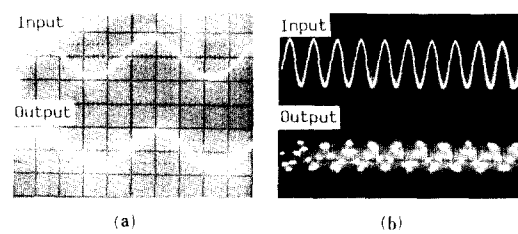

Fig.6. Input and output waveform of the ALP under (a) 100 Hz and (b) 1KHz sinewave input ((a) H:2 msec/div (b)H:1 msec/div, V:5 V/div).

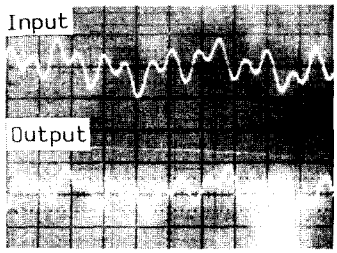

Fig.7. Input and output waveform of the ALP under pseudorandom signal excitation (H:2 msec/div, V:5 V/div).

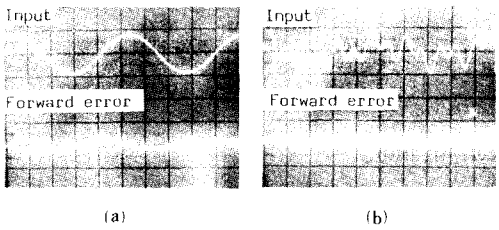

Fig.8. Initial operation of the ALP under (a) sinusoidal wave (b) pseudorandom wave excitation (H:2 msec/div, V:5 V/div).

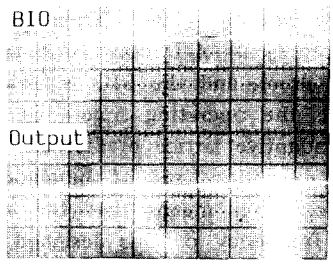

Fig.9. Interrupter BIO's waveform and the output of the ALP (H:50 μsec/div, V:5 V/div).

V. Conclusions

The adaptive lattice predictor which is the basic element for digital signal processing was implemented on the DSP in real time. For 4-stage lattice structure, the maximum running frequency was obtained as 6.41 KHz in this experiment. An improvement by the factor 2 of the speed can be obtained by simply replacing the employed microprocessor with its improved version (TMS320C25). This system can be easily modified for various application areas involving adaptive line enhancer, adaptive noise canceller, adaptive lattice Wiener filter and speech processing by burning into EPROM as firmware.

Reference

[1] F. Itakura and S. Sato, "Digital filtering techniques for speech analysis and synthesis," *in Proc. 7th Int. Cong. Acoust.*, Budapest, pp. 261-264. 1971.

[2] L.J. Griffiths, "A continuously adaptive filter implemented as a lattice structure," *in Proc. IEEE Conf. Acoust., Speech Signal Process.*, pp. 683-686, 1977.

[3] B. Friedlander, "Lattice filters for adaptive processing," *Proc. IEEE*, vol. 70, no. 8, 1982.

[4] J.D. Markel and A.H. Gray, Jr., "Roundoff noise characteristics of a class of orthogonal polynomial structures," *IEEE Trans. Acoust., Speech Signal Process.*, vol. ASSP-23, pp. 473-486, 1975.

[5] TMS32020User's Guide, Texas Instruments Inc., 1985.