

# 마이크로프로세서용 칠회용 고속 Fourier 變換

손 유 익 , 최 흥 문

Son, Yoo-Ek , Choi, Heung-Moon

(경북대학교)

## I. 序 論

離散 Fourier 變換 ( Discrete Fourier Transform ; DFT ) 을 효율적으로 수행하기 위한 고속 Fourier 變換 ( Fast Fourier Transform ; FFT ) 은 현재 디지털 신호처리 분야에서 널리 사용되고 있다.

이론 컴퓨터에서 적용하기 위한 FFT 自体에 대한 연구는 1965년 J.W. Cooley 와 J.W. Tukey 에 의해 그 理論이 體系化된 이래 많은 연구가 진행되어 왔다.

최근에는 소형 및 저價格의 특징을 가지고 있는 마이크로프로세서를 칠회용 FFT 에 대해서도 많은 연구가 진행되고 있다. 1978년 반근후와 成田敏一은 마이크로프로세서에서 整數型 연산으로 처리되는 FFT 프로그램을 作成 하였으며 overflow 를 豫備하여 데이터의 범위를 넘어서는 副 프로그램을 수었다. 또한 같은 해 織田照夫는 實數型 연산을 사용하여 종의 淸정한 結果를 얻을 수 있는 방법에 관하여 연구 발표 하였으나 整數型 연산보다 느린 단점이 있다. 이들은 마이크로프로세서의 소프트웨어 측면 위주 하였기 때문에 전용 FFT 프로세서로서의 시스템은 간단히 할 수 있었으나 몇가지의 특정 信號 처리의 경우 때에는 처리속도면에서 다소 부족한 점이 있다.

本 연구에서는 마이크로프로세서를 칠회용 FFT 에 있어서, 가장 많은 시간을 차지하는 부분이 乘算을 대체하여, 速度도 빠르고 부호도 함께 처리하는 그의 補數 방식 並列乘算機를 제작 하였으며 이것을 소프트웨어 乘算이 代置하여 사용함으로써 이제 다른 속도개선에 대하여 연구 하였다.

## II. 시스템 構成

Decimation-in-Frequency FFT 알고리즘을 채택하여 다음의 6개 부분으로 나누어 FFT 프로그램을 作成 하였다.

- a) overflow 관련 프로그램.
- b) 데이터 正規化 프로그램.
- c) butterfly 연산 프로그램.
- d) 複素乘算 프로그램.
- e) 乘算 프로그램.
- f) bit reversal 의 rearrange 프로그램

本 FFT 프로그램은 수행하기 위하여 마이크로프로세서의 기억 용량을 확장시키고  
 하드웨어 승산기를 제작하여 부속시켜 그림 1과 같은 시스템을構成하였다.

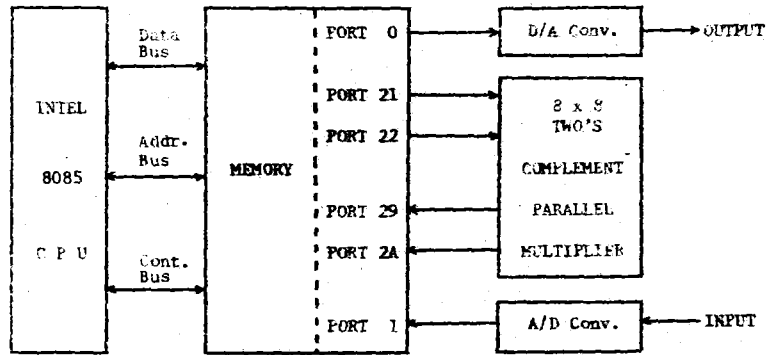


Fig. 1. Block diagram of the FFT processing system using microprocessor with hardware multiplier.

그림 1의 乘算機는 속도개선을 위하여 FFT 프로그램을 처리시간이 가장 많이  
 소요되는 乘算 프로그램의 演算을 대신 하여 高速乘算機 (fast multiplier)로서  
 하드웨어 알고리즘에 의해 그림 2와 같이 구성하였다. 즉,  $m$  bit 의 被乘數  $A$ 를  
 $A = (a_{m-1} a_{m-2} \dots a_0)$ ,  $m$  bit 의 乘數  $B$ 를  $B = (b_{m-1} b_{m-2} \dots b_0)$ 라 하고  
 각각의 2의 補數를  $A_s$  및  $B_s$  라 하면  $A_s, B_s$ 는

$$A_s = -a_{m-1} 2^{m-1} + \sum_{i=0}^{m-2} a_i 2^i$$

$$B_s = -b_{m-1} 2^{m-1} + \sum_{i=0}^{m-2} b_i 2^i$$

이 되기 위하여 最終積 (final product)  $P_s$ 는

$$P_s = (-a_{m-1} 2^{m-1} + \sum_{i=0}^{m-2} a_i 2^i) (-b_{m-1} 2^{m-1} + \sum_{i=0}^{m-2} b_i 2^i)$$

$$= (a_{m-1} b_{m-1} 2^{2m-2} + \sum_{i=0}^{m-2} \sum_{j=0}^{m-2} a_i b_j 2^{i+j}) - (\sum_{i=0}^{m-1} a_{m-1} b_i 2^{m+i} + \sum_{i=0}^{m-2} b_{m-1} a_i 2^{m+i})$$

$$= -p_{2m-1} 2^{2m-1} + \sum_{i=0}^{2m-2} p_i 2^i$$

로 나타낼 수 있다. 위의 (4)식 등에서  $2^{2m-1}$ 의 부호를 가지고 있는 두번째 항을  
 2의 補數로 변환하고 變形된 部分積 配列 (modified partial product  
 array)을 얻어 이 배열에 따라 Full-adder를 사용하여 그림 2와 같은 2의  
 補數方式 並列 乘算機를 구성하였다.

소문트웨어 乘算의 경우 이는 실재의 乘算 번을 위한 프로그램이 같은 번의 이차와 무로  
 변별 프로그램 과의 후가 되어 높은 乘算 번의 소모되나, 본 연구에서 2층 2과 제곱된  
 2의 稀數方式의 高速並列 乘算機를 사용 함으로써 무는 변별 프로그램 필요없고  
 乘算 프로그램 자체도 2층의 乘算 번으로 구성되므로 프로그램의 길이가 짧아지고 여러  
 다 乘算의 수행 시간도 줄어 처져 전체적으로 2층의 乘算 번을 볼 수 있다.

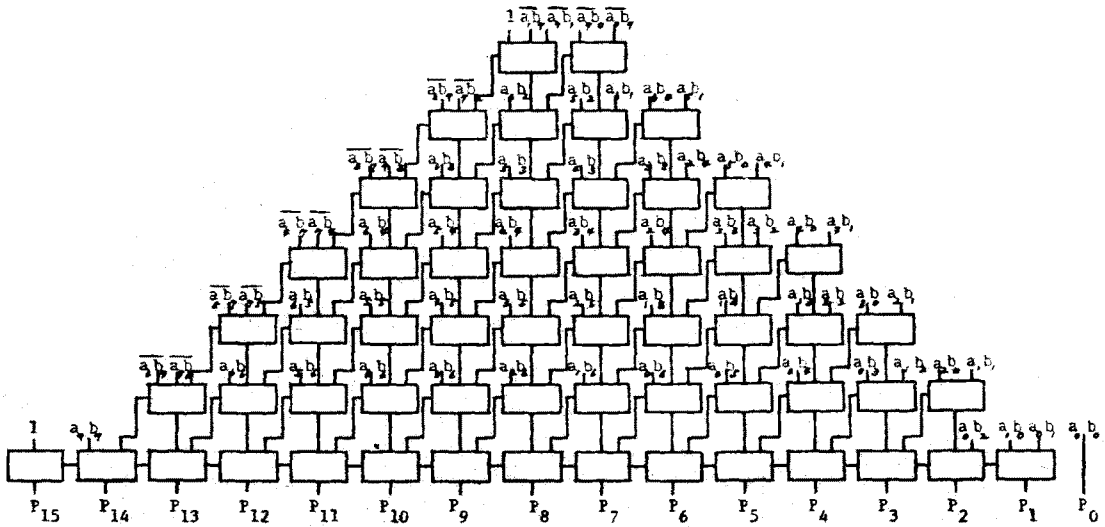


Fig. 2. Two's complement parallel multiplier using Full-adder.

## 夏. 結 論.

INTEL 8085 마이크로프로세서 를 사용 하여, 高速 Fourier 變換에 있어서 처리속도를 개선  
 하는 방법에 의하여 연구 하였다.

高速 Fourier 變換에서 대부분의 처리 시간이 乘算에 소모됨을 기도로 하여 8x8  
 bit 의 2의 稀數方式 並列 乘算機를 구성하고 이를 소문트웨어 乘算과 代置 하여  
 사용 하였다.

그 結果, 마이크로프로세서의 클럭 주파수 3MHz, Sample 數 128 에 대하여 소문트웨어  
 乘算의 경우에는 835 ms 가 소모되었으리, 본 연구에서 제곱된 高速 並列 乘算機  
 를 사용한 경우에는 358 ms 가 소모되어, 약 2.3 배의 속도개선을 볼 수 있었다.

8 bit 마이크로프로세서의 整數型 연산에 의하여 생긴 오차는 HP 3000 등형 컴퓨터의  
 결과와 비교하여 5% 이내 이었다.

따라서 입력 데이터가 8 bit 로 처리될 수 있는 경우 제곱된 乘算의 내에서는 충분히  
 실효성이 있는 시스템이 되리라 생각 한다.