

TMS320C67x 기반 병렬신호처리시스템의 설계와 성능분석

문 병 표[†] · 박 준 석^{††} · 전 창 호^{†††} · 박 성 주^{††††} · 이 동 호^{††††} · 한 기 택^{†††††}

요 약

본 논문에서는 TMS320C67x를 사용한 병렬신호처리시스템의 설계와 성능을 분석한다. 특히 보드 단위의 프로세서부 설계에 중점을 두어 메모리의 구성 및 내부버스 연결방식 면에서 서로 다른 네 가지의 모델을 제안하고 성능분석을 위하여 2D FFT를 병렬로 처리할 수 있는 여러 가지의 방식을 제시한다. 그리고 제안된 구조상에서 여러 방식으로 2D FFT를 실행할 경우의 지역메모리의 접근, 프로세서간 통신, 그리고 보드간 통신에 소요되는 시간을 척도로 하여 네 가지 보드 모델들의 성능을 비교·분석한다.

성능분석 결과, 성능과 보드 구성의 복잡도를 종합적으로 고려할 때 지역메모리와 공유메모리를 함께 갖춘 모델이 가장 바람직한 것으로 나타났다.

Design and Performance Analysis of A TMS320C67x-based Parallel Signal Processing System

Byung-Pyo Moon[†] · Joon-Seok Park^{††} · Chang-Ho Jeon^{†††} · Sung-Joo Park^{††††} ·
Dong-Ho Lee^{††††} · Ki-Taek Han^{†††††}

ABSTRACT

This paper deals with a design and performance analysis of a parallel signal processing system based on TMS320C67x. With an emphasis on the board-level design of the processor unit four models are proposed with different memory configurations and internal bus schemes. Several approaches to parallel processing of 2D FFT are also presented to be used for performance analysis. The performance of four board models are estimated and compared in terms of the time spent for local memory access, inter-processor communication, and inter-board communication.

The results of performance analysis show that, when performance and implementation complexity are taken into account, the model with both local and shared memories is the most desirable.

※ 본 논문은 수중음향특회연구센터(과제번호: UA-23)의 연구지원
금에 의한 연구결과임

† 정 회 원 : (주)베리택 연구원

†† 준 회 원 : 한양대학교 대학원 전자계산학과

††† 종신회원 : 한양대학교 전자컴퓨터공학부 교수

†††† 정 회 원 : 한양대학교 전자컴퓨터공학부 조교수

††††† 정 회 원 : 국방과학연구소 선임연구원

논문접수 : 1999년 6월 7일, 심사완료 : 1999년 10월 22일

1. 서 론

급속한 디지털화의 추세로 영상처리 및 음성처리 또는 통신 등 여러 분야에서 DSP 칩이 광범위하게 사용되고 있지만 고품질 영상처리, 레이더, 소나, 침단 의료 서비스 등과 같이 방대한 양의 데이터 처리를 요구하는 디지털 신호처리 응용분야에서는 단일 DSP 칩만으로는 만족스러운 처리속도를 얻기가 힘들다. 따라서 최근에는 여러 개의 DSP 칩을 다중으로 연결하여 처리속도와 신뢰성을 높인 병렬시스템구조가 디지털 신호처리 응용분야에 많이 사용되고 있다[1]. 예를 들면 음성신호의 효율적인 처리를 위한 시스템이 Paisley 대학에서 개발되었고, Oakland 대학, NTT 등에서는 영상처리를 위한 시스템이 개발되었으며, 군사용의 고성능 신호처리 시스템인 AN/UYS-2가 미 해군에 의해 개발되었다[2-5].

그러나 이들 시스템은 특정한 응용분야에 적합하도록 설계된 구성요소 및 구성방식을 사용하여 개발되었기 때문에 기존 시스템과의 호환성 유지가 어려우며 시스템의 확장성도 떨어진다는 단점이 있다. 뿐만 아니라 범용성이 부족하여 다양한 응용프로그램의 개발에도 많은 어려움이 있다.

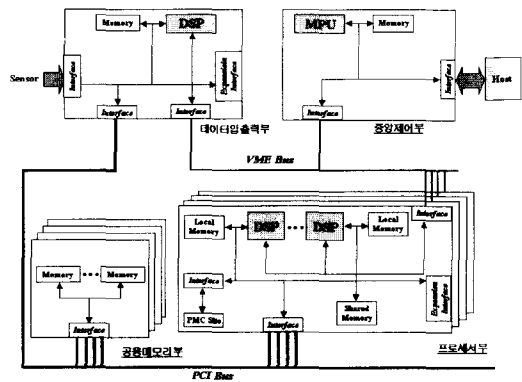
본 논문에서는 상용화된 DSP 칩을 이용하여 고속의 신호처리가 필요한 분야에서 범용으로 사용할 수 있는 병렬신호처리시스템을 제안한다. 이 시스템은 기존의 시스템에 비하여 확장성이 우수하며 데이터 전송과 시스템의 제어를 위한 버스를 각각 분리한 이중버스구조를 채택함으로써 시스템의 전체적인 처리율을 높인 것이 특징이다. 특히 버스연결구조와 메모리의 구성이 다른 네 가지의 프로세서보드 모델을 설계하고 그 성능을 분석한다.

본 논문의 구성은 다음과 같다. 2장에서는 병렬신호처리시스템의 기본구조를 설명하고 이 시스템의 핵심부인 프로세서부를 버스과 메모리의 구성방식에 따라 네 가지의 서로 다른 모델로 제안한다. 3장에서는 성능분석용 벤치마크로써 2D FFT의 병렬처리 방식을 제시하고 4장에서는 2D FFT의 네 가지 병렬처리 방식에 근거하여 네 가지 보드 모델의 성능을 분석하고 비교한다. 그리고 5장에서 결론을 맺는다.

2. 병렬신호처리시스템

2.1 기본구조

(그림 1)은 고속 신호처리 분야에서 사용될 수 있는 병렬신호처리시스템의 기본적인 구성도이다. 이 시스템은 데이터입출력부, 공용메모리부, 중앙제어부, 그리고 프로세서부, 이렇게 크게 4개의 서브시스템으로 구성되어 있다.



(그림 1) 병렬신호처리시스템의 구성도

시스템의 전체적인 동작을 개괄적으로 살펴보면 DSP 칩, 메모리, PCI 버스, 센서 그리고 확장 및 외부버스와 연결을 위한 인터페이스로 구성된 데이터입출력부는 센서로부터 아날로그 신호를 입력받아 디지털 데이터로 변환하는 전처리 과정을 수행한 후에 그 결과를 공용메모리부로 보낸다. 공용메모리부는 보드단위로 용량의 확장이 가능하며 각각의 보드 내에는 다수의 메모리 소자들이 PCI 버스로 연결되어 시스템 전체에서 필요로 하는 데이터를 저장한다. 중앙제어부는 데이터의 흐름과 시스템 전체의 동작을 제어하며 마이크로프로세서(MPU), 메모리, PCI 버스, 그리고 호스트 및 외부버스와 연결을 위한 인터페이스로 구성되어 있다. 호스트는 중앙제어부와 연결되며 사용자 인터페이스를 제공하고 시스템 감시기능을 담당하게 된다. 보드단위로 확장이 가능한 프로세서부는 실제적인 산술·논리 연산을 수행하며 각 보드는 여러 개의 DSP 칩, 지역메모리 또는 공유메모리, PCI 버스, 그리고 확장 및 버스와 보드내부의 구성요소들간의 연결을 위한 인터페이스로 구성된다. 특히, 고속으로 대량의 데이터

이동이 필요한 데이터입출력부, 공용메모리부, 프로세서부 사이의 연결은 전송률이 뛰어난 PCI 버스로, 시스템 제어 및 감시와 프로그램 전송이 필요한 중앙제어부, 프로세서부, 데이터입출력부 사이의 연결은 VME 버스를 사용한 이중버스구조를 가진다.

2.2 프로세서부 모델

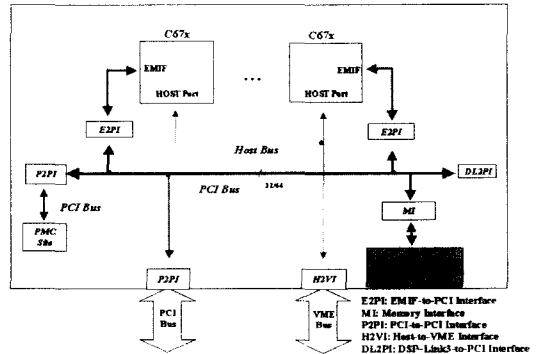
이 시스템의 핵심부인 프로세서부는 여러 개의 TMS320C67x DSP 칩을 가진 보드로 구성되며 보드단위로 확장이 가능하다. 현재 많은 DSP 칩들이 개발되어 사용되고 있지만 산술연산 능력이 뛰어나고 개발환경이 우수한 TI사의 TMS320C67x를 제안한 시스템의 연산소자로 사용하였다[6]. TMS320C67x의 주목할 만한 특징으로는 6ns의 명령어 사이클 시간, 167MHz의 동작속도, 1GFLOPS의 산술연산 성능, 1Mbit의 내부메모리, 400MB/s 대역폭의 32비트 외부메모리 인터페이스(TI사에서는 EMIF 버스라고 명명함), 2개의 직렬포트, 1개의 데이터 전송포트(TI사에서는 호스트포트라고 명명함) 등이 있다[7]. 각종 프로그램이나 제어신호는 VME 버스를 사용하여 중앙제어부로부터 호스트포트를 통하여 프로세서로 전달되고 연산에 필요한 데이터는 PCI 버스를 통하여 공용메모리로부터 프로세서에 공급된다. 각각의 보드 내에는 4개의 프로세서와 보드수준의 전역버스인 PCI 버스, 그리고 프로세서와 메모리의 직접연결을 위한 EMIF 버스가 있다. 또한 보드의 성능 및 기능을 확장하기 위해 사용되는 PMC(PCI Mazzanine Cards) 모듈과 DSP~LINK3를 갖추고 있다.

TMS320C67x에 기반한 병렬시스템으로는 Daytona67, Monaco67, 그리고 Barcelona67 등의 상용 DSP보드가 있다[12]. 이들은 공통적으로 보드내 PCI버스와 일정량의 보드 내 지역메모리 라는 확실적인 구조를 가지고 있어서 응용분야의 특성에 따라서는 최적의 구조라고 할 수 없다. 본 논문에서는 TMS320C67x 특유의 EMIF포트와 호스트포트 까지 활용하여 메모리구성과 프로세서간 통신경로를 달리하는 네 가지의 모델을 제안하고 그 성능을 비교분석 한다.

가) 모델-SO(Shared memory Only)

(그림 2)와 같이 모델-SO는 모든 모델의 공통적인 구성요소와 함께 보드 수준의 공유메모리로 구성되어 있다. 모델-SO의 장점은 보드 내에 프로세서 개별용 지역메모리를 별도로 두지 않고 보드내의 모든 프로세

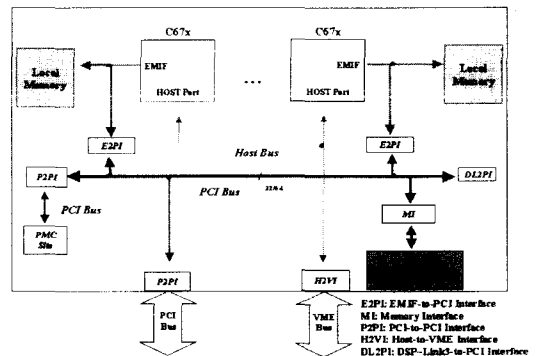
서들이 공유하는 공유메모리만으로 구성되어 있기 때문에 다른 모델들과 비교하여 상대적으로 회로 복잡도가 낮으며 구현이 용이하다는 것이다. 그러나 프로세서들이 처리해야할 데이터가 모두 공유메모리에 저장되어 있으므로 연산을 위한 데이터를 읽어오고 연산결과를 저장하기 위해서는 모든 프로세서가 PCI 버스를 빈번히 사용해야 하고 그로 인해서 많은 지연이 발생할 수 있다는 단점이 있다. 그리고 프로세서간 통신도 PCI 버스를 통한 공유메모리에 의해서 이루어진다.



(그림 2) 모델-SO

나) 모델-SP(Shared and Private memory)

(그림 3)에 나타낸 모델-SP는 보드 내에 프로세서 개별의 지역메모리와 보드수준의 공유메모리를 함께 가진 모델이다. 지역메모리로는 TMS320C67x가 EMIF 버스를 통하여 직접적으로 지원하는 SDRAM과 SBSRAM을 사용하며 별도의 메모리 인터페이스를 필요로 하지 않는다. 프로세서 전용으로 필요한 데이터는 자신의

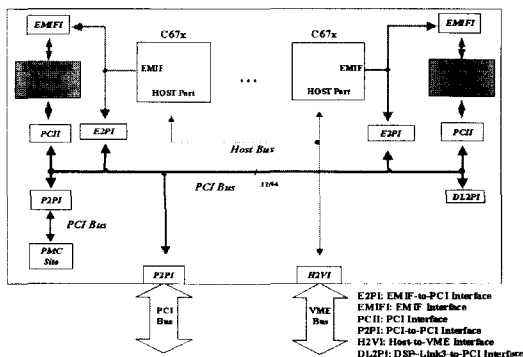


(그림 3) 모델-SP

지역메모리에 저장되고 이 데이터에 대해서는 고속으로 접근이 가능하다. 그리고 프로세서간 통신이 필요한 데이터나 지역메모리의 용량을 초과하는 다량의 데이터는 보드내의 공유메모리에 저장된다. 모델-SP는 지역메모리와 공유메모리를 모두 가지고 있기 때문에 그에 따른 메모리 소자의 수와 인터페이스 회로로 인하여 모델-SO와 비교하여 보드 회로의 복잡도가 크다.

다) 모델-DPO(Dual-port Private memory Only)

모델-DPO는 메모리의 구성에 있어서 (그림 4)와 같이 보드수준의 공유메모리 없이 이중포트 메모리를 사용한 프로세서 개별용 지역메모리만으로 구성된 모델이다. 각각의 프로세서에 전용으로 필요한 데이터는 EMIF 버스를 통하여 고속으로 접근이 가능하고 프로세서간 통신이 필요한 데이터에 대해서는 프로세서 A가 EMIF 버스를 통하여 자신의 지역메모리에 데이터를 기록하고 프로세서 B가 PCI 버스를 통하여 읽어드는 형식으로 이루어진다. 따라서 프로세서간 통신에 있어서 보드내의 PCI 버스에 대한 사용빈도를 줄일 수 있기 때문에 공유버스의 사용으로 인한 지연시간이 감소한다는 장점이 있다. 그러나 현재의 상용화된 이중포트 메모리는 단위용량이 적기 때문에 다량의 데이터를 저장하기 위해서는 메모리의 확장이 필요하고 그로 인하여 메모리 소자의 수가 증가하게 되므로 보드 회로의 복잡도 측면에서는 모델-SO에 비하여 크고 모델-SP와 비슷한 정도이다.

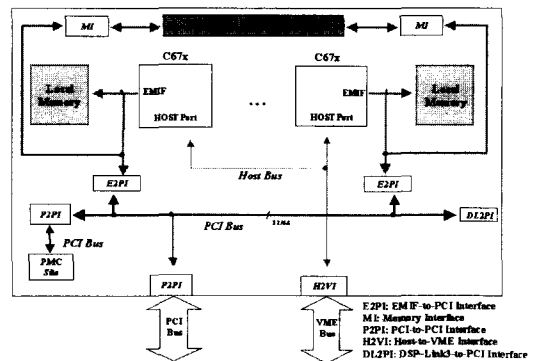


(그림 4) 모델-DPO

라) 모델-QSP(Quad-port Shared and Private memory)

(그림 5)에 나타낸 모델-QSP는 보드 내에 프로세서

개별용 지역메모리와 4중포트 메모리를 사용한 보드수준의 공유메모리를 함께 가진 모델이다. 지역메모리는 TMS320C67x가 지원하는 SDRAM과 SBSRAM을 사용한다. 연산에 필요한 데이터는 각 프로세서의 지역메모리에 저장되고 프로세서간 통신은 보드의 PCI 버스를 사용하지 않고 4중포트 메모리를 통하여 이루어지기 때문에 프로세서간 통신에 있어서 공유버스로 인한 지연은 전혀 없다. 그러나 지역메모리와 공유메모리를 함께 가지고 있기 때문에 메모리 소자의 수는 물론 그에 따르는 인터페이스 회로도 많이 필요하다. 뿐만 아니라, 프로세서와 지역메모리 사이의 연결 및 프로세서들과 4중 포트 메모리의 연결로 인하여 연결구조도 복잡하다. 따라서 보드 회로의 복잡도는 다른 모델들에 비하여 가장 크다고 할 수 있다. 또 상용화된 4중포트 메모리의 접근시간이 보통 50ns로써 20ns 내외인 다른 메모리소자에 비해 월등히 길다는 점이 문제이다.



(그림 5) 모델-QSP

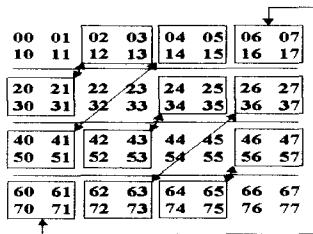
3. 벤치마크 알고리즘

3.1 2D FFT

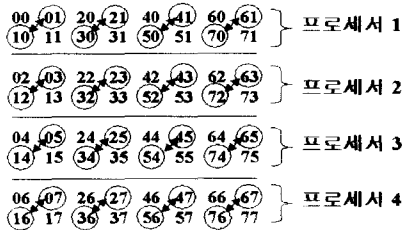
2D FFT는 2차원 데이터에 대해서 FFT를 수행하는 연산으로 유체역학에서 속력벡터, 그라디언트의 계산 그리고 이미지 처리에서 특징추출 등에 많이 필요하다 [8]. 연산량이 많은 2D FFT의 실행은 세 단계로 이루어지는데 첫 번째 단계는 행과 열의 2차원 데이터 매트릭스 중에서 행 방향으로 1D FFT를 수행하는 것이다. 두 번째는 매트릭스 변환을 하는 단계로서 이것은 아직 연산이 이루어지지 않은 열 방향의 데이터를 행

| | | | | | | | | |
|----|----|----|----|----|----|----|----|-----------------|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | } 프로세서 1 (열1,2) |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | |
| 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | } 프로세서 2 (열3,4) |
| 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | } 프로세서 3 (열5,6) |
| 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | |
| 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | } 프로세서 4 (열7,8) |
| 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | |

(a) 초기상태



(b) 서브매트릭스간의 데이터 이동



(c) 서브매트릭스내의 데이터 이동

| | | | | | | | | |
|----|----|----|----|----|----|----|----|-----------------|
| 00 | 10 | 20 | 30 | 40 | 50 | 60 | 70 | } 프로세서 1 (행1,2) |
| 01 | 11 | 21 | 31 | 41 | 51 | 61 | 71 | |
| 02 | 12 | 22 | 32 | 42 | 52 | 62 | 72 | } 프로세서 2 (행3,4) |
| 03 | 13 | 23 | 33 | 43 | 53 | 63 | 73 | |
| 04 | 14 | 24 | 34 | 44 | 54 | 64 | 74 | } 프로세서 3 (행5,6) |
| 05 | 15 | 25 | 35 | 45 | 55 | 65 | 75 | |
| 06 | 16 | 26 | 36 | 46 | 56 | 66 | 76 | } 프로세서 4 (행7,8) |
| 07 | 17 | 27 | 37 | 47 | 57 | 67 | 77 | |

(d) 재구성된 매트릭스

(그림 6) 매트릭스 변환과정의 예

방향 1D FFT가 가능하도록 메모리에 연속적으로 위치시키기 위해 열 방향의 데이터를 행 방향으로 재배치하는 작업이다. 세 번째는 행 방향으로 변환된 열

방향의 데이터에 대해서 다시 행 방향 1D FFT를 수행하는 단계이다.

여기서 2D FFT를 병렬로 처리하는데 한 가지 문제가 되는 것은 사용하는 메모리의 구성방식에 따라서 매트릭스 변환을 하는 동안 데이터 이동량과 그 소요 시간에 차이가 발생한다는 것이다. 메모리의 구성방식에는 공유메모리 방식과 분산메모리 방식이 있다. 공유메모리로 구성된 시스템에서는 공유메모리에 전체 매트릭스가 저장되고, 행 방향으로 1D FFT가 끝난 데이터를 단지 열 방향으로 읽어 들이는 것만으로 매트릭스 변환이 이루어진다. 따라서 매트릭스 변환을 위한 별도의 데이터 이동이 필요없는 반면 공유메모리에 대한 충돌이 발생할 수 있다는 단점이 있다.

분산메모리 방식의 시스템에서는 각 프로세서의 지역메모리에 매트릭스가 분산되어 저장되고 행/열 FFT는 여러 프로세서의 지역메모리 공간에 걸쳐서 실행된다. 특히 매트릭스 변환을 위해서는 매트릭스 일부가 지역메모리 사이를 이동해야 한다. 네 개의 프로세서가 각각 독립된 지역메모리를 가진 분산메모리 시스템에서 매트릭스 변환이 이루어지는 과정을 도식화하면 (그림 6)과 같다[8]. (그림 6(a))는 전체 매트릭스 데이터를 네 개의 프로세서에게 균등하게 분할한 초기 상태를 나타낸다. 변환과정의 첫 단계는 (그림 6(b))에 나타난 것처럼 매트릭스의 열 방향 데이터를 행 방향으로 변환하기 위해서 프로세서간에 매트릭스의 일부(서브매트릭스)를 교환하는 것이다. 이때 역대각선에 있는 데이터는 행/열 방향에 공통으로 나타나는 것이기에 때문에 이동할 필요가 없다. 변환과정의 두 번째 단계는 서브매트릭스간의 데이터 교환이 끝난 후에 각 프로세서의 지역메모리 내에서의 2차적인 데이터 이동 과정으로 (그림 6(c))에 나타나 있다. (그림 6(d))는 전체적인 매트릭스 변환과정이 모두 끝난 후에 새로 구성된 매트릭스이다.

3.2 2D FFT의 병렬처리 방식

앞에서 제안한 네 가지 프로세서부 모델에서 2D FFT를 병렬로 처리하기 위하여 주어진 매트릭스 데이터를 분할하는 방식은 다음과 같이 생각해 볼 수 있다. 그리고 분할방법의 이름에 사용된 약어는 각각의 분할방법의 의미를 나타낼 수 있도록 임의로 부여한 것이다.

가) 방법-SMP : 보드내의 프로세서들에게 하나의 매

트릭스를 균등하게 분배하고 각각의 프로세서는 자신에게 할당된 서브매트릭스에 대해서 2D FFT를 수행한다. 따라서 이 방법은 보드간 통신은 발생하지 않으나 보드내에서의 프로세서간 통신이 빈번하기 때문에 보드 모델들의 프로세서간 통신에서의 성능을 분석하기에 적합한 방법이다.

나) 방법-RCP: 하나의 매트릭스를 두 개의 프로세서가 썬을 이루어 한 프로세서가 행 방향 FFT를 수행하고 저장한 결과에 대해서 다른 프로세서가 열 방향 FFT를 수행하게 된다. 따라서 네 개의 프로세서로 구성된 하나의 보드에서는 두 개의 매트릭스가 동시에 2D FFT 연산이 되며 연산에 필요한 두 개의 매트릭스 데이터를 저장해야 하기 때문에 보드내에 많은 용량의 메모리가 필요한 방법이다.

다) 방법-SMB: 방법-SMP를 보드단위로 확장한 것으로서 매트릭스를 보드들에 균등하게 분배하고 보드 안에서 다시 프로세서들에게 서브매트릭스를 균등하게 나누어 모든 프로세서들은 자신이 가진 서브매트릭스에 대해서 2D FFT를 수행한다. 이 방법은 프로세서간 통신과 보드간 통신이 동시에 필요한 방법으로 빈번한 프로세서간 통신과 보드간 통신으로 인하여 성능의 저하가 있을 수 있다는 단점이 있는 반면, 하나의 매트릭스 데이터가 여러 보드와 프로세서에게 분할되기 때문에 보드별로 필요한 지역메모리 또는 공유 메모리의 용량이 작아도 된다는 장점이 있다.

라) 방법-RCB: 방법-RCP를 보드 단위로 확장한 것이며 하나의 매트릭스를 두 개의 보드가 썬이 되어 하나의 보드는 행 방향 FFT를 수행하고 다른 보드는 열 방향 FFT를 수행한다. 이 방법은 반드시 짝수개의 보드로 구성된 시스템이 필요하며 단일 보드를 가진 시스템에서는 사용할 수 없다.

로세서부 모델에서 수행하여 성능을 분석한다. 성능분석의 척도는 지역메모리의 사용, 프로세서간 통신. 그리고 보드간 통신에서 발생하는 데이터 이동시간과 연산을 위해 프로세서 자체에서 소요되는 연산시간의 합으로 구해진다. 그러나 연산시간은 모든 프로세서부 모델에서 동일하고 TMS320C6xx 시뮬레이터에 근거해 계산해 본 결과 2.81ms 라는 값이 나왔다. 이 시간은 데이터 이동시간에 비해 매우 큰 값으로서 모델간의 성능차이를 나타내기 어렵기 때문에 본 논문에서는 데이터 이동시간만으로 모델들의 성능을 비교한다. 데이터의 이동시간은 매트릭스의 병렬처리 방법에 따라 차이가 있지만 식 (1)에서 처럼 기본적으로 2D FFT를 실행하는데 필요한 지역메모리 접근시간, 프로세서간 통신시간, 보드간 통신시간을 합한 값이다.

$$T_{DT} = T_{LM} + T_{IPC} + T_{IBC} \quad (1)$$

성능분석에 사용된 주요한 파라메터의 수치는 다음과 같다. FFT의 각 포인트는 16비트의 데이터를 가지고 있으며 1024 × 1024 포인트 크기의 데이터를 대상으로 한 단계의 2D FFT를 수행하기 위한 데이터량은 4.19M 바이트, 전체 매트릭스중에서 1 열의 데이터량은 4.1K 바이트이다. 매트릭스 변환을 하는 동안 이동되는 데이터량은 공유메모리를 사용할 경우 4.19M 바이트, 분산메모리를 사용할 경우 3.14M 바이트이다. 그리고 입출력 수행시간에 영향을 미치는 하드웨어 구성 요소로는 SDRAM, 이중포트 메모리, 4중포트 메모리, EMIF 버스, PCI 버스가 있으며 이들을 통한 데이터 전송시간을 <표 1>에 정리하였다. EMIF 버스와 PCI 버스에서의 소요시간은 512 바이트의 데이터를 전송하는데 필요한 시간이다. 메모리들은 EMIF 버스와 PCI 버스의 32비트 데이터폭과 일치시키기 위하여 다중뱅크를 사용하여 구성하였다.

<표 1> 메모리와 버스를 통한 데이터 이동시간

| 하드웨어 종류 | 구성 | 소요시간 |
|----------|------------|--------|
| SDRAM | 16비트 2bank | 1325ns |
| 이중포트 메모리 | 16비트 2bank | 1298ns |
| 4중포트 메모리 | 8비트 4bank | 2580ns |
| EMIF 버스 | 해당없음 | 1280ns |
| PCI 버스 | 해당없음 | 3879ns |

4. 성능분석

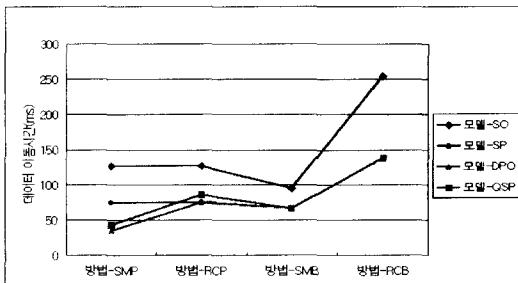
4.1 분석방법

3.2절에서 제시한 병렬처리 방법에 따라 1024 × 1024 포인트 2D FFT를 2.2절에서 제안한 네 가지 프

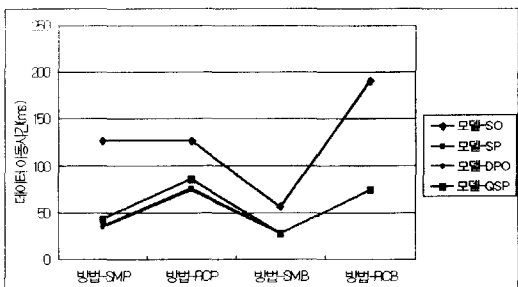
4.2 분석결과

이상의 조건으로 병렬처리 방법에 따른 프로세서부 모델의 성능분석 결과를 (그림 7)과 (그림 8)에서 보이고 있다.

(그림 7)은 공유메모리 방식으로 동작하는 네 가지 모델에서 네 가지의 병렬처리 방법으로 2D FFT를 수행할 경우 지역메모리의 접근, 프로세서간 통신, 그리고 보드간 통신에서 발생하는 데이터 이동시간의 합을 나타낸 것이다. 전체적으로 모델-SO가 가장 긴 데이터 이동시간을 보였으며 모델-DPO가 가장 짧은 데이터 이동시간을 보였다. 그리고 모델-SP와 모델-QSP는 중간정도의 성능을 나타내었다. 병렬처리 방법-SMP에서는 물리적인 접근시간이 가장 짧은 이중포트 메모리로 구성된 모델-DPO가 가장 우수한 성능을 보였다. 그리고 모델-SO와 모델-SP가 매트릭스 변환을 위하여 보드내의 PCI 버스를 통한 보드의 공유메모리를 사용하기 때문에 낮은 성능을 보였으며 특히, 모델-SO는 공유메모리만으로 구성되어 있는 모델이므로 모든 데이터의 접근에 있어서 공유메모리를 사용해야 하기 때문에 가장 낮은 성능을 보였다. 병렬처리 방법-RCP에서도



(그림 7) 공유메모리 방식에서의 성능분석 결과



(그림 8) 분산메모리 방식에서의 성능분석 결과

모델-SO의 성능이 가장 낮게 나타났으며 다른 메모리에 비하여 상대적으로 매우 느린 접근시간을 갖는 4중 포트 메모리로 구성된 모델-QSP도 낮은 성능을 보였다. 병렬처리 방법-SMB와 방법-RCB에서는 모델-SO를 제외한 나머지 모델들의 데이터 처리방식이 공용메모리부를 사용하는 동일한 방식으로 이루어지기 때문에 성능이 동일하게 나타났다.

(그림 8)은 네 가지 모델이 분산메모리 방식으로 동작할 때 네 가지의 병렬처리 방법으로 2D FFT를 수행할 경우 지역메모리의 접근, 프로세서간 통신, 그리고 보드간 통신에서 발생하는 데이터 이동시간의 합을 나타낸 것이다. 데이터 이동에 소요되는 시간은 전체적으로 모델-SO가 가장 길고 모델-DPO가 가장 짧았다. 그리고 모델-SP와 모델-QSP는 중간정도의 성능을 나타내었다. 병렬처리 방법-SMP와 방법-RCP에서는 모델-QSP의 성능이 가장 낮게 나타나는데 그 이유는 데이터의 교환에 사용하는 4중 포트 메모리의 물리적인 성능이 나쁘기 때문이다. 그리고 병렬처리 방법-SMB와 방법-RCB에서는 공유메모리 방식에서 처럼 모델-SO를 제외한 나머지 모델은 데이터 교환을 위한 버스의 사용방식이 동일하기 때문에 모두 같은 성능을 보였다.

5. 결 론

본 논문에서는 TMS320C67x DSP칩을 사용하여 신호처리 응용분야 및 범용으로 사용할 수 있는 병렬신호처리시스템을 제안하고 시스템의 핵심부인 프로세서부를 메모리와 버스 구조가 다른 네 가지 모델로 제안하고 성능을 비교·분석하였다.

2D FFT를 이용한 성능분석과 보드의 복잡도 관점에서 결론을 내리면 다음과 같다.

지역메모리 접근에는 속도가 빠른 이중포트 메모리를 사용한 모델-DPO가 좋지만 실용되고 있는 이중포트 메모리소자의 개별 용량이 적기 때문에 많은 소자가 필요해 보드의 복잡도가 증가한다는 단점을 가지고 있다. 따라서 제안한 시스템의 연산소자인 TMS320C67x가 지원하는 SDRAM을 사용하는 모델-SP가 복잡도와 성능면에서 가장 적당한 모델이다.

프로세서간 통신에서는 모델-DPO와 모델-QSP가 적당한 프로세서부 모델이다. 그 중에서도 모델-QSP가 4중 포트 메모리를 사용하여 프로세서간 통신에는 큰 장점을 가지고 있다. 그렇지만 보드의 복잡도가 커져 비용

이 많이 들고 실제적인 메모리 소자의 성능이 SDRAM, 이중포트 메모리에 비해 떨어지기 때문에 구조적인 장점이 잘 드러나지 않는다는 단점을 가지고 있다.

보트간 통신에서는 모든 모델들이 단일버스를 사용하기 때문에 거의 동일한 성능을 나타낸다. 따라서 다중버스를 사용하거나 다단계 연결망, 매쉬, 크로스바 등 통신총선이 직교, 동시에 여러 모듈 사이의 통신이 가능한 연결구조[9-11]를 사용한다면 성능을 향상시킬 수 있을 것이다.

이상의 분석결과를 종합해 보면 2D FFT의 실행 성능과 복잡도 측면에서 모델-SP가 가장 효율적인 프로세서부 모델이라고 할 수 있다. 그러나 응용분야의 특성에 따라, 예를 들면 프로세서간 통신시간이 많이 요구되는 응용분야의 경우 모델-DPO나 모델-QSP의 구성을 택할 수 있을 것이다.

참 고 문 헌

- [1] A. J. Anderson, "Selection Criteria in the Development of a Multiple Processor Based DSP System," *Journal of Microcomputer Application*, pp.327-345, 1992.
- [2] P. V R Raja and S. Ganesan, "An SIMD Multiple DSP Microprocessor System for Image Processing," *Microprocessors and Microsystems*, pp.493-501, Nov. 1991.
- [3] T. Sawabe, T. Fujii, H. Nakada, N. Ohta, and S. Ono, "A 15GFLOPS Parallel DSP System for Super High Definition Image Processing," *IEICE Trans. Fundamentals*, Vol.39, No.7, pp.786-792, Jul. 1992.
- [4] F. Kurugollu, H. Palaz, H. Gumuskaya, E. Harmanci and B. Orencik, "Advanced Educational Parallel DSP System Based on TMS320C25 Processors," *Microprocessors and Microsystems*, Vol.19, No.3, pp.147-156, Apr. 1995.
- [5] *Navy Signal Processors Program Office*, PMS 429.
- [6] W. W. Smith and J. M. Smith, *Handbook of Real-Time Fast Fourier Transform*, IEEE Press, 1995.
- [7] *TMS320C6x Technical Brief*, Texas Instruments, 1998.
- [8] *Parallel Processing With the TMS320C4x*, Texas Instruments, 1994.
- [9] T. N. Mudge, J. P. Hayes and D. C. Winsor, "Multiple Bus Architectures," *IEEE Computer*, Vol.20, No.6, pp.42-48, Jun. 1987.
- [10] L. N. Bhuyan, Q. Yang, and D. P. Agrawal, "Performance of Multiprocessor Interconnection Networks," *IEEE Computer*, pp.25-37, Feb. 1989.
- [11] T. Lang, M. Valero, and I. Alegre, "Bandwidth of Crossbar and Multiple-Bus Connections for Multiprocessors," *IEEE Trans. Computers*, pp.1227-1233, Dec. 1982.
- [12] http://www.spectrumsignal.com/products/index.cfm?page2load=c6x/c6701_hardware.cfm, Spectrum Signal Processing

문 병 표



e-mail : bpmoon@mail.veritek.co.kr
 1997년 수원대학교 정보통신공학과 졸업(학사)
 1999년 한양대학교 전자계산학과 졸업(석사)
 1999년~현재 (주)베리텍 연구원
 관심분야 : 컴퓨터구조, 병렬처리 시스템, 성능분석

박 준 석



e-mail : jspark@para1.hanyang.ac.kr
 1993년 한양대학교 전자계산학과 졸업(학사)
 1995년 한양대학교 대학원 전자계산학과(석사)
 1998년 한양대학교 대학원 전자계산학과(박사과정 수료)
 관심분야 : 컴퓨터구조, 병렬처리시스템, 성능분석

전 창 호



e-mail : chjeon@para1.hanyang.ac.kr
 1977년 한양대학교 전자공학과 졸업(학사)
 1986년 Cornell 대학교(박사)
 1986년 성균관대학교 전기공학과 조교수
 1989년~현재 한양대학교 전자컴퓨터공학부 교수
 관심분야 : 컴퓨터 구조, 병렬처리시스템, 성능분석



박 성 주

e-mail : parksj@cse.hanyang.ac.kr
 1983년 한양대학교 전자공학과 졸업(학사)
 1983년~1986년 금성사 소프트웨어 개발
 1988년 Univ. of Massachusetts 전기 및 컴퓨터공학과 졸업(석사)

1992년 Univ. of Massachusetts 전기 및 컴퓨터공학과 졸업(박사)

1992년~1994년 IBM Microelectronics 연구스텝

1995년~현재 한양대학교 전자컴퓨터공학부 조교수

관심분야 : 테스트 합성, Built-In Self Test, Scan Design, ATPG, ASIC 설계



이 동 호

e-mail : dhlee@image.hanyang.ac.kr
 1986년 한양대학교 전자공학과 졸업(학사)
 1988년 The University of Texas at Austin 졸업(석사)
 1991년 The University of Texas at Austin 졸업(박사)

1991년~1994년 LG전자 중앙연구소 선임연구원

1994~현재 한양대학교 전자컴퓨터공학부 조교수

관심분야 :

한 기 택

e-mail : kthan@sunam.kreonet.re.kr
 1982년 동국대학교 전자공학과 졸업(학사)
 1993년 부산대학교 전자공학과 졸업(석사)
 1982년~현재 국방과학연구소 선임연구원