

디지털 신호처리 기능을 강화한 32비트 마이크로프로세서

문상국

목원대학교 정보전자영상공학부

A 32-bit Microprocessor with enhanced digital signal process functionality

Sangook Moon

Mokwon University, School of Information-Electronics-Image Engineering

E-mail : smoon@mokwon.ac.kr

요 약

본 논문에서는 16비트 혹은 32비트 고정 소수점 연산을 지원하는 디지털 신호처리 기능을 강화한 명령어 축소형 마이크로프로세서를 설계하였다. 설계한 마이크로프로세서는 명령어 축소형 마이크로 아키텍처의 표준에 따라서 범용 마이크로프로세서의 기능과 디지털 신호처리 프로세서의 기능을 함께 갖추고 있다. 산술연산기능 유닛, 디지털 신호처리 유닛, 메모리 제어 유닛으로 구성되어 있으며, 이 연산 유닛들이 병렬적으로 수행되어 디지털 신호처리 명령어나 로드/스토어 명령어의 지연된 시간을 보상할 수 있게 설계되었다. 이 연산유닛들을 병렬적으로 동작하게 함으로써 5단계 파이프라인의 구조로 고성능 마이크로프로세서를 구현하였다.

ABSTRACT

We have designed a 32-bit microprocessor with fixed point digital signal processing functionality. This processor, combines both general-purpose microprocessor and digital signal processor functionality using the reduced instruction set computer design principles. It has functional units for arithmetic operation, digital signal processing and memory access. They operate in parallel in order to remove stall cycles after DSP or load/store instructions, which usually need one or more issue latency cycles in addition to the first issue cycle. High performance was achieved with these parallel functional units while adopting a sophisticated five-stage pipeline stucture.

키워드

Microprocessor, digital signal processing, reduced instruction set computer

1. 서 론

최근 추세에 따르면 내장형 시스템은 유연성, 지능성 등을 포함하여 여러 가지 기능을 갖추면서도 또한 비용도 저렴해야 한다. 저비용 내장형 프로세서에 마이크로컨트롤러 기능과 디지털 신호처리 기능을 수반한 제품들이 필요하게 되었으며, 이는 보다 발전된 가전 응용에 쓰이게 되었다. 두 개의 분리된 코어를 단순하게 결합하는 형태는 비용면에서 효율적이지 않기 때문에, 단일

코어로 구성된 마이크로컨트롤러와 디지털 신호처리 기능의 복합코어가 필요하게 된다. 이는, 단순히 자원을 두 배로 하지 않으며, 프로그래밍에 유연한 특성을 가지는 RISC [1][2] 마이크로프로세서의 기능과 함께 고성능 디지털 신호처리 기능을 동시에 제공하는 것을 의미한다. 또한, RISC 마이크로프로세서 아키텍처에 기반을 둬으로써, 높은 클럭 주파수를 사용하여 범용 마이크로프로세서 연산과 디지털 신호처리 연산을 동시에 수행할 수 있다.

II. 프로세서 아키텍처

논문에서 설계한 프로세서의 파이프라인 구조는 단일 위상 클럭 주입 방식을 사용하였다. 제안하는 프로세서는 ALU, DSP, load/store 연산에 대해 독립적이고 병렬적인 파이프라인 스테이지를 가지고 있다. ALU 파이프라인은 일반적 RISC 파이프라인 구조와 마찬가지로 5단으로 구성되어 있고, 전체 마이크로프로세서의 제어를 담당한다. 프로세서의 주요 유닛들에 대해 아래와 같이 설명한다. 그림 1은 설계된 마이크로프로세서의 레이아웃 도면이다.

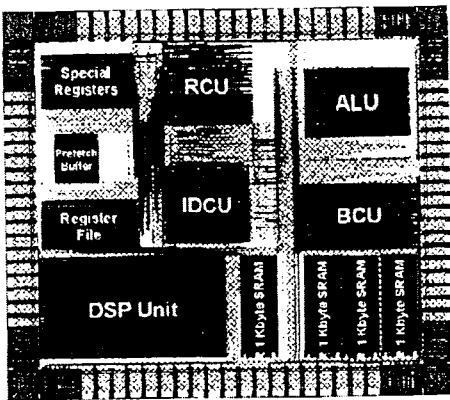


그림 1. 제안한 마이크로프로세서의 레이아웃

II.1 명령어 디코드/제어 유닛

명령어 디코드/제어 유닛은 프리페치 버퍼로부터 명령어를 받아 그 명령어를 디코드한다. 그리고 난 후, 데이터 의존성이나 자원 충돌과 관계되거나 많은 명령어들은 각각 관련된 기능 유닛으로 이슈된다 [3]. 이 유닛은 명령어 디코딩을 통하여 즉치 (immediate) 오퍼랜드를 기능 유닛으로 보내고, 레지스터 주소 데이터를 레지스터 제어 유닛으로 보낸다. 또한, 트랩을 발생시키기도 하는데, 실행 유닛에서 감지된 인터럽트나 익셉션 신호에 의해 발생시킨다 [4]. 파이프라인 제어 동작으로서 모든 다른 유닛들에게 해당 파이프라인에서 어떤 동작을 수행해야 하는지 알려주는 신호를 발생시킨다.

II.2 레지스터 제어 유닛

레지스터 제어 유닛은 두 개의 소스 레지스터의 주소와 하나의 목적지 레지스터의 주소를 계산하여 데이터 값을 레지스터 파일에 기록한다. 이 유닛에서, 목적지 주소는 파이프라인 단계의

Writeback 단계를 사용하고, 로드 (load) 레지스터의 주소는 로드 파이프라인을 사용한다. 여기서 데이터 의존성이 (dependency) 검사되는데, 이 결과는 명령어 디코드/제어 유닛이 인터록 (interlock) 조건을 감지하는데 사용된다. 데이터 포워딩을 위한 제어 신호들은 파이프라인 Decode 단계의 소스레지스터의 주소와 동 단계의 이전 명령어의 목적지 주소와 비교함으로써 추출할 수 있고 이 신호가 각 기능 유닛에 전달된다.

II.3 산술 논리 연산 유닛 (ALU)

산술 연산 유닛은 덧셈/뺄셈을 수행하는 덧셈기, 논리 부울 연산을 수행하는 논리연산기, 쉬프트, 그리고 비교기로 구성되며, 많은 수의 ALU 연산들은 실행 유닛들을 공유한다. 나눗셈 연산은 덧셈기와 쉬프트를 단계적으로 사용하여 구현하였으며, 쉬프트와 비교기를 사용하여 소스 오퍼랜드의 선행 0의 개수를 (leading zeros) 계산한다. 추가적인 주소 발생기를 독립적으로 할당하는 대신, ALU에서 메모리의 유효주소를 계산하도록 설계하였기 때문에, 최적화된 ALU 공간 효율을 보일 수 있다.

II.4 버스 제어 유닛

버스 제어 유닛은 모든 메모리의 접근 영역을 제어한다. 메모리 접근은 두 단계의 파이프라인 구조로 되어 있어서 하나의 double-word 명령어 혹은 두 개의 single-word 명령어가 지연 없이 수행될 수 있다. 파이프라인의 Execution 단계 동안, ALU는 유효 주소를 계산하고 이를 주소 파이프라인에 넘겨 준다. 로드 명령어의 경우, 로드 레지스터의 주소가 로드 파이프라인으로 전달되고 스토어 (store) 명령어인 경우 저장될 데이터의 값이 스토어 파이프라인으로 전달된다. ALU 혹은 DSP 명령어들은 메모리 접근 명령어들과 독립적으로 처리된다.

III. 구현과 검증

정수 테스트 프로그램과 DSP 테스트 프로그램을 사용하여 성능평가를 한 결과, 명령어당 사이클 (CPI) 개수가 대부분 테스트 프로그램에 있어서 1.3~1.9개로 측정되었다. 자세한 결과는 그림 2에 설명하였다. 설계한 마이크로프로세서의 CPI가 1이 넘는 이유는 다중 사이클 명령어와 지연 사이클이 있기 때문이다. 지연 사이클이 생기는 이유는 데이터의 의존성, 로드/스토어 버퍼가 꼭차는 문제, 충분하지 못한 명령어의 공급 등의 원인을 들 수 있다. 프로그램과 데이터를 따로 저장하는 하바드 (Harvard) 구조를 도입함으로써 이러한 문제들을 어느 정도 해결할 수는 있지만, 내장형 시스템의 효율에 대비하여 비용이 너무 높

게 된다.

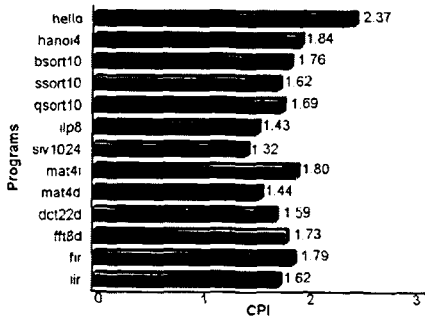


그림 2. 정수, DSP 프로그램에서의 CPI

V. 결 론

본문에서 설계한 내장형 마이크로프로세서는, RISC 마이크로프로세서의 장점과 더불어 디지털 신호처리 기능을 결합시켜 가격대비 효율적인 동시에 높은 성능을 낼 수 있도록 고안되었다. 하나 이상의 이슈 지연 사이클을 갖는 명령어들로 인한 지연을 최소화 하기 위하여, 산술연산을 위한 기능 유닛들과, DSP 연산 유닛, 또한 메모리 접근 명령어들이 병렬적으로 수행되도록 설계하였다. RISC 구조에 기반한 5단계로 이루어진 파이프라인을 채택함으로써, 기존 DSP 프로세서에 비하여 높은 주파수에서 동작하였다.

참고문헌

- [1] M.G.H. Katevenis, Reduced Instruction Set Computer Architectures for VLSI, MIT Press, Cambridge, Massachusetts, 1985.
- [2] D.A. Patterson, "Reduced instruction set computers," Commun. ACM, vol.28, no.1, pp.8-21, Jan. 1985.
- [3] S. Weiss and J.E. Smith, "Instruction issue logic for pipelined supercomputers," Proc. 11th International Symposium on Computer Architecture, pp.110-118, Ann Arbor, Michigan, June 1984.
- [4] J.E. Smith and A.R. Pleszkun, "Implementing precise interrupts in pipelined processors," IEEE Trans. Comput., vol.37 ,no.5, pp.562-573, May 1988.