

내장형 32비트 마이크로컨트롤러에 적합한 VARIABLE PIPELINE STAGE 설계

정영석, 양동훈, 박승호, 이문기
연세대학교 전기전자공학과
전화 : 02-2123-4731 / 핸드폰 : 011-9013-4759

SMART7F: VARIABLE PIPELINE STAGE FOR 32-BIT MICROCONTROLLER

Young-Seok Cheong, Dong-Hun Yang, Seung-Ho Kwak, Moon-Key Lee
Dept. of Electrical Electronics Eng., Yonsei University
E-mail : yscheong@spark.yonsei.ac.kr

Abstract

In this paper, the soft IP (Intellectual Property) of pipeline of 32-bit microcontroller for embedded and portable application is presented. This IP supports variable pipeline stage according to the performance that user wants. In this architecture, three pipeline stages are basically employed and extended to the five pipeline stages. To this purpose, control logic has been partitioned to reflect each pipeline stage.

FPGA platform is used for rapidly prototyping the IP. This is designed using Verilog HDL

I. 서론

프로세서 기술의 진보에 따라서 시스템 온 칩(SoC)을 설계하는 것은 PDA, 핸드폰, 게임기등의 내장형 프로세서 시스템에서 유용해지고 있다.

집적 기술이 고도로 발달함에 따라서 많은 다양한 기능을 작은 칩에 집적할 수 있게 되었지만, 그로인해 설계 시간이 많이 소요되어 시장진입속도가 늦어져서 산업적인 응용분야에서 경쟁력 제고에 문제가 발생하고 있다.

요즘에 들어서 IP(Intellectual Property)을 사용하는 설계는 다양한 기능을 가진 시스템의 개발 시간을 단축할 수 있는 중요한 방법으로 자리잡아가고 있다. 다만 IP를 유용하게 사용하기 위해서는 개발시에, 재사용

성을 충분히 고려하고, 다양한 요구조건을 수용할 수 있도록 설계되어야 한다.

마이크로 컨트롤러는 SoC 시스템에서 가장 많이 사용되는 기능 블록이다.[1] 이 블록에 대한 유용한 IP를 개발하는 것이 중요한 문제가 되고 있다. 다양한 응용 범위를 갖는 마이크로 컨트롤러는 사용자의 의도에 따라서 짧은 개발기간과 고성능에 대한 적절한 교환이 이루어져야 한다.

여기서는 이러한 고려사항에 대해 적절히 대응할 수 있도록 파이프라인의 단 수를 융통성 있게 변형할 수 있는 하드웨어 구조에 대해서 2장에서 살펴보고, 파이프라인의 단 수를 변경함에 따라서 생기는 문제들, 특히 성능 열화에 큰 요인이 되는 사항에 대한 개선사항을 살펴보겠다. 4장에서는 위에서 고려된 구조적인 특성을 FPGA를 통해서 하드웨어적으로 구현하고 그에 따른 성능을 고찰해보았다.

II. VARIABLE PIPELINE STAGE

구조

일반적으로 마이크로컨트롤러에서 파이프라인 구조를 채택하여 성능을 향상할 수 있다. 이 때, 파이프라인의 단 수(number of pipeline stage)가 적절하게 고려되어야 한다. 일반적으로 파이프라인의 단 수가 증가하면 설계상의 복잡도가 크게 증가하고, 분기 명령이나 예외 처리루틴에 의해 사이클의 손실이 생기고 이로인해 개발기간이 크게 증가하게 된다. 반면에, 단

수를 증가하므로써 각 단의 사이클 시간(cycle time)을 줄여서 마이크로컨트롤러의 동작주파수를 증가시킬 수 있다.[2]

대부분의 마이크로컨트롤러에서는 여러 기능 유닛을 제어하고, 파이프라인을 정지(stall)시키고, 예외처리를 수행하는 일을 하나의 단일한 유닛에 의해 수행하는 중앙집중식 컨트롤유닛을 사용한다.(그림1 참조) 마이크로컨트롤러를 설계할 때, 이 컨트롤유닛이 가장 복잡한 부분이 되어 설계 시간이 많이 소요되며, 기능을 확장할 때도 설계를 빠르게 진행하는데 장애가 된다.

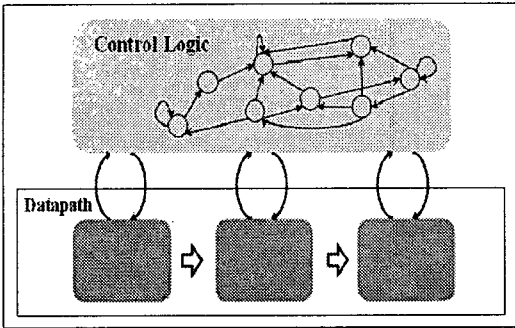


그림 1. Control Unit in a conventional design

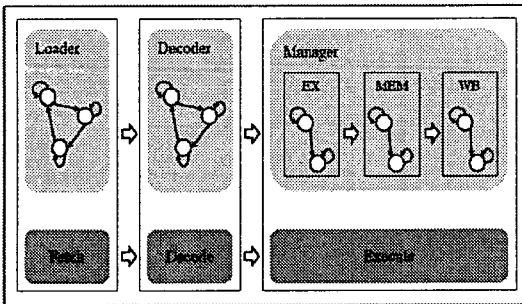


그림 2. Partitioned control unit

그러므로 중앙집중식 컨트롤 유닛을 작고 제어하기 쉬우면서 확장이 용이한 유닛으로 분해하여 variable pipeline control unit을 구현할 수 있다.[3] 이것은 설계자가 마이크로컨트롤러의 용도에 맞춰서 손쉽게 파이프라인의 단 수를 선택하여 최대 5단의 파이프라인 구조로 변경할 수 있다.(그림 2 참조)

기본적인 구조는 Fetch(F), Decode(D), Execute(E)의 3단의 구조를 갖는다. F stage의 컨트롤러는 Loader이고, D stage는 Decoder, E stage는 Manager이다. 3 단 파이프라인 구조에서 E stage는 ALU연산, 주소계산, 메모리 접근등을 수행하기 때문에 많은 사이클이 소요된다. 여기서 E stage를 기능별로 분류하여 각각의 컨트롤 블록을 두어 execution, memory access, write back기능을 구분하여 수행하며 세부적으

로 EX, MEM, WB 컨트롤러에 의해 제어된다.

III. 파이프라인 컨트롤

파이프라인의 단 수를 증가시키므로써 마이크로컨트롤러의 동작주파수를 개선시킬 수 있지만 디펜던시가 더욱 높은 빈도로 발생하게 되면 기껏 개선된 성능을 오히려 열화시킬 수 있다. 그러므로 디펜던시는 파이프라인 단 수를 변화시킬 때 중요하게 고려되어야 할 사항이다.

몇 개의 연속적인 명령어들 중에서 동일한 메모리 어드레스를 접근하게 되면 리소스상에서 충돌이 발생하게 되고, 이런 경우를 구조적 디펜던시(structural dependency)라고 한다. 여기서는 포트가 하나인 메모리 구조를 사용하기 때문에 메모리 참조를 하게 되면 발생할 수 있다. 3단 파이프 라인에서는 높은 성능을 기대하는 것이 아니라 빠른 개발시간을 요구하고 있으므로 이것에 대한 고려는 크게 하지 않았다. 이런 경우에는 후속하는 명령어를 1 클럭 사이클 동안 정지시키므로써 디펜던시를 해결한다. 그러나 4단 파이프라인이나 5단 파이프라인 구조에서는 마이크로 컨트롤러의 성능을 열화시키는 주요한 원인이 되므로 디펜던시를 해결하기 위한 추가적인 구조가 고려되어야 한다.

리소스 충돌에 의한 구조적인 디펜던시를 해결하기 위해서 여기서는 프리페치 버퍼를 두어 성능향상을 얻었다.(그림 4 참조)

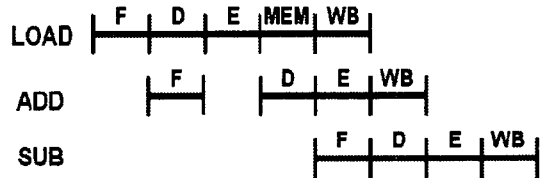


그림 3. Non-prefetch buffer

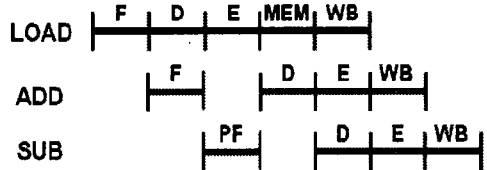


그림 4. prefetch buffer

그림 3을 참조하면 프리페치 버퍼를 사용하지 않을 때 1사이클의 손실이 발생하는 것을 알 수 있다. LOAD 명령어가 4번째 사이클에서 메모리 참조를 하고 있기 때문에 SUB 명령어는 4번째 사이클에서 인출되지 못하고 1사이클 후인 5번째 사이클에서 명령어

를 인출하고 있다. 그러나 그림 4를 참조하면, 프리페치 버퍼를 사용하여 1사이클의 손실을 줄이는 것을 확인할 수 있다.

프로그램에서 명령어의 결과를 레지스터에 저장해야만 사용할 수 있는 구조에서는 다음 명령어가 실행하는 명령어의 결과를 오퍼랜드로 사용할 경우 데이터 디펜던시가 발생한다. 이러한 경우는 포워딩(Forwarding)기법을 이용하여 데이터 패스를 조정해줌으로써 간단하게 문제를 해결할 수 있다.(그림5 참조)[4] 실행유닛(Execution unit)과 라이트백유닛(Write back unit) 사이에서 실행유닛으로 데이터패스를 연결한다. LOAD 명령어의 경우에 파이프라인을 1사이클 정지시키는 로드인터락(load interlock)이 5단 파이프라인에서 발생한다.

여기서는 포워딩의 콘트롤은 MEM과 WB 콘트롤러에 의해 수행된다. 로드인터락 콘트롤은 디코더에서 수행된다. 3단 파이프라인을 사용한 경우는 포워딩 기법을 사용하지 않는다.

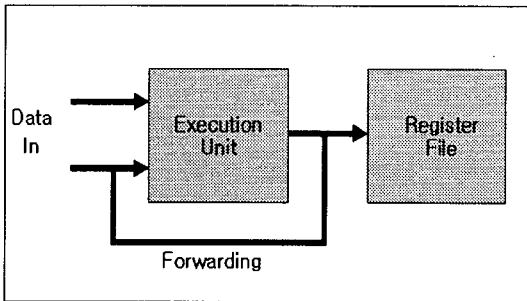


그림 5. Forwarding

IV. 하드웨어 구현과 성능결과

프로세서 성능에 대한 시험은 실제적인 하드웨어 구현을 통해서만 가능하기 때문에, FPGA 프로토타입이 성능 테스트를 하기 위해서 사용된다. 물론 마이크로컨트롤러의 다양한 기능을 추가적으로 검증하는 것도 가능하다.

앞서 기술한 특성을 고려하여 Variable Pipeline Stage를 적용한 32-bit 마이크로컨트롤러 (32-bit SMART7F)를 합성하여 FPGA에 포팅하여 성능 테스트를 수행했다.

FLASH와 SRAM이 명령어메모리와 데이터메모리로 사용되었다. 또한 보드는 LED디스플레이를 갖고 있어서 시뮬레이션 결과를 바로 확인할 수 있다. 시뮬레이션을 위해 디지털 시그널 프로세싱 프로그램을 만들어 사용하였다. 아래 표 1을 참조하면 설계된 디자인의 파이프라인 단수에 따라서 성능의 변화를 확인할 수 있다.

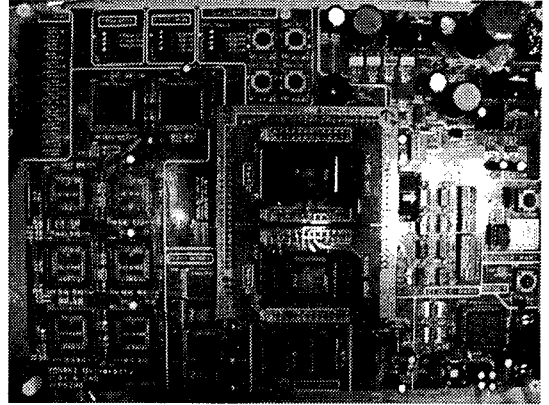


그림 6. FPGA 프로토타입 보드

파이프라인 단수	3단	4단	5단
동작 주파수	30Mhz	70Mhz	80Mhz
트랜지스터 수	32,005	34,512	35,100

표 1. 마이크로컨트롤러의 동작주파수와 칩사이즈

V. 결론

본 논문에서는 마이크로컨트롤러의 성능에 중요한 요인이 되는 파이프라인의 단 수를 선택할 수 있는 구조를 서술하였다. 파이프라인 단 수를 증가함으로써 발생하는 디펜던시는 프리페치 버퍼와 포워딩을 사용하여 해결하였다. 현재의 집적기술을 고려할 때 콘트롤 로직에 의한 칩사이즈 증가보다 데이터패스나 캐쉬에 의한 증가가 크리티컬한 고려사항이므로, 단 수가 증가함에 따라 늘어난 사이즈는 중요한 고려사항은 아니다. 많이 사용되는 3단 파이프라인 구조의 경우 개발 시간이 짧지만 성능이 약하고, 파이프라인의 단수가 증가할 수록 성능은 개선되지만 개발기간이 늘어난다. 이러한 개발여건하에서 개발자가 손쉽게 파이프라인의 단 수를 선택할 수 있는 구조를 이용한 소프트 아이피(soft IP)를 제시했다. 파이프라인의 단 수를 IP상에서 손쉽게 바꿔주으로써 개발자가 임베디드 시스템이나 포터블 어플리케이션에 맞춰서 적합하게 사용할 수 있다.

참고문헌(또는 Reference)

- [1] John L. Hennessy and David A. Patterson, "Computer Architecture: A Quantitative Approach, Second Edition", Morgan Kaufmann, 1996
- [2] Mike Johnson, "Superscalar Microprocessor

- Design." New Jersey: Prentice Hall, 1991.
- [3] Michael Gschwind, Ventina Salapura, and Dietmar Maurer, "FPGA Prototyping of a RISC Processor Core for Embedded Applications," IEEE transactions on Very Large Scale Integration Systems, Vol. 9, NO.2, April, 2001
- [4] Dezso Sima, Terence Fountain, Peter Kacsuk, "Advanced Computer Architectures: A Design Space Approach", Addison-Wesley, 1997