

ARMv7 Core를 위한 3-way SuperScalar Decoder 설계

김 효원, 김 인수, 백 철기, 민 형복
성균관대학교 정보통신공학부

3-way SuperScalar Decoder Design for ARMv7 Core

Hyo Won Kim, In Soo Kim, Chul Ki Baek, Hyoung Bok Min
School of Information&Communication Engineering, Sungkyunkwan University

Abstract - Further evolutions of technologies and needs of users will make mobile equipments improved. To make this happen, processor's good performance is essential. Hence, This paper propose a reform of Instruction Execute and Instruction Decode of contemporary ARMv7 which needs low-power and has the high performance for a faster processor. The first chapter explains why the performance of a processor has to be upgraded, the second chapter shows current technologies. The third chapter explains about the proposal and illustrates the structure. Finally, in the forth chapter, the conclusion will be made. 3-way SuperScalar, that is proposed in this paper, will make designing a faster processor possible. And it will contribute for the advanced performance of mobile equipments.

리된 하드웨어를 이용하는 것이고, 꽤 복잡한 하드웨어 구조를 필요로 한다. 요즘 대부분의 CPU는 파이프라인 방식에 SuperScalar 기술을 접목시키고 있다[2][4].

현재 ARMv7 core에서는 2-way SuperScalar 방식을 사용한다. 이 방식의 핵심은 Instruction Decode 과정에서 2개의 명령어를 동시에 해석하여 Instruction Execute 과정에 각각 다른 연산을 수행한다. Instruction Execute 과정에서는 2개의 ALU pipe와 1개의 MUL pipe, LS pipe를 사용하여 동시에 2개의 명령어에 관한 연산을 수행할 수 있다[4].

따라서 본 논문에서는 이러한 2-way SuperScalar 방식을 개선하여 ARMv7 core의 성능 향상을 위한 3-way SuperScalar 방식을 제안한다.

1. 서 론

오늘날 ARM을 이용하지 않는 모바일 기기는 거의 없을 정도로 ARM을 이용한 응용의 범위가 광범위하게 늘어나고 있다. 작지만 혁신적인 저전력 프로세서 ARM은 모바일 기기에 사용되기 적합한 조건을 갖추고 있기 때문이다[1][2].

또한 제품의 기능이 복잡해지면서 많은 연산을 해야 하는 상황이 많기 때문에 성능 또한 무시할 수 없는 기준이 되어 버렸다. 이러한 요구를 충족시키고자 도입된 것이 현재 ARMv7에서 사용되는 2-way Superscalar 방식이다[4].

하지만 앞으로의 기술 발전을 고려해 본다면 더욱 빠르고 정확한 프로세서가 필요할 것이다. 이에 본 논문에서는 2-way SuperScalar 방식을 개선할 수 있는 3-way SuperScalar 방식을 제안한다.

2. 본 론

현재 혁신적인 기술의 발달과 개개인의 삶의 수준의 향상으로 사용자의 요구 조건이 높아지고, 이에 따라 전자 제품의 질이나 기능적인 면이 상당히 높은 수준에 도달해 있으며, 이는 앞으로도 꾸준히 발전해 나갈 것이다. 이러한 발전은 프로세서의 성능 개선이 뒷받침이 되어야 할 것이다.

지금까지 프로세서의 성능 개선을 위해 많은 노력들이 이어져 왔다. 그 중 가장 혁신적이고 효과적이었던 것은 파이프라인 방식이었다. SuperScalar 방식은 파이프라인 방식과 더불어 프로세서의 성능을 한차원 개선시켰다[3].

SuperScalar 방식은 2개 이상의 명령어를 동시에 해석하여 각각 분리된 하드웨어로 보내지게 된다. 이 구현 방법은 한 클럭당 여러 개의 명령어를 인출하여 각각의 다른 하드웨어(그래픽, FPU) 등에 보낼 수 있다. 이것은 본

3. 결 과

3-way SuperScalar 방식에서는 Instruction Decode 과정에서 동시에 3개의 명령어를 해석한다. 또한 Instruction Execute 과정에 한 개의 ALU pipe를 추가하여 분기와 PC 카운터 등의 연산을 수행함으로써 2-way SuperScalar 방식보다 더 많은 연산을 동시에 처리할 수 있다. 이러한 과정을 통해 보다 빠른 프로세서를 설계한다.

우선 얼마나 처리가 빨라지는가에 대해 알아보기 위해 아래 <그림2>와 같이 Decoder와 ALU pipe 1개씩을 추가하고 dependency는 무시한 채 controller를 설계한다. 그 후 JUMP 명령어에 한하여 명령어 실행에 걸리는 시간이 어느정도 단축되는지 simulation 해 보았다. 그 결과, 클럭당 처리속도가 약간 느려지기는 했지만 같은 수의 명령어를 처리하는 데 소요되는 클럭은 줄게 됨으로써 2-way SuperScalar(<그림3>)에 비해 약 1.5배에 가까운 속도 향상을 확인할 수 있었다. 물론 이 수치는 JUMP 명령어에 한해 실험한 결과이지만, 실제 dependency 관계까지 모두 고려한다고 해도 약 1.3배 이상의 처리 속도의 향상을 기대할 수 있을 것이다.

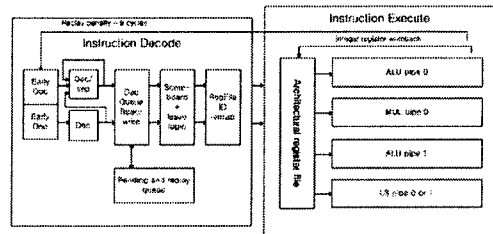


그림 1. 2-way SuperScalar Architecture[4]

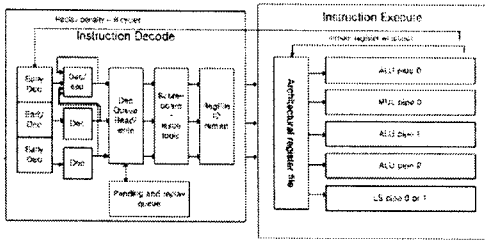


그림 2. Proposed 3-way SuperScalar Architecture

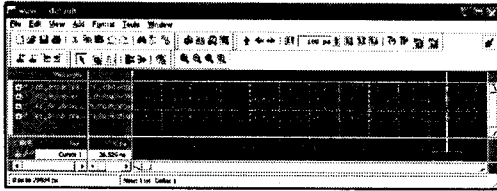


그림 3. 2-Way SuperScalar의 실행 결과 파형

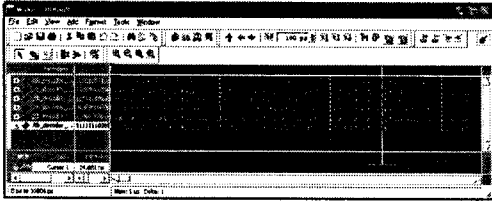


그림 4. 3-Way SuperScalar의 실행 결과 파형

4. 결 론

본 논문에서는 기존의 ARMv7의 2-way SuperScalar 방식을 개선한 3-way SuperScalar 방식을 제안함으로써 앞으로의 모바일 기기 발전의 조건이 되는 프로세서의 성능을 향상시키고자 하였다. 단순히 한 개의 명령어를 더 해석할 수 있는 Decoder와 명령문 분기 또는 PC 카운터의 연산을 처리할 수 있는 ALU를 추가함으로써 동시에 3개의 명령어를 해석하여 수행할 수 있는 방법을 고안하였다. 이에 따라 병렬 연산을 통해 연산 과정에 걸리는 시간을 단축시킴으로써 더욱 빠른 ARM 프로세서를 기대할 수 있을 것이다. 또한 이러한 방식으로 동시에 여러 개의 명령어를 해석하여 수행할 수 있는 다른 방법을 고안하여 더욱 발전된 파이프라인 방식과 접목한다면, 얼마든지 더 빠른 프로세서를 설계할 수 있을 것이라 생각한다.

[참 고 문 헌]

- [1] Andrew N. Sloss, Dominic Symes, Chris Wright, "ARM System Developer's Guide", Cyber Lab System, 2005.
- [2] Furber, Steve b., "Arm System Architecture", Tmecca, 1996.
- [3] John L. H., David A. P., "COMPUTER ARCHITECTURE - A Quantitative Approach", Morgan Kaufmann, 2007.
- [4] Richard Grisenthwaite, "Cortex A8 Processor", ARM Ltd, 2007.