

저에너지 내장형 프로세서를 위한 변형 루프버퍼

¹박정규², ²오형철
¹고려대학교 대학원 전자정보공학과
²고려대학교(서창) 공학부
 ohyeong@korea.ac.kr

A Modified Loop Buffer for a Low-Energy Embedded Processor

¹JeongGyu Park², ²Hyeongcheol Oh
¹Dept. of Elec. & Info. Engineering, Graduate School, Korea University
²School of Engineering, Korea University at Seo-Chang

요 약

루프버퍼는 루프의 부하를 줄이기 위해 일반적으로 사용되고 있는 구조이다. 본 논문은 EISC 내장형 프로세서의 에너지 소모를 줄이기 위하여 변형된 루프버퍼를 제안한다. 제안하는 루프버퍼는 EISC 프로세서가 갖는 특수 명령어의 수행 횟수를 감소시켜, 주요 에너지 소모원인 메모리 접근을 추가로 감소시킨다. 시뮬레이션 결과, 제안하는 루프버퍼는 설계한 프로세서의 수행시간을 5%~13.6% 감소시키며, 메모리 접근횟수를 14.9~37.8% 감소시키는 것을 관찰하였다. 변형된 루프버퍼는 0.18 μ m, 1.8V 공정 표준 셀 라이브러리를 사용하여 약 2792 개의 등가 게이트에 해당하는 면적에서 구현할 수 있다.

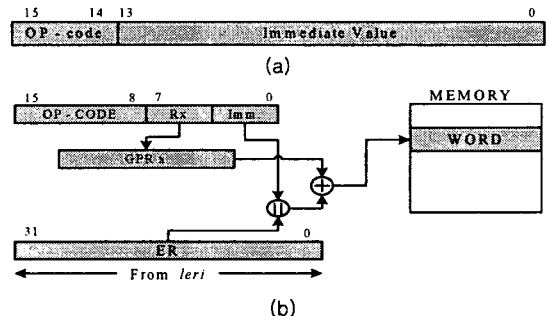
1. 서 론

RFID 태그나 PMP등의 배터리로 동작하는 기기들의 이용이 증가함에 따라, 내장형 프로세서 코어 자체의 에너지 소모 감소 기법에 관한 관심도 날이 증가하고 있는 추세이다. 이에 다양한 관련 기법들이 연구되고 있는데, 본 논문에서는, 프로세서의 에너지 소비의 많은 부분이 메모리 접근에서 발생한다는 관점에서, 프로세서 코어의 메모리 접근을 줄이는 기법에 관심을 둔다.

메모리 접근을 줄이는 기법들 중, 루프버퍼(Loop Buffer: LB)를 사용하는 기법은 루프를 효율적으로 처리하기 위하여 특히 DSP 프로세서들에서 일반적으로 채용되는 기법이다[1]. 루프버퍼는 흔히 프로그램의 제어에 따라, 정해진 횟수만큼 수행할 정해진 수의 명령들을 저장하는 버퍼인 바, DSP 응용은 아니더라도 빈번히 사용되는 작은 루프를 갖는 응용을 반복하여 수행하는 내장형 시스템에서 매우 유용할 수 있다.

본 논문에서는 EISC(Extendable Instruction Set Computer) 구조 기반의 내장형 프로세서 SE3208LE의 전력 성능을 개선하기 위하여 변형된 루프버퍼를 제안한다. 높은 코드 밀도와 적은 메모리 접근을 하는 EISC 구조[2]는 16bit 명령어들과 함께, 즉치 값 크기의 부족을 해결하기 위하여 *leri* 라고 하는 독립적인 명령어를 사용한다(그림 1(a)). 그림 2(b)에 예시한 바와 같이, *leri* 명령어는 즉치 값을 ER(Extension Register)이라고 하는 특수 레지스터에 적재함으로써 큰 즉치 값을 산출할 수 있다. 몇 개의 연속된 *leri* 명령어들이 하나의 긴 피연산자로 합쳐질 수 있는데, *leri* 명령어는 다양한 형태로 확장 사용이 가능하다[3]. *leri* 명령어의 빈도는 15% 이하이나[2], 루프버퍼 기법을 이 경우에 효율적으로 적용하는 방안을 모색하고 그 전력 절감 효과를 알아보고자 함이 본 문 의 동기이다.

본 논문의 구성은 다음과 같다. 제2절에서 SE3208LE 프로세서의 구조를 소개하고, 제3절에서 변형 루프버퍼를 제안한다. 제4절에서는 제안하는 변형 루프버퍼를 평가하고, 제5절에서 결론을 맺는다.

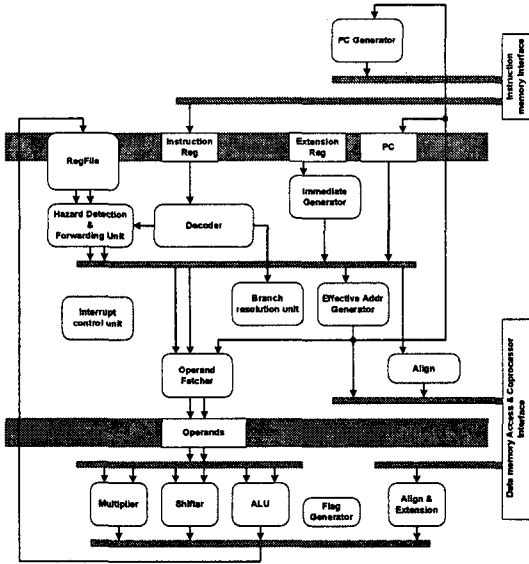


[그림 1] *leri* 명령어[2]. (a) *leri* 명령어 포맷. (b) ER에 이미 적재된 즉치 값의 사용.

2. SE3208LE

SE3208LE는 32비트 EISC구조 중에서 30MIPS 이하 저비용 응용을 대상으로 하는 SE3208 ISA를 구현하는 프로세서이다. 그림2는 SE3208LE의 마이크로 구조를 도시한 것이다. 그림2에서 보인 바와 같이, SE3208LE는 3단 파이프라인 구조로 되어있으며 16비트의 명령어 버스를 갖는다. EISC구조의 특징인 *leri* 명령어들이 프로세서 성능에 미치는 영향을 최소화하기 위한 기법이 개발되어 있으나[4], SE3208LE는 저비용 프로세서이므로 이 기법을 구현하지 않고, *leri* 명령어들을 명령 메모리로부터 하 나씩 패치하여 ER 레지스터를 채운다. SE3208LE는 에너지 소비를 줄이기 위하여 파이프라인 레지스터와 레지

스터파일들에는 클럭게이팅(Clock-gating) 기법을, 연산 유닛에는 연산자 독립(Operator-isolation) 기법을 적용하는 등, 일반적인 저전력 기법 [2]들을 적용하였다.

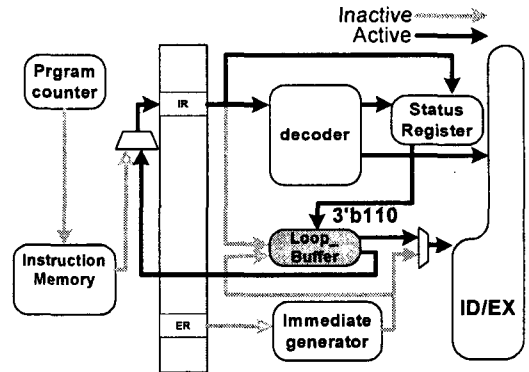


[그림 2] SE3208의 마이크로 구조 [5]

패치해온 명령어와 생성된 즉치값이 루프버퍼에 저장한다. 단 패치해온 명령이 leri 명령일 때는 루프버퍼 콘트롤러는 이를 인지하고 루프버퍼에 저장하지 않고 기다리다가 연속되는 leri 명령 처리 후에 최종 유효 명령과 확장된 즉치 값을 저장하게 한다.

3) 2회 이후의 루프 반복 수행

SE3208LE는 I-메모리 패치에 관한 모든 동작을 하지 않는 대신, 루프버퍼에 저장되어있는 명령어들을 패치하여 수행한다 (그림 3).



[그림 3] 2회 이후 데이터의 흐름

3. 변형 루프버퍼

본 절에서 제안하는 변형된 루프버퍼는 SE3208LE 내부의 상태레지스터(Status Register : SR)의 10번, 8번, 1번의 유효 비트를 사용하여 프로그램으로 제어된다.

루프버퍼의 동작 제어는, 프로그래머가 가장 루프가 빈번하게 일어나는 지점을 찾아 그 시작 지점에 SR의 10번 비트를 셋 시켜주는 명령을 삽입하는 것으로 이루어진다. 그러나 내부적으로는, [표 1]에 보인 바와 같이 SR 비트의 상태가 응용프로그램의 루프 반복 상황에 따라 바뀌며 루프버퍼를 제어하게 된다.

SR bit No.			상태
10	8	1	
3'b0	0	0	루프의 비활성상태
3'b1	0	0	루프의 1회 루프 수행상태
3'b1	1	0	루프의 2회부터 ~ 최종 반복수행상태
3'b1	1	1	루프버퍼는 활성이지만 대기상태

[표1] SR 의 제어 비트에 따른 루프버퍼의 상태

수정된 루프버퍼에는 명령 메모리(I-메모리)로부터 패치해온 명령어와 즉치값이 한번에 저장되는데 동작은 다음과 같다.

1) 루프가 아닐 때
루프버퍼는 비활성 상태로 대기하고 있다.

2) 1회 루프 수행
외부 메모리에서 패치해온 명령어를 수행함과 동시에,

4. 평가

4.1 방법

설계된 SE3208LE 프로세서를 Verilog로 모델화하여, Synopsys사의 Design Compiler를 사용하여 합성하였다. 합성에는 SMIC(Semiconductor Manufacturing International Corporation)사의 0.18µm 공정 표준 셀 라이브러리를 사용하였으며, 성능의 평가에는 Cadence사의 SimVision 과 NC-Verilog 시뮬레이터(Simulator)를 사용하였다.

동작 속도의 평가에는 최악의 조건(worst case process, worst case tree, 125°C junction temperature, 1.65V supply voltage)을 적용하였다. 성능 측정을 위해서, [6]에서 선택된 벤치마크 소스 프로그램을 SE3208용으로 포팅하고 se3208-elf-gcc 컴파일러를 통해 이진 코드로 변환하여 시뮬레이터에 대한 입력으로 사용하였으며, 메모리 모델은 SMIC Memory Compiler를 사용하여 생성한 이상적인 메모리모델을 이용하였다.

4.2 결과

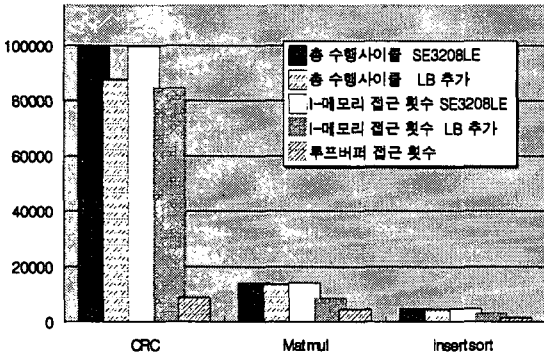
4.2.1 면적 및 수행 사이클 변화

루프버퍼의 구현 비용은 표 2와 같다. SE3208LE에 192B 크기의 LB와 제어회로가 추가되어 면적은 프로세서 코어 대비 약 16.18% 정도 증가하였다. (메모리가 포함된 전체 프로세서 면적에 대해 약 1.5%미만 증가로 평가된다.)

구분	core (gates)	임계경로(ns)
SE3208LE	22775	15.24
SE3208LE + LB	26459.8	15.24

[표2] 구현 비용과 속도에 미치는 영향

그림 4은 3가지의 벤치마크를 수행시켜, l-메모리의 접근 횟수와 프로그램 수행 시간(cycle)을 비교한 것이다. 전체 수행 시간은 약 5%~13.6% 줄어들었으며 l-메모리 접근 횟수는 14.9%~37.8% 감소되었음을 볼 수 있다.



[그림 4] 수행 시간(cycle) 및 메모리 접근(회)

4.2.2 루프버퍼의 수행 결과 분석

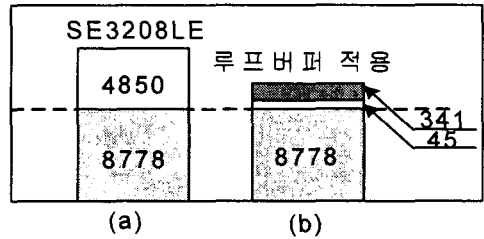
벤치마크 수행 시 수행된 명령어 중 루프버퍼에 저장되어 수행된 명령어의 비율은 표 3과 같다. 루프버퍼는 전체 수행된 명령어 중 10.2~36.5%의 명령어를 처리하고 있고, 루프가 2회째 반복될 때부터 leri 명령어를 수행하지 않기 때문에 SE3208LE 대비 6~9.4%의 명령어가 감소되었음을 볼 수 있다. 이는 전체 명령어중 제외된 leri의 명령어 비율과 같은 것이다. 반면 루프버퍼를 제어하기 위해 최대 3.6%이만의 제어명령이 추가 수행되었다.

벤치마크	수행한 명령어 비율		
	루프버퍼	제외된 leri 명령	추가 명령
CRC	10.23 %	6.42 %	1.0 %
matmul	36.45 %	9.36 %	3.54 %
insertsort	34.9 %	8.73 %	3.4 %

[표 3] 루프버퍼를 사용하여 처리된 명령어 비율

그림 6은 CRC 벤치마크에서, 루프적용이 가능한 세 부분을 선택하여 수행할 때(루프는 총 338회 수행되고 128개 명령어를 포함하며 그 중 45개가 leri 명령어 임), 전체 루프를 수행하면서 SE3208LE가 수행한 명령어 수와 SE3208LE에 LB를 추가한 구조가 수행한 명령어 수의 차이를 보여주고 있다. 그림 6(a)에 도시한 바와 같이, SE3208LE만으로 수행할 때는 총 13628개의 명령어가 수행되는데, 이 중 4850개가 leri 명령어이다. 한편, 그림 6(b)에 도시한 바와 같이, 루프버퍼를 추가하면 첫 루프 수행에서 45개의 leri 명령어가 수행되고, 이후에 leri

명령어는 수행되지 않는다. 그러나, 루프가 시작될 때마다 루프 컨트롤을 위한 제어 명령이 1회 수행되고 (총 338회), 각각의 루프 종료 시에 루프를 초기화하기 위한 제어 명령이 1회 씩, 총 3회 수행된다. 즉, 총 341개의 제어 명령이 추가로 수행된 반면, 4779 개의 leri 명령의 수행이 감소된 것이다.



[그림 6] 루프버퍼에 의한 명령어 수의 감소

5. 결론

본 논문에서는 30MIPS 이하 저비용 응용을 대상으로 하는 SE3208LE 프로세서에 적합하게 변형된 루프버퍼를 제안하였다. 제안된 루프버퍼는 프로그래머의 설정에 따라, 루프 수행 시 외부 메모리로부터 명령어 패치를 중단하고 내부의 버퍼를 사용함으로써 외부 메모리 접근 횟수를 줄인다. 3개의 벤치마크에 대한 실험 결과, 제안된 루프버퍼는 기존의 저에너지구조로 설계된 SE3208LE 프로세서에 비하여 루프내의 leri 명령을 감소시켜 전체 수행 시간을 줄이며 14.9%~37.8%의 l-메모리 접근횟수를 줄일 수 있음을 관찰하였다. 제안된 루프버퍼는 0.18μm, 1.8V 공정에서 약 2792개의 등가 게이트에 해당하는 면적에 구현되며, RFID tag나 센서노드와 같이 저에너지가 요구되는 다양한 응용에 유용하게 사용될 수 있을 것이다.

감사의 글

본 연구는 (주)에이디칩스 및 반도체설계교육센터(IDECC)의 지원을 받아 수행되었습니다.

7. 참고 문헌

- [1] Uh, G.-R. et al: Techniques for Effectively Exploiting a Zero Overhead Loop Buffer. *Computational Complexity*, (2000), pp. 152-172.
- [2] Kim, H.-G., Oh, H.-C.: A Low-Power DSP-Enhanced 32-bit EISC Processor. *HiPEAC2005 - Lecture Notes in Computer Science (LNCS)*, Vol 3793 (2005), Springer-Verlag, pp. 302-316.
- [3] Park, G.-C. et al: Supports for Processing Media Data in Embedded Processors. *HiPC (2004) Poster Present.*, <http://www.hipc.org/hipc2004/posters/park.doc>.
- [4] Oh, H.-C. et al: AE32000: An Embedded Microprocessor Core," *Proc 2nd IEEE (AP-ASIC)*, 2000, pp. 255-258.
- [5] Park, J.-G., Oh, H.-C.: A 32-bit Low-Energy Embedded Processor. *to be presented in JCCI 2006*.
- [6] <http://www.mrtc.mdh.se/projects/wcet/benchmarks.html>