

내장형 프로세서의 가변 목표형 설계 기법

유정영⁰ 김현규* 이기원 오형철**

고려대학교 대학원 전자정보공학과

*(주)에이디칩스

**고려대학교(서창) 자연과학대학 공학부

(jyryu, babyworm, kuwon, hyeong)@atlas.korea.ac.kr

Retargetable Design Scheme of an Embedded Processor

Jung-Young Ryu⁰ Hyun-Gyu Kim* Ki-Won Lee Hyeong-Cheol Oh**

Dept. of Elec. & Info. Eng., Graduate School, Korea University, Korea

*Advanced Digital Chips, Inc., Samsung-2Dong, Kangnam-Ku, Seoul, Korea

**School of Engineering, Korea University at Seo-Chang, Chung-Nam 339-700, Korea

요 약

본 논문은 AE32000 프로세서를 특정 응용 목적의 시스템에 최적화 시키기 위하여 개발한 방법을 소개한다. 내장형 프로세서 시스템에서는 프로세서가 수행할 작업의 내용이 미리 알려진 경우가 많다. 이와 같이 작업내용이 정해진 경우 최적의 명령어 집합을 정하고 사용하는 것이 시스템의 동작 속도와 제작 비용 및 전력 소모에 매우 큰 영향을 줄 수 있다. 각기 다른 목적의 내장형 시스템에 사용할 수 있으며 각 응용 목적에 최적화된 프로세서를 얻기 위해서는 가변 목표형 설계 기법이 요구된다.

1. 서 론

내장형 시스템의 주요 특징중의 하나로 특정 목적의 기능만을 수행한다는 점을 들 수 있다[1]. 따라서 내장형 프로세서 시스템에서는 각 시스템의 용도에 따라 프로세서가 수행할 작업의 내용이 미리 정해지게 되며, 프로세서에 의해 실행되는 명령어들 역시 고정되게 된다. 이와 같이 프로세서의 작업 내용이 정해진 경우 최적의 명령어 집합을 정하고 사용하는 것이 시스템의 동작 속도와 제작 비용 및 전력 소모에 매우 큰 영향을 줄 수 있다. 특히 내장형 시스템에 요구되는 고성능(high performance), 저비용(low cost), 저전력(low power)을 지향하기 위해서는 미리 정해놓은 최적의 명령어 집합에 대한 프로세서의 최적화가 요구된다.

한편, 내장형 프로세서 시스템의 용도에 따라 최적의 명령어 집합은 서로 다르게 나타나게 되며, 이는 곧 각 용도마다 개별적인 프로세서의 최적화 작업이 요구됨을 의미한다. 이러한 문제점을 해결하기 위한 방안으로써 가변 목표형 프로세서의 설계가 필요하다. 이는 최근 들어 Time-to-Market의 중요성이 증가함에 따라 더욱 부각되고 있는 부분으로써 응용분야에 가장 적합한 프로세서를 빨리 얻을 수 있다는 장점이 있다.

본 논문은 AE32000 프로세서[2]를 응용 분야에 최적화 시키기 위하여 개발하는 가변 목표형 설계 기법을 소개한다.

본 논문의 구성은 다음과 같다. 제2절에서는 AE32000 프로세서를 간략히 소개하고, 제3절에서는 가변 목표형 프로세서 설계를 위한 방법 및 코드 생성기에 대해 다루며, 제4절에서 가변 목표형 설계 기법의 실제 적용 예에 대해 설명한 뒤, 제5절에서는 결론을 맺도록 한다.

2. AE32000 프로세서

AE32000 프로세서는 코드 밀도(code density)를 높이기 위해 제안된 EISC(Extendable Instruction Set Computer) 아키텍처[3] 기반의 32비트 프로세서 코어로서 16비트 명령어와 32비트 데이터/어드레스 버스를 지니며 5단계 파이프라인 및 하바드 아키텍처를 채용하였다. 또한 16개의 일반 목적 레지스터(general purpose register), 7개의 특수 목적 레지스터(special purpose register)를 가지고 있으며 32x32=64 곱셈기와 DSP 연산에서 많이 사용되는 32x32+64=64 MAC 연산기 및 32비트 배럴 쉬프트(barrel shifter)를 내장한다. 그림 1은 AE32000의 블록 다이어그램을 보여준다.

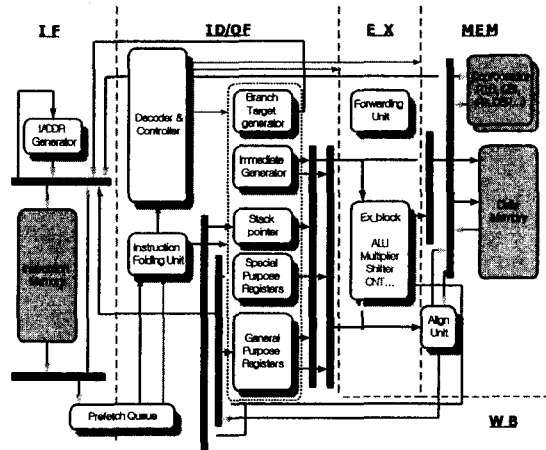


그림 1 AE32000의 블록 다이어그램

3. AE32000의 가변 목표형 설계 기법

설계 자동화 틀을 다양한 범위의 목표에 대해서 적용시킬 수 있을 때 ‘가변 목표형’이라 칭한다. 특히 본 논문에서 가변 목표형이라 함은 서로 다른 다양한 응용 분야에 적용시킬 수 있음을 뜻한다. 가변 목표형 설계는 여러 가지 상황의 발생 가능성으로 인해 특정 목표만을 위한 설계에 비해 매우 어렵다. 그러나 다양한 응용 분야로의 변경이 용이하고 최적화된 설계를 금방 얻을 수 있다는 장점으로 인해 가변 목표형 설계를 위한 노력은 꾸준히 지속되고 있다.

프로세서의 가변 목표형 설계에 있어서 사용되는 접근 방법은 크게 두 가지로 나눌 수 있다. 모든 명령어 수행이 지원되는 모델을 기준으로 하여 최적화된 명령어 집합을 수행하는데 있어 불필요한 블록들을 제거시켜 나가는 하향식(top-down) 접근 방식과, 기본적인 명령어 수행만이 지원되는 모델을 기준으로 최적화된 명령어 집합을 수행하기 위해 필요한 블록들을 추가시켜 나가는 상향식(bottom-up) 접근 방식이 있다.

본 논문에서 적용시킨 방법은 하향식 방식으로서 실험에 기준 모델로 사용된 AE32000은 DSP 명령어 군을 제외한 모든 명령어를 지원하는 프로세서로서, 지원하고 있는 명령어 군은 표 1과 같다.

표 1 기준 모델로 사용한 AE32000의 명령어 군

명령어 군	명령어 수
데이터 처리 및 기타 관련	38 개
분기(Branch) 관련	18 개
읽기(Load) 관련	15 개
저장(Store) 관련	6 개
보조 프로세서 관련	7 개
계	84 개

크게는 표 1에 나와 있는 각 명령어 군에 따라, 작게는 각 명령어군 내의 명령어 종류 및 레지스터 주소에 따라 기준 모델의 HDL 코드 상에서 사용되는 부분이 달라지게 된다. 따라서 HDL 코드 상에서 사용되는 명령어의 종류에 따른 연관성을 비교한 후 불필요한 부분을 삭제하고 문법에 맞는 HDL 코드를 재생성하는 설계 자동화 틀을 만들 수 있겠다. 목표 가변적인 HDL 코드 생성기를 만들기 위해서 본 논문에서는 다음과 같은 방법을 사용하였다. 데이터는 파라미터로서 정의되어 그 크기가 조정될 수 있도록 하였으며 사용되는 명령어의 종류에 따라 각 블록의 입, 출력 포트 및 그에 해당하는 코드 부분이 선택되어 출력되어 지도록 되어 있다. 전체적인 코드 생성 과정은 그림 2와 같다.

그림 2에서 보는 바와 같이 특정 응용 분야에 대한 소스 코드를 AE32000용의 컴파일러로 컴파일 시킨 후 BINUTILS를 이용해 binary 파일을 얻는다. 이후 얻어진 bin 파일을 이용해 프로파일링 과정과 HDL 코드 생성 과정을 거쳐 특정 목적에 맞게 재설계된 코드를 얻는다. 생성된 코드는 검증 과정을 통해 동작 상의 에러 여부와 최적도를 평가하게 된다.

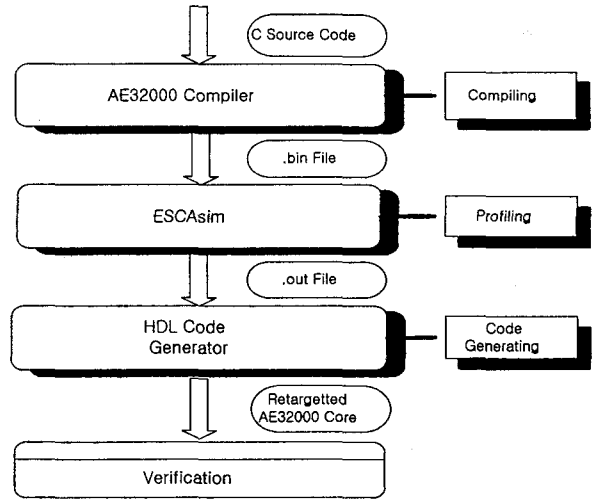


그림 2 목표 가변적인 설계 코드를 얻는 과정

3.1 프로파일링(Profiling)

소스 프로그램을 컴파일해서 bin 파일을 추출한 후, AE32000의 주기 정밀형(cycle-accurate) 시뮬레이터인 ESCAsim을 실행시켜 프로그램 수행 중 사용된 명령어와 사용되지 않은 명령어에 대한 정보를 얻는다. ESCAsim은 수행 기반(execution-driven) 방식의 시뮬레이터로서 주기 단위로 프로세서의 동작을 묘사하도록 되어 있다. 현재 설계되어 있는 ESCAsim은 주기 단위의 파이프 상태와 메모리에 적재/저장되는 값들에 대한 프로파일 데이터를 제공해 주지만 사용된 명령어의 종류만을 알기 위해선 별도의 작업이 필요하다. 우리가 원하는 프로파일 데이터를 직접적으로 얻기 위하여 ESCAsim의 디코더(decoder)부분에 각 명령어의 사용 유무를 체크한 후 시뮬레이션 종료 시 출력해 주는 기능을 추가하였다.

3.2 HDL 코드 생성기의 설계

HDL 코드 생성기는 C++ 언어로 설계하였으며 각 기능 유닛(functional unit) 및 제어 신호(control signal) 생성 모듈은 각각의 매크로(macro)로 정의하였다. 각 매크로 함수는 프로파일 데이터를 참조하여 알맞은 출력 코드를 선택해 주는 코드 선택부와 그에 해당하는 HDL 코드를 출력해주는 코드 출력부로 구성되어 있다.

HDL 코드 생성기의 동작과정은

- 1) ESCAsim에서 생성된 프로파일 데이터를 입력으로 받아 수행된 명령어의 종류, 수행되지 않은 명령어의 종류 및 부가 정보들을 기록하고,
- 2) 파라미터 정의부에서 1)의 정보를 참조하여 각 데이터의 파라미터들을 정의한다.
- 3) 함수 호출부에서 1), 2)에서 얻은 정보들을 참조하여 각 모듈에 대한 매크로를 순차적으로 호출한 뒤,
- 4) 해당 매크로에 정의되어 있는 바에 의해 Verilog 코드를 선택, 출력하도록 되어 있다.

AE32000 프로세서의 HDL 코드 상에서의 모듈은 크게

사용되는 명령어의 종류에 무관(independent)한 모듈과 사용되는 명령어의 종류에 따라 그 입력포트와 출력포트 및 동작 정의가 변경되는 모듈의 두 가지로 나눌 수 있다. 함수 호출부에서 이 두 가지의 분류에 의해 명령어들에 따른 의존성을 비교하게 되고 이에 해당하는 인수들(ex. 입출력 포트명, 입력 및 출력 포트의 개수, 신호의 비트폭 등)을 선택, 함수를 호출하는 역할을 담당한다. 명령어의 종류에 관계없이 존재해야 하는 모듈이나, 항상 사용되는 명령어들에 관련되는 모듈은 비교 절차를 거치지 않고 무조건 호출되어 출력되어 진다. 그림 3은 HDL 코드 생성기의 대략적인 구성도를 보여준다. 함수 호출부 외, 파라미터 정의부에서는 프로파일 데이터에 따른 프로세서의 구성요소, 주요 신호들의 비트폭 등을 정의하게 된다. 함수 호출부에 의해 호출되는 모듈들은 파이프라인 래치(latch), 각 파이프라인 단계에서의 기능 유닛 및 신호 생성 블록 등으로 이루어져 있다.

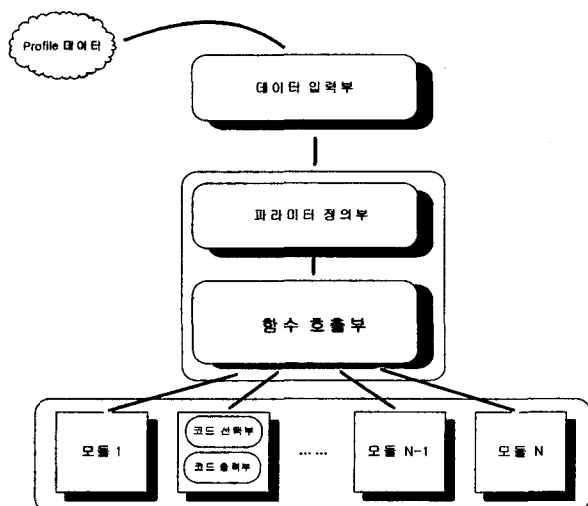


그림 3 HDL 코드 생성기의 구성도

4. 가변 목표형 설계 기법의 적용 예

디지털 카메라 등의 화상 입력 장치로부터 영상(image)을 입력받아 움직임의 변화를 감지하고, 움직임 발견 시 그 영상을 압축하여 저장하는 내장형 시스템에 AE32000 프로세서를 사용하였다. 이와 같은 내장형 시스템에서 AE32000 프로세서는 입력된 영상을 JPEG 알고리즘에 의해 압축한 뒤 이를 메모리에 저장하는 역할만을 담당하고 있다. 본 논문에서는 JPEG 프로그램의 수행에 최적화된 명령어 집합을 정한 뒤, 목표 가변형 설계 기법을 적용하여 AE32000 프로세서를 최적화 시키는 작업을 수행하였다. 표 2는 JPEG의 수행 시 사용되는 명령어 집합에 대해 ESCAsim의 프로파일 과정을 거친 결과를 보여주고 있다. 표 1의 기준 모델에서의 명령어 군과는 달리 JPEG 프로그램의 수행에 있어 사용되는 명령어 집합은 매우 간소화 되었음을 알 수 있다.

표 2 JPEG 프로그램 수행 시 사용되는 명령어 군

명령어 군	명령어 수
데이터 처리 및 기타 관련	16 개
분기(Branch) 관련	6 개
읽기(Load) 관련	9 개
저장(Store) 관련	4 개
보조 프로세서 관련	0 개
계	35 개

프로세서에서 사용되는 명령어의 종류에 따라 그 크기와 사용 유무가 영향을 받는 것은 주로 ID단의 디코더(decoder)와 EX단에 수행되는 기능 유닛들이다. 표 3은 기준 모델과 JPEG을 목표로 하는 모델의 합성 결과 중 위의 두 곳에서의 면적(area)을 보여준다. 합성은 Synopsys사의 Design Compiler와 삼성 0.35 μ m 공정 셀 라이브러리를 사용하여 수행하였다.

표 3 Decoder와 Functional Unit의 합성 결과

	기준 모델	JPEG 목표 모델
Decoder	1003.9	605.0
Functional Unit	13912.4	13591.9

(단위 : 동가 게이트)

JPEG 프로그램을 사용하는 경우, 기능 유닛들(ALU, multiplier, shifter 등) 중 적은 게이트 수를 가지는 CNT(counter) 유닛을 제외한 모든 기능 유닛을 사용하기 때문에 기능 유닛에서의 면적 이득은 적지만 전체적으로 1400여 게이트를 줄일 수 있었다.

5. 결론

본 논문에서는 AE32000 프로세서를 특정 응용 목적으로 사용하고자 할 경우의 최적화 방법에 대해 알아보았다. HDL 코드 생성기를 통하여 특정 응용에 최적화된 코드를 자동적으로 생성하게 하였으며 임의의 응용 분야라도 간단한 프로파일 과정만을 거치면 최적화된 설계를 바로 얻을 수 있도록 하였다. 내장형 프로세서 분야에서의 Time-to-Market의 중요성이 점점 증가하는 점을 고려할 때 이러한 작업을 통해 응용 분야에 적합한 설계를 손쉽게 얻을 수 있으며, 최적화된 설계를 얻음으로써 시스템의 비용을 절감할 수 있다는 장점을 지닌다.

감사의 글

본 연구는 한국반도체연구조합(COSAR)과 (주)에이디칩스 및 반도체설계교육센터(IDEC)의 지원을 받아 수행되었습니다.

참고 문헌

- [1] R. Camposano, J. Wilberg, Embedded System Design, Design Automation for Embedded Systems, an International Journal, Kluwer Academic Publishers, pp5-50, 1996.
- [2] H.-C. Oh, et al., "AE32000: An Embedded Micro-processor Core", Proc. of AP-ASIC, pp.255-258, Aug. 2000.
- [3] H. Lee, P. Beckett, B. Appelbe, High-performance extendable instruction set computing, Proc. of ACSAC, pp.89-94, Jan. 2001.