

## SystemC 방법론을 적용한 JPEG Encoder 설계

오세준, 조원경  
경희대학교 전자공학과  
전화 : (031) 201-2196

### Design of a JPEG Encoder Using SystemC Methodology

Se-Jun Oh, Won-kyung Cho  
Dept. of Electronic Engineering Kyung-Hee Univ.  
E-mail : ohsejun@vlsi.kyunghee.ac.kr

#### Abstract

In this paper we present the design of a JPEG Encoder using SystemC Methodology. Our methodology supports the efficient mapping of C/C++ functional descriptions directly into hardware. The use of C/C++ to model all parts of the system provides great flexibility and enables faster simulation compared to existing methodologies. The designer can estimate system performance and verify functional correctness of the designs using commonly available software compilers. A design flow in SystemC begins with an untyped description in C++, using a library of new data types useful for modeling hardware. The description can be compiled and simulated for functional correctness. Then, the design may be refined by adding interface specification and timing information, and again the typed description can be compiled with a standard C++ compiler, simulated, and debugged.

#### 1. 서론

최근 반도체 공정 기술의 발달로 기존의 ASIC 설계 방식이 하나의 칩에 시스템을 구현하는 개념의 system-on-chip 설계방식으로 일반화 되어가는 추세이다. 면적 단위의 최적화 요구로부터 빠른 설계 기간을 요구하는 방향으로 변함에 따라 re\_usable design, IP based design 이라고 하는 새로운 개념의 설계방법이 강조되고 있다.

복잡한 시스템 설계 절차 초기에 정확한 설계 사양은

설계 기간의 가장 큰 부분을 차지하는 검증(verification)과 디버깅(Debugging) 과정을 줄여 복잡한 알고리즘을 빠른 시간 내에 설계할 수 있게 한다. C/C++언어로 작성된 하드웨어 모델은 실행형 설계 사양이 되므로 하드웨어 설계 시 별도의 테스트 환경을 마련할 필요가 없게 된다. 이런 점 때문에 많은 하드웨어, 소프트웨어 설계자들은 executable specification을 만들려고 하고 있다. executable specification은 실행되었을 때 시스템과 같은 동작을 표현하는 본질적인 C/C++ 프로그램이다.

C/C++는 간결하고 효과적인 시스템 기술수단이므로 소프트웨어 알고리즘이나 인터페이스를 작성하는데 적당한 언어로 사용되었으며 이미 많은 설계자가 이 언어에 익숙할 뿐만 아니라 많은 개발환경이 구비되어 있기 때문에 복잡한 시스템의 기술 방법으로 C/C++언어를 이용하는 것이 필요함을 설계자들은 느끼고 있다. 그러나 시스템을 기술하는데 필요한 병렬성(concurrency), 입출력 신호의 지정(input-output mode), 이벤트 처리(signal event processing) 등의 요소들이 부족하다.

최근에 발표된 systemC는 시스템 설계용으로 만들어진 표준 C++언어 라이브러리로서, C++언어의 표준 객체형뿐만 아니라 HDL(Hardware Description Language)의 MVL(Multi-Valued Logic)과 같은 RTL 수준의 하드웨어 적인 객체형들을 지원하기 위한 다양한 클래스(class)를 제공하여 최상위 추상화 수준인 시스템 모델링으로부터 낮은 수준의 RTL 모델링이 가능하다. [1]

본 논문에서는 C/C++를 기반으로하는 SystemC 방법론을 통해서 하드웨어 구조에 맞는 알고리즘 개발 및 성능 예측, 테스트와 검증의 용이함을 보여 복잡한 알고리즘을 빠른 시간 내에 설계할 수 있음을 고찰해 보고자 한다.

## 2. 본문

### 2.1 SystemC 디자인 방법론

기존의 설계 방법론은 개념과 알고리즘을 검증하기 위해서 C/C++로 기술하는 것부터 시작한다. 개념과 알고리즘의 검증이 끝난 다음 C/C++모델은 실제 하드웨어로 구현되기 위해서 VHDL이나 Verilog로 바꾸어 기술되었다. 그림1은 기존의 설계 방법을 보여준다.

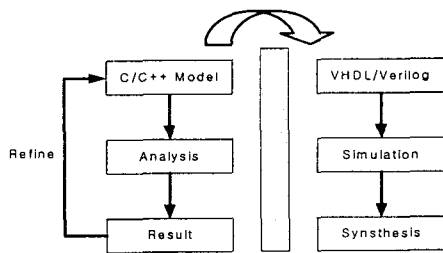


그림 1. 기존의 설계 방법

Fig. Current Design Methodology

이러한 접근 방식에는 몇 가지 문제점이 있다.

- 1) C/C++알고리즘을 한번에 타이밍 정보와 하드웨어 구조를 표현하는 HDL로 변환하는 작업은 어렵고 검증하는데 시간이 오래 걸린다.
- 2) C/C++ 모델은 단순히 Reference 로 제공되고 HDL로 변환된 다음에 HDL모델은 개발의 초점이 된다.
- 3) 대량의 테스트 벡터를 이용한 검증이 어렵다.
- 4) 가장 큰 문제는 하드웨어를 고려한 설계를 할 수 없다는 것이다.

하위 단계에서부터 설계하는 방법은 구현과 검증에 시간과 노력이 많이 소요되며 미처 검증되지 못하는 부분은 오류 가능성이 높아진다. 따라서 설계의 추상화 단계를 높이는 것이 매우 중요하다고 할 수 있다.

설계 과정에서 상위 단계는 하위 단계로 구현상의 요구 사항을 제공하고 하위단계는 이것을 바탕으로 하드웨어 구현을 진행한다. 그리고 구현 결과는 상위 단계로 전달함으로써 구현의 정확도 검증이나 오류 발생 시 상위 모델의 재설계를 유도하게 된다.[4]

SystemC 설계 접근 방식은 설계의 신뢰도를 결정하는 테스트 및 검증의 자동화에 큰 기여를 할 수 있다는 점과 설계 전 단계에서 하드웨어 구조에 맞는 알고리즘의 개발 및 시스템 예측, 그리고 시뮬레이션 환경의 구축이라는 측면에서 매우 큰 의미가 있다. [5]

### 2.2 C/C++ 기반의 JPEG Encoder 설계

본 논문에서는 systemC 디자인 방법론을 적용하여 JPEG Encoder 설계에 적용해 보았다. 그림 2 는 C/C++ 언어 기반의 시스템 설계 과정을 보여준다.

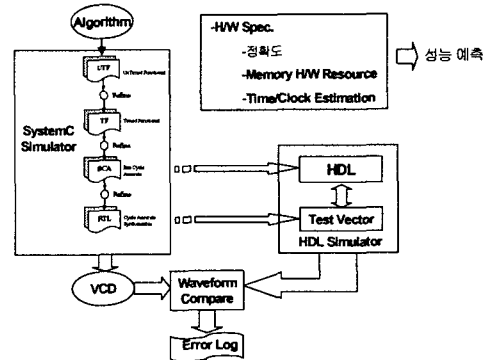


그림 2 설계 과정

Fig. Design Flow

#### 2.2.1 Untimed Functional Level(UTF)

구체적인 타이밍 정보와 하드웨어 구조의 표현 없이 순수한 기능적 모델을 말한다. 각 이벤트의 논리적 진후 관계만 존재하는 순차적 형태이다. 그러므로 시뮬레이션 되었을 때 0 시간이 걸리게 된다.[1] 그림 4 는 JPEG Encoder의 UTF 모델을 보여준다. 여기서 TestBench 는 JPEG Encoder를 Fixed-point C 모델로 구현하였으며 SystemC 모델과 C 모델과의 출력을 비교하여 JPEG Encoder 기능을 검증한다. H.261 권고안의 난수 발생 방법에 의거하여 테스트 벡터를 생성 후 이를 C 모델과 SystemC 모델에 적용하여 그 출력 결과를 비교함으로써 설계 검증을 할 수 있게 했다.

#### 2.2.3 Timed Functional Level(TF)

UTF 형태로 분할된 각 알고리즘은 하드웨어를 감안하여 개발되며 처리 시간 예측이라는 시간 개념이 들어간다.[1] 이 단계의 모델링을 TF 라하여 시스템의 전반적인 스케줄링(Scheduling)이 가능해진다. 여기서 시간적 개념은 오직 delta 시간의 지속을 표현하며 클럭(Clock)이 아니다. 그림 3은 JPEG Encoder의 UTF모델을 보여준다.

## SystemC 방법론을 적용한 JPEG Encoder 설계

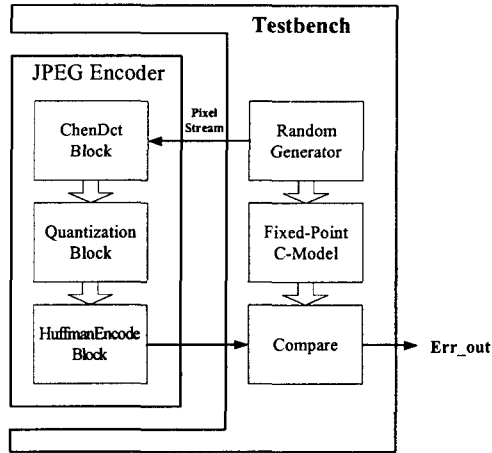


그림 3 JPEG Encoder 의 UTF 모델  
Fig. UTF model of the JPEG Encoder

이 모델은 알고리즘을 하드웨어 구조에 맞게 설계해야 하기 때문에 다음과 같은 설계 사양을 감안하여 모델링했다. DCT 블록은 Signed Digit 부호를 사용한 Hardwired Multiplier를 사용하여 연산기를 설계한다. Cosine 계수들은 짝수 행과 홀수 행으로 분리하면 규칙적으로 배열되고 7개의 계수를 이용하여 4x4 행렬로 구성된다. DCT 연산을 위한 각 Cosine 계수는 4~6 bit의 Signed Digit로 표현되며 입력 데이터와 Cosine 계수의 승산은 가산으로 처리되므로 그 구조에 적합한 Carry Save Adder를 사용한다. 양자화 블록은 DCT 연산의 결과로 얻어진 계수를 양자화 행렬과 양자화 Scale 값에 의한 제산 과정을 수행하는 단계로서 하드웨어 구조에 적합한 방법으로 승산을 수행한 후 bit shift operation을 수행함으로써 구하고자 하는 양자화 값을 구하게 된다. 즉 양자화 행렬 값들과 양자화 Scale 값들에 대한 역수를 구한 후 12bit scale-up 한 값을 구하여 DCT 연산 결과 출력 값과 승산을 수행한 후 오른쪽으로 24bit shift함으로써 양자화된 계수들을 구할 수 있다. Zigzag 스캔은 어드레스에 의하여 양자화된 데이터의 순서를 결정한다. 또한 연속적으로 입력되는 양자화 데이터를 재배열 하기위해 메모리가 필요하다.

### 2.2.4 Bus Cycle Accurate Level(BCA)

BCA 모델은 기능 블록들간의 통합 시뮬레이션이 가능하도록 개발하며 점진적으로 CA 수준의 세부 설계를 수행한다. 분할 전 각 블록들간의 인터페이스를 고려하여 설계의 재사용성(Design reusability)을 높이는 것이 중요하다. UTF, TF 수준의 Abstract 포트는 모듈간의 인터페이스를 위해 데이터, 주소, 제어를 가지는 버스 포

로토크로 정제(Refinement)된다. 본 논문은 enable-handshake 프로토콜을 사용했다.

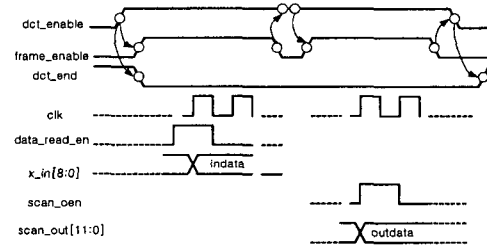


그림 4. 입출력 포트 인터페이스  
Fig. 4 In/Out Port Interface

### 2.2.5 Cycle Accurate Synthesizable Level(CA)

시스템의 동작이 클럭(clock)에 동기화되어 작동하게 되는 RTL모델이다. 여기서 얻은 타이밍 결과를 가지고 전체적인 시스템을 검증할 수 있다. 이 모델은 모든 설계 사양을 Timed 하게 구현했으므로 HDL로 변환 과정이 쉬우며 같은 타이밍 결과를 얻을 수 있다. 그림 4는 CA 수준의 JPEG Encoder의 입출력 타이밍을 보여준다. 그림 5는 DCT 블록과 HuffmanEncode 블록에서 데이터 처리를 위한 메모리 쓰기와 읽기 타이밍이다. 여기서 클럭의 상승에지와 me 신호의 상승 에지에서 메모리가 작동하게 된다.

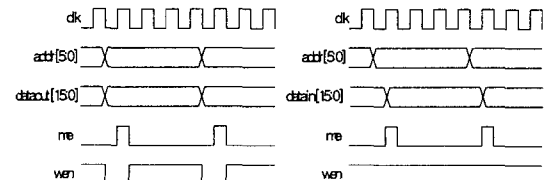


그림 5. 메모리 쓰기와 읽기 타이밍  
Fig. Memory write and read timing

## 3. 실험 결과

H.261 권고안의 난수 발생 방법에 의하여 입력 테스트 벡터를 생성하였다. 난수 범위는 5~-5, 255~-256, 300~-300에 대하여 각각 검증하였다.

표 1은 각 Unit들의 enable 신호 후 지연되는 클럭 수를 나타내었다. Encoder 입력 enable 신호 후 6개 블록에 대한 DCT/Quantization/HuffmanEncode에 소요되는 시간은 DCT 지연 + Quantization 지연 + HuffmanEncode 지연 + (64×6 Block)=142 + 4 + 68 +

(64×6) = 478 clock 이다.

Unit	지연(clock)
DCT	142
Quantization	4
HuffmanEncode	68

표 1. 각 유닛에 대한 초기 지연

Table 1. Initial latency for each Unit

#### 4. 결론

본 논문에서는 C++언어 기반의 SystemC 방법론을 이용하여 JPEG Encoder을 설계하였다. 분할된 알고리즘은 하드웨어 설계에 적합하도록 개발했다. 하드웨어에 적합하게 개발된 알고리즘과 입출력 타이밍 결과를 토대로 세부 설계했다.

복잡한 알고리즘의 기능 모델을 고도의 추상화 수준에서 하드웨어 구조에 적합하게 빠른 모델링이 가능하며 C/C++ 언어 환경에서 작성된 Executable Spec. 으로부터 표준화된 VCD 형식을 따르는 테스트 벡터를 이용하여 HDL로 설계한 RTL 모델이 정확하게 동작하는 것을 쉽게 검증할 수 있다. 즉 설계의 신뢰도를 결정하는 테스트 및 검증이 쉽다. 다시 말해서 전통적인 설계 방법보다 더 정확하고 빠른 시뮬레이션을 통하여 설계 기간을 단축하는데 기여함을 실제 설계 예를 통하여 고찰하였다.

향후 과제로는 float-point 알고리즘을 쉽고 빠르게 fixed-point로 모델링 할 수 있는 SystemC Fixed-point 데이터형을 사용한 방법과 SystemC 모델과 VHDL 모델간의 통합 시뮬레이션(Co-Simulation)을 통한 검증 방법이 연구되어야 하겠다.

#### 참고문헌

- [1] synopsys, Coware, "SystemC User's Guide ver1.0.2" 2001
- [2] Lukai Cai, Junyu Peng, Chun Chang, Design of a JPEG Encoding System, SpecC Technical report
- [3] Joachim Gerlach, Wolfgang Rosenstiel, "System Level Design Using the systemC Modeling Platform", SystemC Technical Paper, 2000.
- [4] Michael Keating, Pierre Bricaud, "reuse Methodology Manual For System-on-Chip

Designs", Kluwer Academic Publishers, 1998.

- [5] Stan Liao, Steve Tjiang, Rajesh Gupta, "An Efficient Implementation of Reactivity for Modeling Hardware in the SCENIC Design Environment", proceedings of the Design Automation Conference DAC'97, pp.70-75, June 97.
- [6] Giovanni De Micheli, "hardware Synthesis from C/C++ Models", pro-ceedings of the DATE'99 conference, pp.382-383, March 99.