

SoC를 위한 JPEG2000 IP 설계 및 구현

정재형*, 한상균*, 홍성훈**, 김영철**

*전남대학교 전자공학과, **전남대학교 정보통신공학부

JPEG2000 IP Design and Implementation for SoC Design

Jae-Hyung Jung*, Sang-Kyun Han*, Sung-Hoon Hong**, Young-Chul Kim**

Dept. Electronic Engineering, Chonnam National University,

Dept. Electronic & Computer Engineering, Chonnam National University

E-mail: rebite@vip.chonnam.ac.kr, hsh@chonnam.ac.kr

요약

JPEG2000은 기존의 정지영상압축부호화 방식에 비해 우수한 비트율-왜곡(Rate-Distortion)특성과 향상된 주관적 화질을 제공하며 인터넷, 디지털 영상카메라, 이동단말기, 의학영상 등 다양한 분야에서 적용될 수 있는 새로운 정지영상압축 표준이다. 본 논문에서는 SoC(System on a Chip)설계를 고려한 JPEG2000 인코더의 구조를 제안하고 IP(Intellectual Property)를 설계 및 검증하였다. 구현된 JPEG2000 IP는 DWT(Discrete Wavelet Transform)블록, 스칼라양자화블록, EBCOT(Embedded Block Coding with Optimized Truncation)블록으로 구성되어 있다. IP는 모의실험을 통해 구현 구조에 대한 타당성을 검증하였고, 반도체설계자산업 연구센터에서 제시한 'RTL Coding Guideline'에 따라 HDL을 설계하였다. 특히, DWT블록은 구현시 많은 연산과 메모리 용량이 필요하므로 영상을 저장할 외부 메모리를 사용하였고, 빠른 곱셈과 덧셈연산을 위한 3단 파이프라인 부스곱셈기(3-state pipeline booth multiplier)와 캐리예측 덧셈기(carry lookahead adder)를 사용하였다. 설계된 JPEG2000 IP들은 삼성 0.35 μ m 라이브러리를 이용하여 Synopsys사 Design Analyzer 툴을 통해 논리합성하였으며, Xilinx 100만 게이트 FPGA칩에 구현하여 그 동작을 검증하였다. 또한, Hard IP 설계를 위해 Avanti사의 Apollo툴을 이용하여 Layout을 수행하였다.

1. 서론

VLSI 기술의 발전으로 보다 많은 양의 로직이 단일 칩에 집적 가능하게 되었고, 이는 SoC시대의 도래를 가능하게 했다. SoC^[3]를 가능하게 하기 위해서는 많은 종류의 IP가 필요하고, 공정 변환을 쉽게 하기 위

해서는 합성이 가능한 RTL 설계가 절실히 요구된다.

본 논문에서는 새로운 정지영상 압축표준인 JPEG2000을 SoC설계에 적용하기 위해 IP블록들을 'RTL Coding Guideline'에 맞게 설계하고, 설계된 IP를 FPGA칩으로 구현하여 동작을 검증하였으며, 삼성 0.35 μ m 라이브러리를 사용하여 합성 및 Layout을 수행하였다. 또한, 설계된 IP가 얼마나 IP의 표준에 맞게 설계되었는지 TransEDA사에서 제공하는 LINT tool을 이용하여 검증하였다.

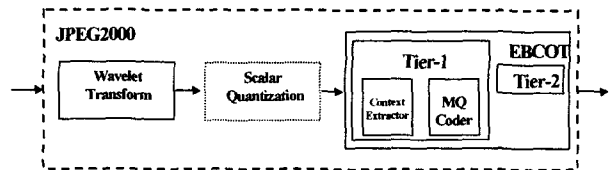


그림 1. JPEG2000 블록 다이어그램

그림 1은 JPEG2000^[1] 블록 다이어그램을 나타낸 것으로, JPEG2000은 크게 저주파 성분과 고주파 성분을 분리하여 부밴드(subband)를 형성하는 DWT블록, 부밴드별로 DWT계수에 대해 양자화를 수행하는 양자화블록, 양자화된 DWT계수값을 비트플랜 단위로 나눈 후 인접한 계수간의 상태정보를 이용하여 중복성을 줄이는 EBCOT블록으로 구성되어진다. EBCOT^[2]는 크게 Tier1과 Tier2로 구성되어지는데, Tier1은 컨텍스트 기반의 산술부호화에 의해 실제로 데이터를 압축하는 부분이고, Tier2는 압축된 데이터를 패킷(packet) 단위로 묶어 전송하는 부분이다.

그림 2는 본 논문에 기술된 IP 설계과정을 나타낸 것으로서, IP들은 하드웨어 구현 시 높은 성능 향상을 얻을 수 있도록 IP의 기본 구조와 알고리즘 분석 후 분석된 알고리즘의 상위 수준의 동작을 검증하기 위해 C-모델링 및 HDL 동작적모델링(Behavioral Modeling)을 통해 설계하였다. 또한, 모델링된 블록들은 'RTL Coding Guideline' 및 '합성 Guideline'에 따

※ 본 논문은 “전남대 IDEC 반도체 설계 툴” 지원과 “반도체 설계자산업연구센터의 연구비 지원”에 의해 연구되었음

라 HDL RTL 코딩과 시뮬레이션을 통해 동작 검증하였으며, HDL RTL Coding으로 검증된 IP들은 FPGA 검증 작업을 하였다. FPGA 검증된 IP는 삼성 0.35 μ m CMOS Phantom Cell 라이브러리를 이용하여 논리 합성하였고, Place & Route를 거쳐 Hard IP로 개발하였다. 이러한 과정에서 SoC 설계사양을 결정하여 IP를 통합 검증하였다.

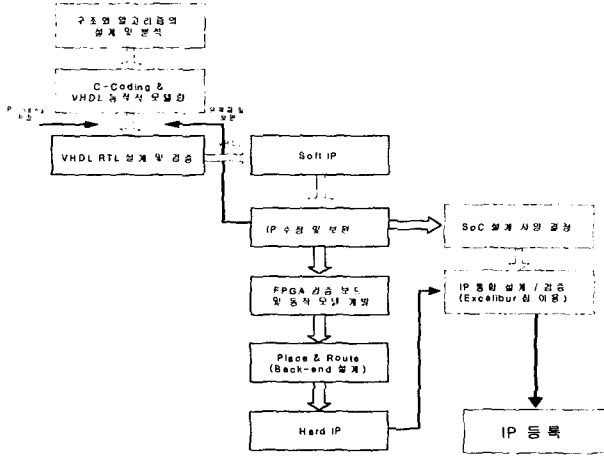


그림 2. JPEG2000 IP 설계과정

본 논문의 구성은 2절에서 DWT IP, 3절에서 양자화 IP, 4절에서는 EBCOT IP에 대한 각 IP의 구조 및 모의실험결과를 기술하였으며, 5절에서는 H/W 실험결과를 보이고, 6절에서 결론 및 향후 연구 계획에 대해 언급한다.

2. DWT IP 구조 및 모의실험

DWT IP는 그림 3에 나타낸 바와 같이 크게 점대칭 확장, DWT 필터링, 데이터 재정렬의 3개 블록으로 구성되어 있다. 설계된 DWT IP는 1차원 구조의 DWT를 설계하였으며 2차원의 1-Level DWT된 영상 결과를 얻기 위하여 1차원 구조의 DWT를 수평방향으로 적용 후 수직 방향으로 수행함으로써 2차원 DWT 출력결과를 얻을 수 있도록 하였다.

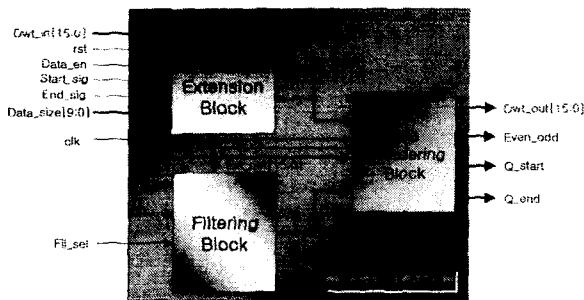


그림 3. DWT IP 블록 다이어그램

2.1 점대칭 확장 블록

실제 큰 이미지에 대해 Wavelet 변환을 수행하면 많은 양의 메모리가 필요하게 된다. 만약 사용할 수 있는 메모리의 양이 제한되어 있다면 입력 이미지를 블록으로 쪼갠 다음 각 블록에 대해 독립적으로 Wavelet변환을 수행하게 된다. SSWT⁽⁴⁾(Spatially Segmented Wavelet Transform) 방법은 이러한 목적을 수행하기 위해 제안되었다. 그러나 오버래핑(overlapping)하지 않은 Wavelet변환이 하나의 이미지 압축 시스템으로 통합되었을 때, 이미지의 왜곡현상(Boundary Artifact)을 피할 수 없게 된다. 이러한 왜곡현상은 특히 전체 이미지를 작은 타일(Tile)로 쪼갠 경계부분들에서 가장 두드러지게 나타나게 된다. 이러한 단점을 극복하기 위해 각 블록의 경계부분에서 점대칭 확장 방법을 사용하게된다. 그림 4는 점대칭 확장의 예를 나타낸 것으로, 확장된 샘플 $x[i]$ (점선)은 대칭점을 통과하는 화살표에 의해 원래의 입력신호 $X[i]$ (실선)로부터 만들어진다. 확장된 샘플 $x[n]$ 을 계산하기 위해서 입력신호 $X[n]$ 의 처음과 끝에서 식 1의 공식이 사용된다.

$$x[i] = 2 \times P - X[i] \quad (1)$$

여기에서 P는 픽셀 시작점과 끝점의 화소값을 의미하며, $X[i]$ 는 i번째 입력영상의 화소값을 나타내고, $x[i]$ 는 점대칭 확장된 화소값을 나타낸다.

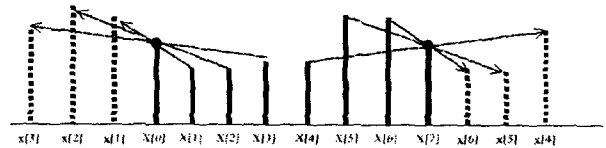


그림 4. 점대칭 확장의 예

2.2 DWT 필터링 블록

DWT 필터링은 임펄스 응답(impulse response) $h[n]$ 을 갖는 half band 저역필터와 임펄스 응답 $g[n]$ 을 갖는 half band 고역필터에 입력신호 $x[n]$ 을 통과시키는 과정이다. 구현된 DWT 필터링 블록은 식(2),(3)과 같은 처리를 수행하도록 설계하였다. 또한, DWT 필터링 블록은 Daubechies 9/7 필터계수와 5/3 필터계수 두 가지를 두어 사용자가 선택할 수 있도록 하였다.

$$y_{high}[k] = \sum_n x[n] \cdot g[2k-n] \quad (2)$$

$$y_{low}[k] = \sum_n x[n] \cdot h[2k-n] \quad (3)$$

여기서, $y_{high}[k]$ 는 고역 필터링 후 다운샘플링을 한 출력이고, $y_{low}[k]$ 는 저역 필터링 후 다운 샘플링을 한 출력이다.

구현된 1차원 DWT필터 블록은 전치형 (Transposed form)을 사용함으로써 고속연산을 수행할 경우 덧셈기의 출력을 저장하기 위한 레지스터의 수를 줄여 수행성능을 높였다. 그림 5는 전치형의 필터 구조를 나타낸 것으로 필터계수와 입력 픽셀값을 곱해주는 곱셈 연산으로는 3단 파이프라인된 부스곱셈기를 이용하였고, 덧셈 연산으로는 캐리예측 덧셈기를 이용하였으며, 내부 연산의 형태는 2의 보수 형태로 연산을 수행하도록 하였다.

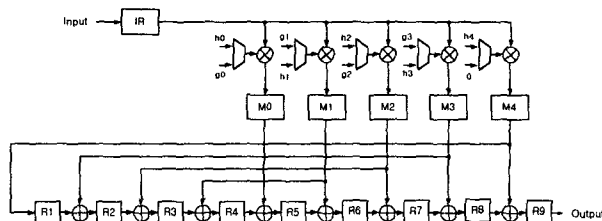


그림 5. DWT 필터 블록도

2.3 데이터 재정렬 블록

DWT는 홀수와 짝수개의 입력 시퀀스에 대해 각각 다른 동작을 한다. 홀수개의 입력 시퀀스의 경우, 필터링 블록에서 데이터의 재정렬을 수행하지 않아도 역 DWT를 하여 원래의 입력 샘플값을 그대로 복원할 수 있지만, 짝수개의 입력 시퀀스에 대해서는 필터 뱅크의 완전 복원 특성을 보존하기 위해서 데이터 재정렬이 요구된다. 그림 6은 데이터 재정렬에 대한 블록도를 나타낸 것으로, 데이터 재정렬은 한 라인의 마지막 저주파 필터링된 픽셀값과 고주파 필터링된 픽셀값에 대해서만 연산을 수행하므로 그림과 같은 구조로 구성되게 된다.

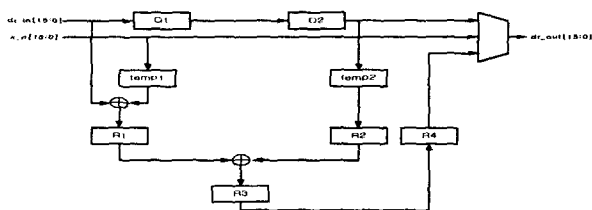


그림 6. 데이터 재정렬 블록도

2.4 DWT IP 모의실험

C-프로그램은 HDL의 동작적모델링과 비슷한 수준에서 수행을 하였으며, HDL 코딩 전 I/O 비트 및 필터 계수 비트등을 변수로 하여 최적의 PSNR을 가지는 비트를 결정후 HDL에 적용하였다. 그림 7은 16bits I/O에 대해 필터계수 비트수에 따른 영상이미지의 PSNR값을 측정하여 나타낸 결과이다. 16비트의 I/O와 필터계수가 14비트 일 때 가장 좋은 PSNR값(60~70dB)을 나타내었다.

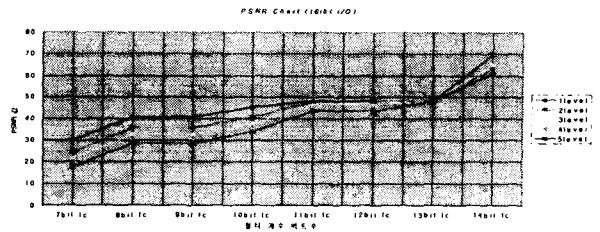


그림 7. 영상 PSNR 측정 결과

그림 8은 C-모델링을 이용한 영상 실험 결과로서, 256×256 "Lena"영상을 3-Level까지 DWT를 수행한 결과를 보여주고 있다.



(a) 입력 영상 (b) 3-Level (c) 복원 영상

그림 8. C-Coding을 이용한 영상 시뮬레이션

3. 양자화 IP 구조 및 모의실험

JPEG2000에서의 양자화^[5]는 DWT변환된 대역별 영상정보를 중요도에 따라 다듬고 정리하는 부분으로서, 일반적으로, 저주파 대역에 대해서는 양자화 간격을 적게하고 고주파 대역에 대해서는 양자화 간격을 크게하여 영상의 화질과 압축률을 조절한다. 그림 9는 양자화 IP 전체 블록도를 나타낸으로 대역별로 양자화 간격을 임의로 조절할 수 있도록 설계하였다.

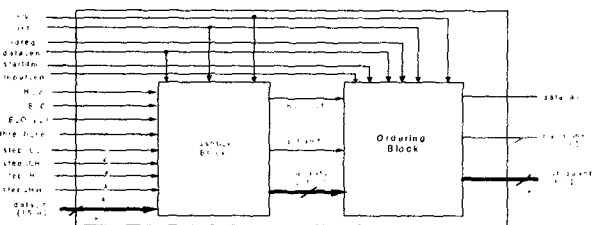


그림 9. 양자화 IP 블록 다이어그램

3.1 양자화 블록

JPEG2000 표준에서 제시하는 양자화 방식은 SQ (Scalar Quantization)를 기본 방식으로 사용한다. 구현한 양자화 IP에서는 SQ를 적용하였는데, 입력 DWT 계수 $x_i[n]$ 에 대해 식 (4)와 같이 독립적으로 양자화 처리하여 출력 계수 $q_i[n]$ 를 얻는다.

$$q_i[n] = \text{sgn}(x_i[n]) \cdot \left\lfloor \frac{|x_i[n]|}{\Delta b} \right\rfloor \quad (4)$$

여기서 $\lfloor \cdot \rfloor$ 은 절단(truncation)이고, 양자화 간격으로서

4b는 부대역에 따라 임의로 설정할 수 있다.

그림 9의 양자화 블록은 부대역별로 사용자에게 의해 정해지는 양자화간격에 의해 DWT계수를 나누는 연산을 수행하고, 밴드정렬 블록은 양자화된 DWT 계수를 SDRAM에 저장할 때 연속적인 쓰기(Burst write)처리에 의한 메모리 접근시간을 줄이기 위해 각 부대역별로 연속적인 출력을 하게 하였다.

3.2 양자화 IP 모의실험

그림 10은 C-모델링을 이용한 입력 영상 이미지에 대한 양자화 결과를 레벨에 따라 보여주고 있다. 각각의 밴드에 따라 사용자가 양자화간격을 할당하여 입력 데이터 값을 나눈 후 복원한 영상이다.



(a) 입력 영상 (b) 1-Level (c) 3-Level (d) 복원영상

그림 10. C-Coding을 이용한 영상 시뮬레이션

4. EBCOT IP 구조 및 모의실험

EBCOT Tier1은 각 부 밴드별로 압축부호화가 수행되는 부분으로서 양자화된 입력 DWT 계수값이 저장된 SDRAM에서 부밴드를 64×64이하의 크기를 갖는 code-block단위로 입력받는다. 양자화된 DWT 계수는 MSB에서 LSB까지 비트플레인(bit-plane)으로 나누어 부호화된다.

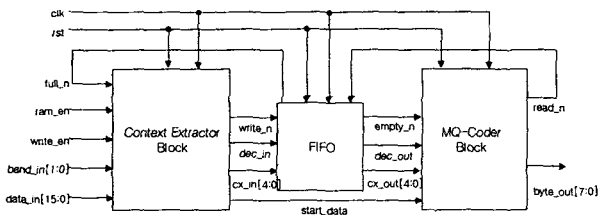


그림 11. EBCOT IP 블록 다이어그램

그림 11은 EBCOT IP 블록의 구조로서 컨텍스트 추출부, FIFO, MQ-Coder 3개의 서브블록으로 구성하였다. 컨텍스트 추출부에서는 입력된 데이터의 특징에 따라 결정값 및 컨텍스트를 출력시키고, MQ-Coder에서 결정값과 컨텍스트를 입력으로 받아 컨텍스트에 의해 확률값이 선택되고, 결정값에 의해 압축된 데이터가 출력된다. 또한 컨텍스트 추출부와 MQ-Coder블록간 데이터 입출력 동기를 위해 FIFO를 설계하였다.

4.1 컨텍스트 추출부 블록

입력된 코드블록을 비트플랜으로 나눈 후, 각 비트플랜의 픽셀들을 열(row)방향으로 스캔한 후 각 샘플에 대해 3가지 패스(pass)별로 컨텍스트를 추출하고 산술부호화(MQ-Coder)를 수행한다.

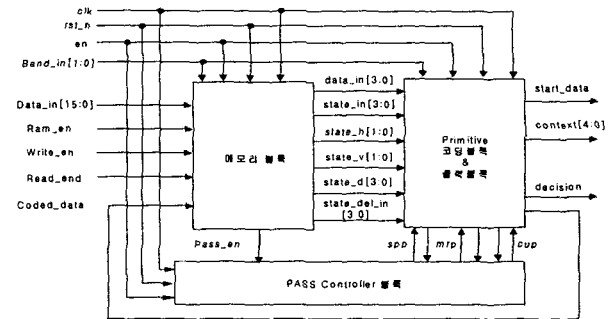
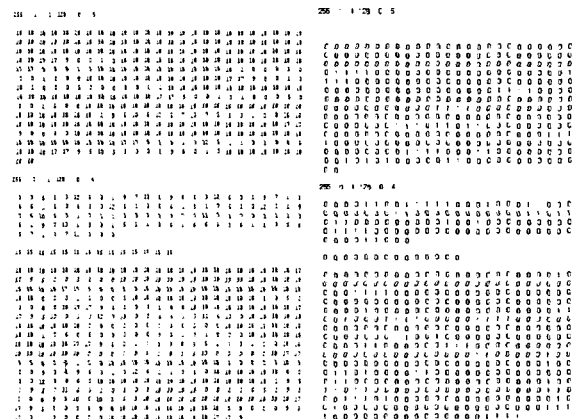


그림 12. 컨텍스트 추출부 블록 다이어그램

그림 12는 컨텍스트 추출부 구조를 나타낸 것으로 크게 메모리, 패스 컨트롤러, Primitive 코딩으로 나누어진다. 그림12의 메모리에서는 데이터를 열 방향으로 4개씩 스캔하면서 데이터 및 상태정보를 출력하며, 패스컨트롤러에서 샘플의 상태 정보에 따라 선택신호에 의해 3가지 패스 중 하나의 패스가 결정된다. 패스1은 "Significant Propagation Pass"로서 하나 이상의 유효한 이웃을 가진 상태들은 이 패스에서 부호화되어지고, 패스2는 "Magnitude Refinement Pass"로서 상위 비트플랜의 유효픽셀(significant pixel)들은 이 패스에서 부호화되어진다. 패스3은 "Clean Up Pass"로서 처음 두 개의 패스에서 부호화되지 않은 픽셀들이 이 패스에서 부호화되어진다. 패스가 결정되면, Primitive 블록에서는 컨텍스트를 추출하기 위해 4가지 코딩 방식(Zero Coding, Run-Length Coding, Sign Coding, Magnitude Refinement Coding)을 적용하였다.

4.2 컨텍스트 추출부 블록 모의실험



(a) 패스별 컨텍스트 (b) 패스별 결정값

그림 13. 패스별 컨텍스트 및 결정값

그림 13은 1 Level DWT변환된 256×256 "Pepper"영상을 입력받아 출력된 패스별 컨텍스트 추출 결과와 결정값을 보여주었고 있다. 그림 14는 출력된 결정값을 입력으로 받아 생성된 복원영상으로 입력영상과 동일함을 확인함으로써, 알고리즘 및 구현구조의 타당성을 검증하였다.

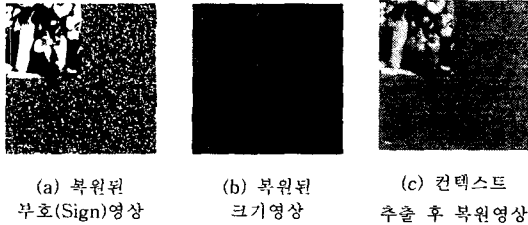


그림 14. 컨텍스트 추출 후 복원영상

4.3 MQ-Coder 블록

MQ-Coder는 크게 확률예측, 구간갱신, byteout 및 컨트롤러 4개의 서브블록으로 구성하였다. 그림 15는 MQ-Coder의 내부 구성을 나타내었다. "TEMPC"은 FLUSH과정으로 MQ-Coder를 끝내는 역할을 하며, "COUNTER"은 내부의 카운터값이 0이 되면 8비트의 압축된 결과가 출력되도록 카운터값을 제어하는 역할을 한다.

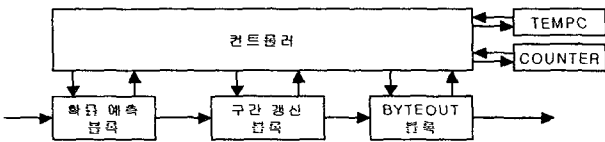


그림 15. MQ-Coder 블록 다이어그램

5. H/W 실험결과

그림 16은 VHDL로 작성한 RTL 코드를 검증하기 위해 수행한 영상 시뮬레이션 절차를 보여준다 실제 영상을 C-프로그램에 의해 영상 데이터 파일로 변환하고, 이 데이터 파일을 입력으로 하여 로직 및 타이밍 시뮬레이션을 수행한다. 시뮬레이션 결과에 의해 얻은 출력 영상 데이터 파일을 다시 C-프로그램을 이용하여 영상으로 변환한 후, 알고리즘 결정에 사용한 C-모델링의 출력결과와 비교함으로써 RTL 코드의 정확성을 검증하였다.



그림 16. 영상 시뮬레이션 절차

그림 17은 2차원 DWT에 대한 영상 시뮬레이션의 예를 나타낸 것으로, 수평 DWT의 경우, 8비트의 입력

영상을 C-프로그램을 통해 16비트로 확장한 후, VHDL의 테스트 벤치 파일에서 텍스트 파일 형태로 DWT블록의 입력값을 받아들이며 시뮬레이션을 수행하고, 수직 DWT의 경우, 영상을 90° 회전시킨 후 DWT블록의 입력값을 받아들이며 시뮬레이션을 수행한 결과 1-Level 2D-DWT된 영상을 얻을 수 있었다. 그림 18은 DWT IP 및 양자화 IP에 대한 VHDL 영상 시뮬레이션 결과를 나타내었다.

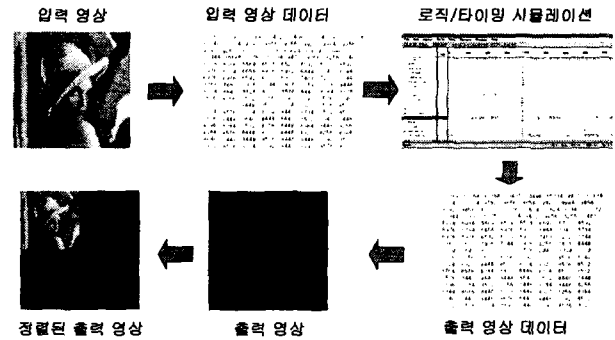


그림 17. 영상 시뮬레이션의 예



그림 18. 영상 데이터 시뮬레이션 결과

그림 19와 20은 22MHz에서 동작하는 Xilinx 100만 게이트 FPGA칩에 다운로드하여 로직 디버거를 이용한 시뮬레이션 결과의 일부분으로서, 출력값이 동작적 시뮬레이션 결과와 똑같이 출력됨을 확인할 수 있었다.

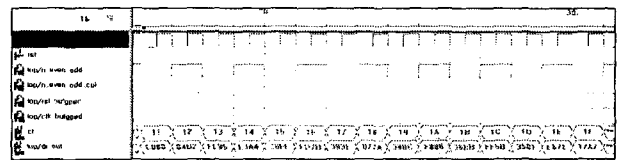


그림 19. DWT IP FPGA 시뮬레이션 결과

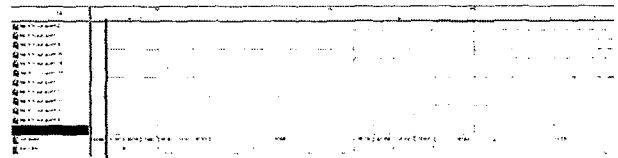


그림 20. 양자화 IP FPGA 시뮬레이션 결과

그림 21과 22는 컨텍스트 추출부와 MQ-Coder RTL 시뮬레이션 결과로서 256×256 1 Level 양자화된 DWT "Pepper"영상을 입력받아 출력된 합성 전 시뮬

레이션 결과를 보여 준다. 각 블록의 출력값은 동작적 시뮬레이션 결과와 똑같이 출력됨을 확인할 수 있었다.

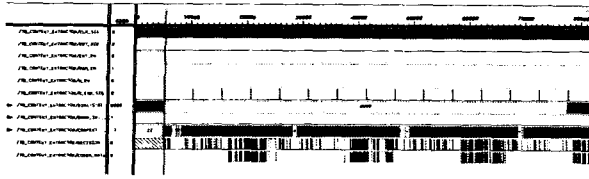


그림 21. 컨텍스트 추출부 RTL 시뮬레이션 결과

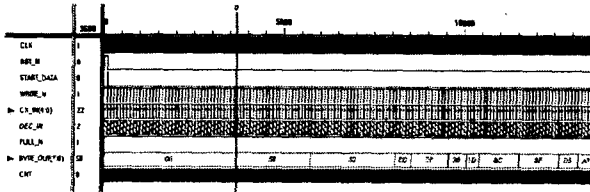


그림 22. MQ-Coder RTL 시뮬레이션 결과

그림 23과 24는 아플로 툴을 이용하여 DWT IP와 양자화 IP를 P&R한 Layout 결과 그림을 보여 주고 있다. 칩 사이즈는 5×5mm와 4×4mm 크기이다.

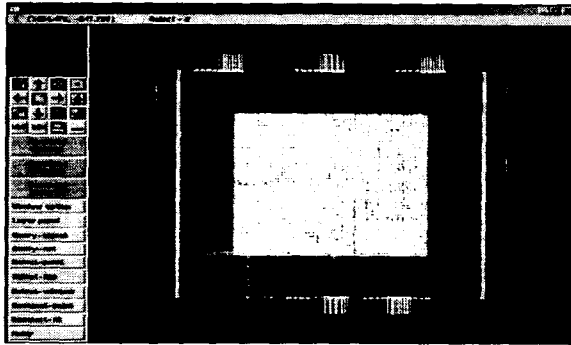


그림 23. DWT IP Layout 결과

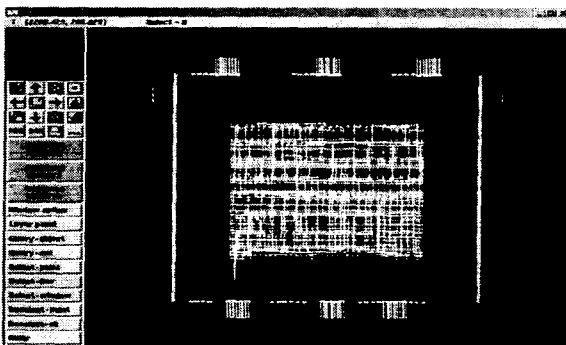


그림 24. 양자화 IP Layout 결과

그림 25는 각 IP는 TransEDA사의 Verification Navigator 중 VN-Check라는 LINT 툴을 이용하여 결과 파일의 일부를 나타낸 것이다. 코딩 표준, 코딩 스타일, 주석처리의 여부, port 신호 및 내부 signal 등에 관한 선언, 사용자 정의 등에 관련하여 VHDL로 작성한 코드가 얼마나 Rule에 맞게 설계되었는지 확인할 수 있었다.

타일, 주석처리의 여부, port 신호 및 내부 signal 등에 관한 선언, 사용자 정의 등에 관련하여 VHDL로 작성한 코드가 얼마나 Rule에 맞게 설계되었는지 확인할 수 있었다.

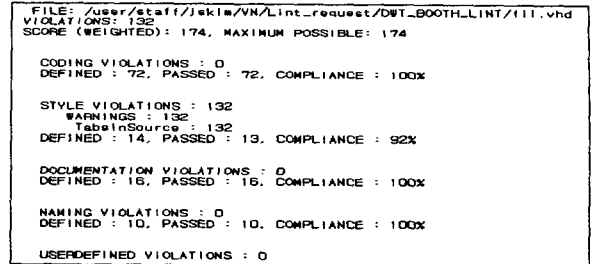


그림 25. LINT 툴 사용 결과

6. 결론 및 향후계획

본 논문에서는 SoC설계를 고려하여 JPEG2000 내부 블록들을 재사용 가능한 IP로 설계하고 검증하였다. 또한, HDL로 설계된 IP들은 C와 HDL를 병행하여 영상시뮬레이션을 수행하였고, 논리합성을 통해 설계된 IP의 성능을 측정하였다. 설계된 IP는 반도체설계사 산연구센터에서 제공하는 'RTL Coding Guideline'에 따라 HDL을 이용하여 하드웨어 설계를 수행하였으며, TransEDA사의 LINT툴을 이용하여 설계된 IP가 얼마나 표준에 맞게 설계되었는지를 검증하였다. 향후계획은 통합 JPEG2000 IP 설계를 수행할 계획이다. 향후계획은 소프트웨어 영역인 마이크로 프로세서로서 ARM9 Core와 하드웨어 영역인 100만 게이트 FPGA 칩 그리고 메모리 등을 이용하여 SoC 설계를 수행할 것이다.

참고문헌

- [1] C. Christopoulos, MediaLab, Ericsson Research, Sweden, "JPEG2000 Verification Model 8.5 (Technical description)", ISO/IEC/JTC1/SC29/WG1 N1878, Sept. 2000
- [2] D. Taubman, "High Performance Scalable Image Compression With EBCOT", Proc. of IEEE International Conference on Image Processing, Kobe, Japan.1993, vol.3,pp.344-348
- [3] Michael Keating and Pierre Bricaud, "Reuse Methodology Manual for System-on-a-Chip Design", KAP, 1998
- [4] A. Said and W.A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees" IEEE Trans. Circuits syst. Video Technol, vol.6,pp.243-250, Jun 1996.
- [5] 김용규, "균일 양자화기의 통계적 특성을 이용한 웨이블릿부호화기의 성능개선", 한국통신논문집, Vol. 26 No.10B May.09,2001.