

# 디지털 시네마용 Motion JPEG2000 인코더의 FPGA 설계

종신회원 서영호\*, 정회원 최현준\*\*, 종신회원 김동욱\*\*

## FPGA Design of Motion JPEG2000 Encoder for Digital Cinema

Young-Ho Seo\* *Lifelong Member*, Hyun-Jun Choi\*\* *Regular Member*,  
Dong-Wook Kim\*\* *Lifelong Member*

### 요 약

본 논문에서는 주요 영화사들로 구성된 DCI(Digital Cinema Initiatives)에 의해 디지털 시네마를 위한 영상 압축 표준으로 제정된 Motion JPEG2000 부호화기를 FPGA를 타겟으로 구현하였다. JPEG2000의 주요 구성요소인 리프팅-기반의 DWT(Discrete Wavelet Transform)와 EBCOT(Embedded Block Coding with Optimized Truncation)의 Tier 1을 하드웨어로 구현하였고, Tier 2과정은 소프트웨어로 구현하였다. 디지털 시네마를 위해 입력 영상의 크기(tile size)는 최대 1024×1024까지의 고해상도를 지원할 수 있도록 하였고, 실시간성을 보장하기 위해 3개의 엔트로피 부호화기를 사용하였다. Verilog-HDL을 이용하여 하드웨어로 구현했을 경우 Altera사의 Stratix EP1S80에서 32,470 LE (logic element)에 해당하는 자원을 사용하면서 FPGA에 사상되었고, 150Mhz의 주파수에서 안정적으로 동작하였다.

**Key Words** : JPEG2000, Digital Cinema, Hardware, FPGA, Lifting, EBCOT

### ABSTRACT

In the paper, a Motion JPEG2000 coder which has been set as the standard for image compression by the Digital Cinema Initiatives (DCI), an organization composed of major movie studios was implemented into a target FPGA. The DWT (Discrete Wavelet Transform) based on lifting and the Tier 1 of EBCOT (Embedded Block Coding with Optimized Truncation) which are major functional modules of the JPEG2000 were setup with dedicated hardware. The Tier 2 process was implemented in software. For digital cinema the tile-size was set to support 1024×1024 pixels. To ensure the real-time operations, three entropy encoders were used. When Verilog-HDL was used for hardware, resources of 32,470 LEs in Altera's Stratix EP1S80 were used, and the hardware worked stably at the frequency of 150Mhz.

### I. 서론

지난 10여 년 동안 정보 및 데이터의 저장매체나 전송기술은 큰 발전을 이루었다. 또한 유선 통신에서 나아가서 무선 기반의 통신을 위한 제반적 기술 및 기반 인프라에 대한 구축이 괄목할 만한 성장을 이루었다. 이와 함께 사용자들은 영상이나 비디오

서비스와 같은 대용량의 정보와 우수한 서비스를 요구하고 있어 이러한 대용량의 데이터들의 효율적인 처리기술의 중요성이 높아지게 되었고 이에 대한 연구가 활발히 진행되어 왔다. 영상 정보를 효율적으로 처리하고자 하는 가장 대표적인 기술이 JPEG과 MPEG 및 H.26X의 표준들이며, 이들 표준들을 응용한 소프트웨어(software, S/W) 혹은 하드

※ 본 연구는 한국과학재단 특정기초연구(R01-2006-000-10199-0)지원으로 수행되었음.

\* 한성대학교 정보통신공학과 (yhseo@hansung.ac.kr), \*\* 광운대학교 전자재료공학과 (hjchoi, dwkim)@kw.ac.kr

논문번호 : KICS2006-11-474, 접수일자 : 2006년 11월 15일, 최종논문접수일자 : 2007년 3월 2일

웨어(hardware, H/W) 제품들이 경쟁적으로 쏟아져 나오고 있다. JPEG 또는 MPEG은 이산 코사인 변환(Discrete Cosine Transform, DCT)을 기반으로 하는 기술로서 여러 각도의 기술적 진보에도 불구하고 블록효과라는 필연적인 단점을 갖고 있다. 이를 보완 및 대체하기 위한 기술이 최근 10여 년간 연구되고 있는데 대표적인 것이 웨이블릿(wavelet)을 기반으로 하는 영상처리이다. DCT와는 다르게 이산 웨이블릿 변환(Discrete Wavelet Transform, DWT)은 블록효과를 제거할 수 있을 뿐 아니라 전체영상을 대상으로 인간의 시각에 따른 처리가 가능하여 JPEG2000<sup>[1]</sup>의 표준 변환으로 이미 지정되었다.

디지털 시네마는 기존의 35mm 필름을 대체하기 위해 최근 10년간 높은 화질, 보안, 그리고 저장기술 등의 다양한 연구를 통해 발전해왔다. 디지털 시네마는 4096×2048이상의 화소 수, 10비트 이상의 색상차성분 표현범위, 그리고 24fps(frame per second)의 프레임율을 요구한다. 압축하지 않은 디지털 시네마의 영상 데이터는 약 1.3terabyte이며, 이는 DVD(Digital Video Disk) 데이터양의 약 300배에 해당한다. 디지털 시네마의 비교적 많은 데이터양을 줄이기 위해 2000년 정지영상 국제 표준으로 채택된 JPEG2000은 주요 영화사들로 구성된 DCI(Digital Cinema Initiatives)의 영상압축 표준으로 채택되면서 다시 주목받고 있다.

JPEG2000은 크게 세 부분으로 나눌 수 있는데 DWT를 수행하여 저주파 성분과 고주파 성분을 분리하여 부대역을 만드는 웨이블릿 변환, 양자화, 그리고 EBCOT(Embedded Block Coding with Optimal Truncation) 과정으로 나눌 수 있다. 이 중에서 중요한 것은 웨이블릿 변환부와 EBCOT부이다.

첫 번째로 DWT는 컨벌루션(convolution) 방식에 비해서 우수한 성능을 보이는 리프팅(lifting) 방식이 주로 연구되고 있다. 리프팅 기법은 기본적인 웨이블릿 변환(컨벌루션 방식)을 이용한 필터링 기법에 비해서 메모리량과 메모리에 대한 참조 횟수가 적고 정변환과 역변환이 동일한 구조로 이루어진다는 장점을 가지고 있다<sup>[2-4]</sup>. 대표적으로 연산의 효율성을 위해 입력 영상을 블록으로 구분지어 처리하는 방식<sup>[2]</sup>과 (5,3)필터를 이용하여 데이터를 인터리빙하는 구조<sup>[3]</sup>들이 연구되었다. 또한 EZW(Embedded Zerotree Wavelet) 알고리즘 기반의 양자화 방식에 적합한 리프팅 구조<sup>[4]</sup>와 예측(predict)과 갱신(update)의 과정을 처리하는 단위 동작을 전체로 확

장하는 구조<sup>[5]</sup>들이 제시되었다. JPEG2000에서 규정하고 있는 여러 종류의 필터에 대해서 적용 가능하도록 확장성을 가진 연산단위와 이들에 의한 리프팅 구조도 제안되었다<sup>[6]</sup>. 데이터패스부가 완벽히 파이프라인화되어 있지 않으면 임계경로가 큰 단점을 가지고, 리프팅 연산을 그대로 H/W화하여 구조적으로 간결성을 유지하려면 연속된 입력 데이터에 대한 처리결과가 모호하게 되고 리프팅 연산과 내부 메모리와의 관계를 명확히 보이지 못한다<sup>[5],[6]</sup>. 이를 해결하고자 전체적으로 파이프라인된 리프팅 연산과 확장성을 가진 H/W 구조<sup>[7]</sup>가 소개되었는데, 고속의 성능을 갖지만 리프팅 알고리즘을 그대로 사상하였기 때문에 파이프라인을 위해 많은 수의 레지스터가 요구되고 라인 기반의 리프팅 방식을 명확히 나타내지 못하는 단점이 있다.

다음으로 JPEG2000은 EBCOT(Embedded Block Coding with Optimized Truncation)<sup>[8-10]</sup>라는 특별한 엔트로피 부호화 방법을 채택하고 있다. 이 방법 중 Tier 1에 속해 있는 컨텍스트(context) 추출 알고리즘은 주위 데이터들과의 상관도에 따라 부호화하는 방법으로 그 효율성이 매우 뛰어나 전체 JPEG2000 연산의 50%를 차지할 만큼 많은 양의 연산이 필요하다<sup>[11]</sup>. 정지영상을 압축하는 JPEG2000의 응용분야는 영상의 추출 후 사용까지 이러한 연산을 충분히 수행할 시간이 있는 분야도 있으나, 추출 즉시 압축하여 전송하여야 하는 등의 응용분야도 최근 들어 많이 고려되고 있어 과도한 연산량을 줄일 필요가 있다<sup>[11-14]</sup>.

본 논문은 다음과 같이 구성된다. 2장에서는 디지털 시네마의 개념에 대해 설명하고, 3장에서는 제안한 전체 하드웨어 구조와 사양, 및 내부 하드웨어 구조를 설명한다. 4장에서 제안한 하드웨어의 구현 결과를 보이고 상용 제품들과 비교를 한다. 마지막으로 5장에서 결론을 맺는다.

## II. 디지털 시네마 시스템

최근 멀티미디어 매체는 급속도로 디지털화 되고 있지만 영화만큼은 아날로그 방식을 고수하고 있다. 이는 영화를 보기 위해 필요한 기록매체인 필름을 대체할 만한 매체가 100년이 넘는 동안 거의 전무했기 때문이다. 하지만 최근 관객들과 영화관계자들 사이에서 필름의 화질에 대한 불만이 점점 높아져 가고 있는 실정이다. 최근 HD의 등장은 필름의 화질에 대한 불만을 더욱 가속화 시키고 있다. 이런

과정에서 상영자체에만 관심을 기울이던 것에서 점점 더 본질적인 문제로 접근하는 연구들이 진행되고 있다. 바로 상영기술로서의 디지털화뿐만 아니라 영상의 촬영자체의 기술로서의 디지털화의 접근이다.

이와 같은 영향으로 디지털 시네마의 개념은 일반적으로 생각하던 디지털 영상개념을 뛰어넘어 촬영, 후반작업, 그리고 상영에 이르기까지 전 과정의 디지털화를 포함하고 있다. 그림 1에서 디지털 시네마의 개념을 도시화 하였다. 그림에서 보이듯이 디지털 카메라로 촬영된 영상은 MJPEG2000 부호화기를 통해 손실·무손실 압축되어 저장되거나 전송되어 후반작업을 거친다. 완성된 디지털 영상은 유선 인터넷 또는 위성을 통해 전송되고 디지털 영사기의 JPEG2000 복호화기를 통해 복원되어 상영된다. 디지털 시네마에서는 압축기술뿐만 아니라 영화의 배급을 위해 유·무선을 이용한 온라인 전송 시 보안 문제도 중요시되고 있다.

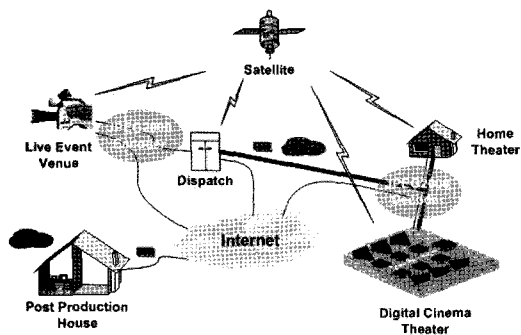


그림 1. 디지털 시네마의 시스템도  
Fig. 1. System diagram of digital cinema.

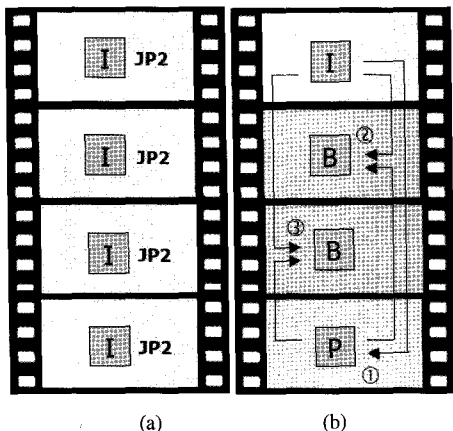


그림 2. MJPEG2000과 MPEG의 부호화 방식<sup>[15]</sup>  
Fig. 2. Encoding method of MJPEG2000 and MPEG.

표 1. 디지털 시네마의 사양  
Table 1. Specification of digital cinema.

Application	projection Electronic Cinema	production HDTV	Projection Digital Cinema	Production Digital Cinema
Parameter	1280×720 YUV422 16-bit 24fps	1920×1080 YUV422 20-bit 24fps	4096×2048 RGB444 30-bit 24fps	4096×3112 RGB444 36-bit 24fps
Resulting Data range	350MBit/s	995MBit/s	6040MBit/s	11000GBit/s
Compression ratio	20:1	7:1	12:1	5:1

디지털 시네마의 영상 압축 기법은 그림 2의 (b)와 같은 일반적인 동영상 압축 기법인 MPEG의 인터 프레임 부호화 기법과는 달리 (a)와 같이 각 프레임을 독립적으로 인트라 프레임 부호화를 수행하는 Motion JPEG2000이 사용된다. 이는 JPEG2000 표준 문서 Part 3에 기술되어 있는 기법이다.

표 1에서는 디지털 시네마의 기술적인 사양을 보이고 있는데 디지털 시네마는 4096×2048이상의 화소수, 10비트 이상의 각 색차성분 표현범위, 그리고 24fps의 프레임율을 요구한다.

### III. 제안한 H/W의 전체적인 구조와 사양

본 논문에서는 차세대 정지영상 압축 표준으로 채택된 JPEG2000 영상압축 프로세서의 구조를 제안하고 최소의 H/W로 구현하였다. 구현된 하드웨어에서 데이터 패스(data-path)부는 웨이블릿 변환을 수행하는 리프팅 기반의 DWT 커널(kernel)부, 엔트로피 부호화부, 그리고 입출력을 위한 인터페이스와 버퍼들로 구성된다. 제어부는 크게 세 가지 부분으로 나뉘는데 프로그래밍 레지스터와 명령어를 디코딩하는 제어 디코더부, 디코딩된 정보를 이용하여 제어 신호를 생성하는 제어신호 발생부, 그 밖에 상태를 외부로 알리는 상태 레지스터로 구성된다. 전체 구조를 그림 3에 나타내었다. 표 2에서는 H/W에 대한 사양을 나타냈다. H/W로 입력되는 데이터는 영상데이터와 제어데이터로 나누어진다. 영상데이터는 동작이 시작되면 동작이 끝날 때까지 계속적으로 입력되지만 제어신호를 만들기 위한 제어데이터는 초기에 한번만 입력되면 된다. 제어데이터를 위한 별도의 포트를 두는 것은 동작의 효율성과 상관없이 하드웨어 자원의 소모를 가져온다. 따라서 포트를 함께 사용하고 두 데이터간의 구분을 위한

최소의 제어신호를 추가하였다. 영상의 크기와 웨이 블릿 필터링 레벨을 비롯한 전반적인 압축 과정은 프로그래밍에 의해서 결정된다.

본 논문의 H/W에서는 Tier2는 설계하지 않았는데 그 이유는 다음과 같다. Tier2는 구성된 packet 을 네트워크 혹은 디코더부의 조건에 따라서 출력하는 단계로써 이미 버퍼에 들어있는 압축된 비트 스트림을 선택만하는 비교적 단순한 과정이다. 따라서 펌웨어를 이용한 간단한 MCU 혹은 시스템 레벨에서의 소프트웨어로 처리하는 것이 일반적인 형태이므로 H/W 설계에 포함시키지 않는다.

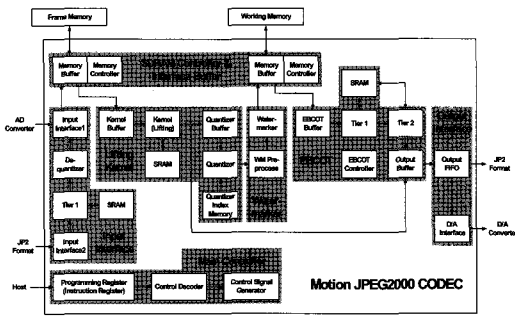


그림 3. 전체 H/W 블록 다이어그램  
Fig. 3. Global H/W block diagram.

표 2. 구현한 하드웨어 사양  
Table 2. Specification of implemented H/W.

	Category	Specification
1	Transform	Forward
2	Image size	User defined
3	The number of DWT level	Max. 6 level
4	DWT filter	(9,7),(5,3)
5	The number of entropy coder	3
6	Programming register	16

### 3.1 DWT 커널의 H/W 구조

본 절에서는 앞장에서 언급한 전체적인 H/W 구조 중에서 일부 중요한 구조를 가지는 블록에 대해서 설명한다<sup>[16]</sup>.

#### 3.1.1 Lifting의 H/W 기반 최적화

리프팅 연산이 수행되는 절차를 도식적으로 나타내면 그림 4와 같다. (9,7) 필터를 이용한 리프팅은 리프팅 계수만 다르고 방식은 동일한 총 4단계의 곱셈과 덧셈과정을 거치는데, 이를 식 (1)-(5)에 나타냈다. 식 (4)의 결과는 식 (5)의 스케일링 과정을

거친다.

$$d_i^{(1)} = d_i^{(0)} + \alpha (s_i^{(0)} + s_{i+1}^{(0)}) \quad (1)$$

$$s_i^{(1)} = s_i^{(0)} + \beta (d_i^{(1)} + d_{i-1}^{(1)}) \quad (2)$$

$$d_i^{(2)} = d_i^{(1)} + \gamma (s_i^{(1)} + s_{i+1}^{(1)}) \quad (3)$$

$$s_i^{(2)} = s_i^{(1)} + \beta (d_i^{(2)} + d_{i-1}^{(2)}) \quad (4)$$

$$s_i = \zeta s_i^{(2)}, d_i = d_i^{(2)} / \zeta \quad (5)$$

여기서  $s_i^{(0)}$ 과  $d_i^{(0)}$ 는 각각  $x_{2i}$ 과  $x_{2i+1}$ 에 해당한다.

리프팅은 동일한 연산이 반복적으로 수행되면서 전체 연산을 구성하는데 이러한 동일한 연산을 단위 연산으로 볼 수 있다. 단위 연산은 같은 시간에 곱셈과 덧셈이 동시에 수행되지 않으므로 하나의 곱셈기와 하나의 덧셈기로 모든 연산을 수행할 수 있는데, 그림 4에 시간적인 클럭순서에 따른 연산방식과 요구되는 레지스터 및 레지스터에 저장되는 데이터를 나타냈다.  $\otimes$ 와  $\oplus$ 기호는 각각 곱셈과 덧셈 연산을 나타낸다.

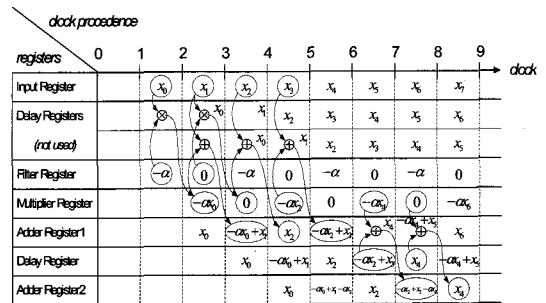


그림 4. 리프팅 연산의 시간 재구성  
Fig. 4. Time rescheduling of lifting.

리프팅의 특성상 하나의 중간 연산결과가 여러 최종 연산결과에 영향을 미친다. 예를 들어 그림 4에서  $s_{i+1}^{(0)}$ 과  $\alpha$ 의 곱셈결과는  $d_i^{(0)}$ 뿐만 아니라  $d_{i+1}^{(0)}$ 를 위한 연산에도 사용된다. 제한한 LBFC에서 다중으로 사용되는 중간 연산결과들을 구조화된 데이터 경로와 연산순서에 따라 최소의 대기지연시간만을 가지면서 여러 연산에 관여한다. 따라서 동일한 결과를 가지는 중간 연산들은 두 번 이상 수행되지 않아서 최소의 연산수와 최적의 생존시간을 가지고 효율적인 리프팅을 구성한다.

3.1.2 Lifting 커널의 H/W 구조

(5,3) 필터와 (9,7) 필터를 모두 수용하면서 연속적으로 입력되는 데이터를 처리할 수 있는 리프팅 단위 셀의 구조를 제안하고 이를 그림 5에 나타냈다. 그림 5는 재구성된 리프팅 연산코드를 H/W 구조로 변환한 것이다. 제안한 H/W를 리프팅 기반의 필터링 셀(LBFC, Lifting-Based Filtering Cell)이라고 한다.

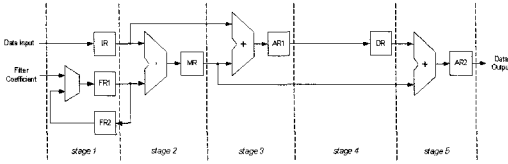


그림 5. 수평방향 LBFC의 구조  
Fig. 5. Architecture of horizontal LBFC(LBFC\_hor).

라인방식의 웨이블릿 변환기법은 2차원 데이터를 대상으로 이루어지는데, 수평방향의 1차원 변환을 수행하면서 2차원 변환을 수행할 수 있는 결과가 마련되면 1차원 변환과 함께 수직방향의 2차원 변환을 동시에 수행하는 것이다. 이와 유사한 과정을 그림 6에 보이고 있는데, 수평방향의 리프팅이 수행되면 일정 대기 지연시간 이후에 저주파와 고주파 계수에 해당하는  $L_{x,y}$ 와  $H_{x,y}$ 가 번갈아가며 매 클록마다 출력된다. 출력된 계수들은 버퍼링 과정없이 곧바로 수직방향의 리프팅 연산에 사용된다. 수평방향의 리프팅 결과가 다섯 번째 줄( $L_{4,0}$ )에 해당하면 일정 대기 지연시간 후에 유효한 수직방향의 리프팅 결과( $LL_{0,0}$ ,  $LH_{0,0}$ )를 출력한다.

수직방향의 리프팅을 수행하기 위해서 라인 기반의 필터링을 채택하고 있으므로 그림 5에서 각 레지스터들이 그림 7과 같이 라인 버퍼(Line buffer)로 대체된다. 수평방향을 위한 리프팅과 동일한 구조를 가지고 있고, 하나의 블록으로 네 부대역의 계수들을 순서대로 출력( $LL_{x,y} \rightarrow LH_{x,y} \rightarrow HL_{x,y} \rightarrow HH_{x,y}$ )한다.

그림 8은 그림 5와 7의 LBFC를 이용한 리프팅 방식의 필터링 구조인 LFDWT(Lifting Filter for DWT)와 리프팅커널(Lifting Kernel)을 나타낸다. (9,7) 필터를 이용하는 경우 4단계의 연산을 거치면서 리프팅 연산이 수행되기 때문에 4개의 LBFC가 요구되고 20 클록의 대기 지연을 가진다. (5,3) 필터를 이용하는 경우는 2개의 LBFC만 필요로 한다. 또한 (5,3) 필터를 사용할 경우 (9,7) 필터에 비

해서 1/2의 H/W 자원만 사용하여 두 배의 필터링을 얻을 수 있는데, 그림 6에서 앞의 두 LBFC와 MUX 다음의 두 LBFC에 의해서 병렬적인 필터링을 수행하고, 이 때 MUX에 의해서 3번째 LBFC로 직접 입력한다. 리프팅을 이용한 필터링은 레벨단위로 이루어지고 f의 속도로 입력된 "Din"은 연산을 거친 후 f의 속도로 네 개의 출력을 직렬로 출력하여 결국 입력과 동일한 f의 출력율을 가진다.

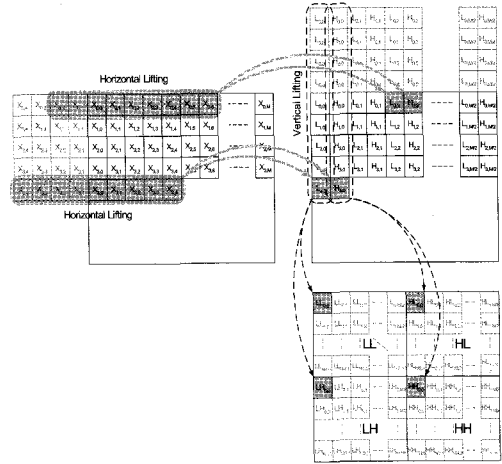


그림 6. 라인기반의 수직방향 리프팅 연산  
Fig 6. Line-based vertical lifting arithmetic.

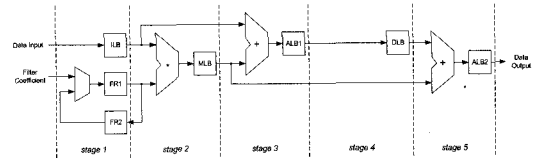


그림 7. 수직방향 LBFC의 구조  
Fig. 7. Architecture of vertical LBFC(LBFC\_ver).

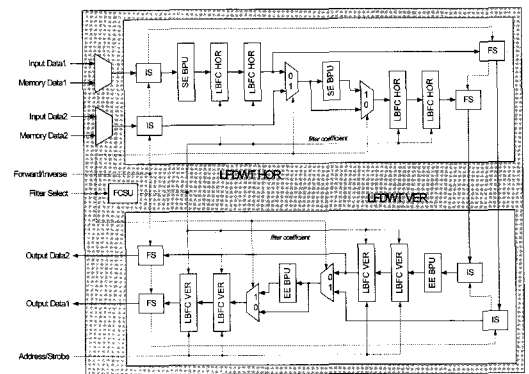


그림 8. LFDWT를 이용한 Lifting Kernel의 구조  
Fig. 8. Architecture of Lifting Kernel using LFDWT.

### 3.2 EBCOT의 H/W 구조

EBCOT알고리즘은 JPEG2000 전체 연산의 70% 이상을 차지하고 있다. 특히 컨텍스트 추출 알고리즘의 수행에서 50%정도의 연산시간을 소비한다. 따라서 실용적인 S/W 및 H/W로의 구현을 위해서는 새로운 알고리즘의 개발이 요구된다. 본 논문에서는 고속 연산을 위한 선행처리 방법으로 제로트리를 기반으로 하는 컨텍스트 추출방법을 제시 한다<sup>[16]</sup>.

#### 3.2.1 제안한 컨텍스트 추출 및 MQ 코딩

웨이블릿 변환은 주파수 영역에서 위치 정보와 주파수 정보를 표현함으로써 영상 압축을 위한 영상의 계층적 구조를 제공한다. 저주파 부분으로 갈수록 웨이블릿 계수들의 크기는 커지고 계수들 간의 변화폭도 크다. 반면, 수직/수평 고주파수 부분으로 갈수록 웨이블릿 계수들의 크기가 작아지고 변화폭도 "0"을 중심으로 작아진다. 이때, 저주파수 부분에 해당하는 부분을 "coarse", 고주파수 부분에 해당하는 부분을 "fine"이라고 한다.

제안한 알고리즘은 다음과 같은 전제조건을 기반으로 한다. coarse부분의 계수가 어떤 임계값 이상으로 중요 계수(Significance Coefficient)로 결정이 되면 fine 부분의 같은 방향성, 같은 위치의 모든 계수들도 중요 계수로 정해진다<sup>[16]</sup>.

이 때, coarse 부분에 해당하는 계수들을 "parent", fine 부분에 해당하는 계수들을 "children" 이라고 한다. 그림 3에서 제로트리 구조를 나타내었다.

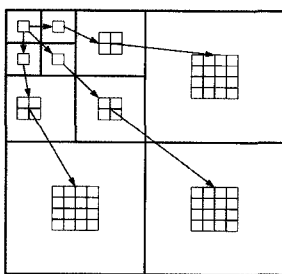


그림 9. 웨이블릿 영역에서 부대역간 제로트리 구조  
Fig. 9. Zero-tree structure in wavelet domain.

5-Level 웨이블릿 변환을 수행하였을 경우, LL5 영역은 기존의 방법으로 코딩을 수행한다. 그리고 컨텍스트 추출 알고리즘의 고속 연산을 위해 고주파 부대역(LH5, HL5, HH5)의 코드블록 계수 값 중에 하나라도 임계값 이상의 값이 있으면 중요 코드블록(Significance Code Block, SCB)으로 정하고 하위 부대역에서 상관도를 가지는 코드블록 역시

SCB로 정한다. 그리고 컨텍스트 추출을 위한 알고리즘의 수행은 SCB로 정해진 코드블록에 한해서만 수행이 된다. 비중요 코드블록(Insiguificance Code Block, ICB) 으로 판정 된 코드블록은 미리 계산된 컨텍스트와 결정값을 출력함으로써 연산 속도를 향상시킨다<sup>[16]</sup>.

#### 3.2.2 제안한 EBCOT의 H/W 구조

그림 10에 컨텍스트 추출부의 전체적인 블록도를 나타내었다. 컨텍스트 추출을 위해 사용되는 5개의 비트평면들을 저장하기 위해 FPGA에서 지원하는 듀얼포트램(Dual-port RAM)을 사용하였고, 4가지 부호화(제로부호화, 크기 정제 부호화, 런 길이 부호화, 부호 부호화)을 수행하는 전용하드웨어 엔진들을 설계하여 사용하였다. 그림 11에서는 컨텍스트 추출 프로세서에서 출력된 컨텍스트와 결정값으로 산술부호화를 수행하는 MQ 부호화기의 전체 블록도를 보이고 있다. MQ 부호화기는 컨텍스트 추출 프로세서에서 출력된 컨텍스트와 결정값을 입력 받아 확률 예측 표에서 미리 정해진 확률값을 Q 레지스터에 저장하여 압축 연산을 진행하게 된다. 압축된 데이터는 B 레지스터에 채워져 8-bit 단위로 출력된다.

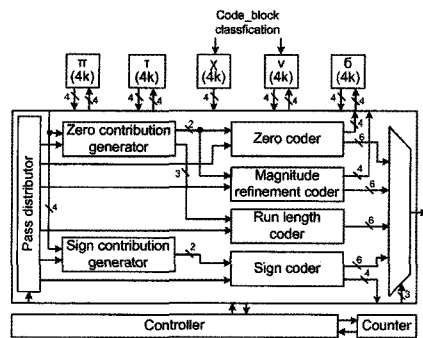


그림 10. 컨텍스트 추출부의 구조  
Fig. 10. Architecture of context extractor.

그림 10에서 보이듯이 제안된 컨텍스트 추출 프로세서는 6개의 전용 하드웨어 블록들로 구성되어 있다. Zero Contribution 생성기와 Sign Contribution 생성기로부터 contribution 값을 입력받은 4개의 부호화 블록들은 정해진 컨텍스트와 결정값을 출력하고, 마지막 단의 MUX를 통해 정해진 패스별로 출력된다. 또한 그림 12에서 zero 부호화기와 Zero Contribution 생성기의 구조를 보이고 있다. 그림 13에서는 contribution을 계산하는 Sample Array의

내부 구조를 보이고 있다. 그림 14에서는 Sample Array의 연산 순서를 보이고 있는데 Sample Array에서 계산된 contribution 값을 받아 Zero Coder는 미리 정해진 9개의 컨텍스트(0~8)를 출력한다.

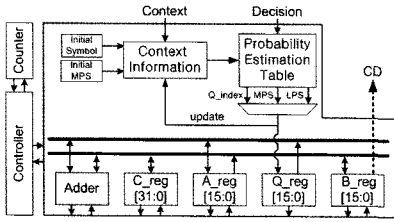


그림 11. MQ 부호화기의 구조  
Fig. 11. Architecture of MQ encoder.

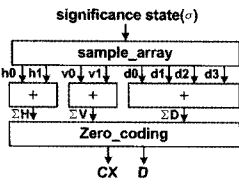


그림 12. Zero contribution 생성기와 coder의 구조  
Fig. 12. Architecture of zero contribution generator and coder.

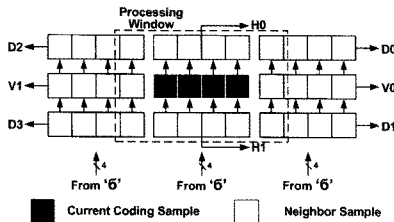


그림 13. Sample array의 구조  
Fig. 13. Architecture of sample Array.

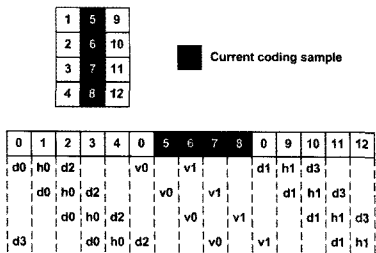


그림 14. Sample array의 연산 순서  
Fig. 14. Operational sequence of sample array.

IV. 구현 결과

본 논문에서 제안한 MJPEG2000 부호화기는 Altera의 FPGA를 타겟으로 구현하였다. S/W를 이

용하여 검증하였고 이를 Verilog-HDL을 사용하여 RTL 수준의 H/W로 사상하였다. 입력 영상을 타일 단위로 처리하고 라인버퍼는 최대 1024×1024 크기의 타일을 수용할 수 있도록 하였다. 내부적 동작 메모리로 FPGA 내부의 ESB(Embedded System Block)을 사용하였다. 타겟 FPGA는 Altera사의 Stratix EP1S80 칩을 사용하였고, FPGA 사상을 위해 Synplify에서 합성을 수행하였다.

실험영상을 4-레벨 웨이블릿 변환한 후 제안한 선형 처리과정을 적용한 결과로 LL4에서의 IC(Insignificant Coefficient)의 비율은 7~8%에 불과하나 LH4, HL4, HH4의 다음 고주파 부대역에서는 20~40%로 급증하여 최저주파수대역인 LL4에 에너지가 집중되어 있음을 알 수 있다<sup>16)</sup>. 특정 트리구조를 이루는 부대역들에서 IC의 개수는 고주파 부대역으로 갈수록 높아지나 그 비율은 반드시 증가하지는 않는다. 또한 각 트리구조의 IC 비율은 그 영상의 방향성에 따라 값이 서로 다름을 알 수 있다. 또한 컨텍스트를 추출하기 위한 연산량에 대해 제안한 방법을 적용한 결과를 살펴보면 트리구조를 이용함으로써 임계값 이하의 계수들 및 그 자손들은 컨텍스트 추출 과정을 거치지 않으므로 계산량이 확연히 감소(최소 4%에서 최대 67%)함을 알 수 있다<sup>17)</sup>.

구현한 하드웨어는 Stratix EP1S80 FPGA에서 32,470 LEs에 해당하는 자원을 사용하면서 사상되었고, 150Mhz의 주파수에서 안정적으로 동작하였다. 본 논문에서 FPGA를 타겟으로 구현한 결과를 표 3에서 상용 제품 및 기존 연구결과와 비교하였다.

표 3. 기존 연구와의 비교  
Table 3. Comparison of recent research.

	[18]	[19]	[20]	Ours
Tech.	FPGA XC2V3000	ASIC 0.18µm	FPGA XC2V6000	FPGA Stratix EP1S80
Title Size	128×128	128×128	512×512	1024×1024
Cblk Size	32×32	32×32	64×32	32×32
DF	(5,3)-lossless (9,7)-lossy	(5,3)-lossless (9,7)-lossy	(5,3)-lossless (9,7)-lossy	(5,3)-lossless (9,7)-lossy
EC	8	3	10	3
Clk	96Mhz	200Mhz	139.3Mhz	150Mhz

그림 15에는 본 논문을 통해 구현한 MJPEG2000 코덱을 검증하기 위해 제작된 FPGA 검증용 보드이다. CCD 카메라, A/D 보드, LCD 패널, LCD 스케일러, 및 FPGA 보드로 구성되어 있다.

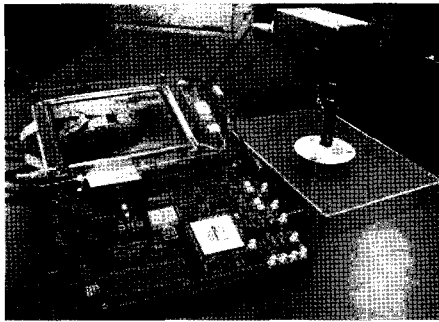


그림 15. 구현한 회로의 검증 시스템  
Fig. 15. Verification system of the implemented circuits.

### V. 결론

본 논문에서는 디지털 시네마에서 디지털 영화의 압축표준으로 채택된 JPEG2000을 H/W로 구현하였다. JPEG2000의 구현을 위해서 JPEG2000 알고리즘을 H/W 구현에 적합하게 최적화하는 과정을 거쳤고, 이 과정은 S/W를 이용하여 수행하였다. 수정된 JPEG2000 알고리즘은 Verilog-HDL을 이용하여 H/W로 구현되었고 FPGA에 프로그래밍하여 동작을 검증하였다. 구현된 전체 하드웨어는 크게 리프팅 프로세서와 EBCOT, 입출력 인터페이스, 외부 메모리 제어기, 그리고 제어기 등으로 구성된다. 이 중에서 가장 중요한 것은 리프팅 프로세서와 EBCOT Tier-1 블록에 해당한다.

본 논문에서 제안한 MJPEG2000 코덱은 IP화하여 디지털 시네마를 비롯하여 차세대 방식의 디지털 카메라와 고성능 DVR 시스템 등의 다양한 응용 분야로 확장할 계획이다.

### 참고 문헌

[1] JPEG2000 Final Committee Draft (FCD). JPEG2000 Committee Drafts.[Online]. Available: <http://www.jpeg.org/CDs15444.htm>.  
 [2] K. K. Parhi and T. Nishitani, "VLSI architectures for discrete wavelet transforms," *IEEE Trans. VLSI Syst.*, vol. 1, pp. 191-202, June 1993.  
 [3] A. Grzeszczak, M. K. Mandal, S. Panchanathan, and T. Yeap, "VLSI implementation of discrete wavelet transform," *IEEE Trans. VLSI Syst.*, vol. 4, pp. 421-433, June 1996.

[4] G. Lafruit, L. Nachtergaele, J. Bormans, M. Engels, and I. Bolsens, "Optimal memory organization for scalable texture codecs in MPEG-4," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, pp. 218-243, Mar. 1999.  
 [5] M. Ferretti and D. Rizzo, "A parallel architecture for the 2-D discrete wavelet transform with integer lifting scheme," *J. VLSI Signal Processing*, vol. 28, pp. 165-185, July 2001.  
 [6] K. Andra, C. Chakrabarti, and T. Acharya, "A VLSI architecture for lifting-based forward and inverse wavelet transform," *IEEE Trans. on Signal Processing*, vol. 50, no. 4, April 2002.  
 [7] G. Dillen, B. Georis, J. D. Legat, and O. Cantineau, "Combined Line-Based Architecture for the 5-3 and 9-7 Wavelet Transform of JPEG2000," *IEEE Transactions on Circuit Syst. Video Technol.*, vol. 13, no. 9, Sep. 2003.  
 [8] D. Taubman and HP Labs, "Report on core experiment CodEbf22, EBCOT: Embedded block coding with optimized truncation," Tech. Rep.N1020R, ISO/IEC JTC1/SC29/WG1, October 1998.  
 [9] D. Taubman, "High performance scalable image compression with EBCOT," *IEEE Trans. Image Processing*, Vol. 9, pp. 1158-1170, July 2000.  
 [10] D. Taubman, E. Ordentlich, M. Weinberger, and G. Seroussi, "Embedded block coding in JPEG2000," *IEEE Trans. Signal Processing: Image Commun.*, Vol. 17, No. 1, pp. 49-72, Jan. 2002.  
 [11] C. Lian, K. Chen, and L. Chen, "Analysis and Architecture Design of Block-Coding Engine of EBCOT in JPEG2000," *IEEE Trans. on Circuit and Systems for Video Technology*, Vol.13, No. 3, pp. 219-229, March 2003.  
 [12] K. Andra, C. Chakrabarti, and T. Acharya, "A High-Performance JPEG2000 Architecture," *IEEE Trans. on Circuit and Systems for Video Technology*, Vol. 13, No. 3, March 2003.  
 [13] J. S. Chiang, Y. S. Lin, and C. Y. Hsieh, "Efficient Pass-Parallel Architecture for EBCOT in JPEG2000," *IEEE Int. Circuits and Systems*, Vol. I, pp. 773-776, 2002.  
 [14] H. Yamauchi, S. Okada, K. Taketa, and T.

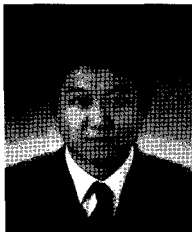


Ohyama, "A Single-Chip JPEG2000 Encoder Processor Capable of Compressing D1-Image at 30frames/s without Tile Division," *IEICE Trans. ELECTRON.*, Vol.,E87-C, No. 4, April 2004.

- [15] E. Edwards and S. Goessel, "JPEG2000 for Digital Cinema Applications," April 25, 2001.
- [16] 서영호, 김동욱, "Motion JPEG2000을 위한 리프팅 프로세서의 ASIC 설계", *한국통신학회 논문지* 제 30권 제 7C호, pp. 647-657, 2005, 8.
- [17] 최현준, 서영호, 김동욱, "DWT 계수의 트라구조를 이용한 네트워크적용적 JPEG2000 컨텍스트 추출방법", *한국통신학회 논문지* 제 30권 제 9C호, pp. 939-948, 2005, 9.
- [18] JPEG2000 Decoder: BA111JPEG2000D Factsheet. Barco-Silex, October 2003.
- [19] K. Andra, C. Chakrabarti, and T. Acharya, "A High-Performance JPEG2000 Architecture," *IEEE Trans. on Circuit and Systems for Video Technology*, Vol. 13, No. 3, March 2003.
- [21] A. Descampe, F. Devaux, "A Flexible, Line-Based JPEG2000 Decoder for Digital Cinema," *IEEE MELECON 2004*, May 2004.

**서 영 호 (Young-Ho Seo)**

중신회원



1999년 2월 : 광운대학교 전자재료공학과 졸업(공학사).  
 2001년 2월 : 광운대학교 일반대학원졸업(공학석사).  
 2000년 3월~2001년 12월 : 인티스닷컴(주) 연구원.  
 2004년 8월 : 광운대학교 일반대학원졸업(공학박사)

2003년 6월~2004년 6월 : 한국전기연구원 연구원  
 2004년 12월~2005년 8월 : 유한대학 연구교수  
 2005년 9월~현재 : 한성대학교 전임강사  
 <관심분야> 2D/3D 영상 및 비디오 처리, 디지털 홀로그래, SoC 설계, 워터마킹/암호화  
 e-mail : yhseo@hansung.ac.kr

**최 현 준 (Hyun-Jun Choi)**

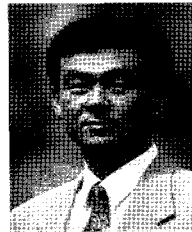
정회원



2003년 2월 광운대학교 전자재료공학과(공학사)  
 2005년 2월 광운대학교 전자재료공학과(공학석사)  
 2005년 3월~현재 광운대학교 전자재료공학과 박사과정  
 <관심분야> Image Processing, 디지털 홀로그래, 디지털 콘텐츠 보안, FPGA/ASIC 설계

**김 동 욱 (Dong-Wook Kim)**

중신회원



1983년 2월 한양대학교 전자공학과 졸업(공학사).  
 1985년 2월 한양대학교 대학원 졸업(공학석사).  
 1991년 9월 Georgia공과대학 전기공학과 졸업(공학박사).  
 1992년 3월~현재 광운대학교 전자재료공학과 정교수. 광운대학교 신기술 연구소 연구원.

2000년 3월~2001년 12월 인티스닷컴(주) 연구원.  
 <관심분야> 디지털 VLSI Testability, VLSI CAD, DSP 설계, Wireless Communication