

논문 2008-4-8

## 저전력 휴대용 멀티미디어를 위한 H.264 디블록킹 필터 설계

## Design of H.264 deblocking filter for the Low-Power Portable Multimedia

박상우\*, 허정화\*\*, 박상봉\*\*\*

Sang Woo Park\*, Jeong Hwa Heo\*\*, Sang Bong Park\*\*\*

**요약** 본 논문은 휴대용 멀티미디어를 위한 저전력 H.264 디블록킹 필터를 제안하였다. H.264 디블록킹 필터는 총 8개의 입력 픽셀에 대한 각각의 필터링 연산 과정을 필요로 하며, 각 필터링 과정에서 p, q 픽셀에 대해 공통 구조를 가지고 있다. 이 때 쓰이는 공통의 필터계수와 레지스터를 공유함으로써, 적은 게이트로 구현하였다. 또한 많은 연산을 필요로 하는 필터링 연산을 특정한 조건을 이용하여, 조건에 만족하면 일부 또는 전체의 필터링을 수행하지 않음으로써 저전력의 효율적인 구조를 설계할 수 있다. 제안한 H.264 디블록킹 필터 구조는 기존 논문들의 핵심 필터링 부분과 비교하여 각각 33.31%와 10.85%의 게이트 감소효과를 나타내었다. 또한 본 논문의 전체 블록은 삼성 0.35um 표준 셀 라이브러리 공정을 사용하여 구현하였으며, 최대 동작 주파수는 108MHz, 최대 처리능력은 CCIR601 형식에서 33.03 frame/s이다.

**Abstract** This paper proposed a H.264 deblocking filter for the portable low-power multimedia. In H.264 deblocking filter, total 8 input pixels in filtering operations needs own filtering operation process respectively, and each filtering process has common structures for each filtering operation. By sharing common filter coefficients and registers, we have designed and implemented an smaller gated module, and moreover filtering operations are skipped on some or whole pixels what if we use some specific condition to operate filtering modules that need lots of operations. In the core of filtering modules, we achieve 33.31% and 10.85% gate count reduction compared with those of filtering modules of the conventional deblocking filter papers. The proposed low-power deblocking filter is implemented by using samsung 0.35um standard cell library technology, the maximum operationh frequency is 108MHz, and the maximum throughput is 33.03 frames/s with CCIR601 image format.

**Keywords** : H.264, deblocking filter, low-power, boundary strength, blocking artifacts

## 1. 서 론

현재 실용화 단계인 화상회의, 화상전화, 디지털 멀티미디어 방송(Digital Multimedia Broadcasting, DMB), 그리고 고품질의 화질을 나타낼 수 있는 고선명 텔레비전(High Definition Television, HDTV) 등은 멀티미디어 정보의 많은 영상이 부호화/복호화 되어야 하는데 이 정

보는 한 화면의 픽셀을 블록 단위로 연산하고 전송해야 하므로 많은 정보를 차지하고 있다. 이러한 많은 영상정보를 압축하고 전송하는데 과거의 H.261부터 지금의 H.264/MPEG-4 Part10까지 많은 압축 표준안이 발전되어 오고 있다. [1][2][3]

H.262/MPEG-4 AVC는 비디오 표준은 이전에 많이 사용되고 있었던 MPEG-2를 대체하는 새로운 비디오 코덱의 표준으로서, 기존의 MPEG-2보다 두 배의 압축률, MPEG-4 ASP보다 40%의 압축률로 동일한 화질을 구현할 수 있는 압축 기술이다. 그러나 이동통신환경에서

\*정회원, 아이앤씨테크놀로지(I&amp;C Technology)

\*\*정회원, 세명대학교

접수일자:2008.7.15, 최종수정일자:2008.8.2

H.264를 실시간 처리하기 위해서는 많은 연산량을 필요로 한다. 그 중 H.264의 전체 연산량의 20~30%를 차지하는 디블록킹 필터는 연산량을 줄이기 위해 많은 연구가 이루어지고 있다. 인코더와 디코더에서 처리되는 모든 화면들은 블록단위 기반의 블록 이산 역현 변환(block discrete cosine transform, BDCT)를 하기 때문에 블록킹 현상(blocking artifacts)이 발생하는데, 이 블록킹 현상을 제거하기 위하여 디블록킹 필터를 사용하고 있다. H.264 디블록킹 필터 연구의 분야는 메모리 액세스 시간을 줄이는 스케줄링 방식의 연구와 필터링 연산 블록의 연산량 감소의 연구 분야로 나뉘어 진행되고 있다.<sup>[4]</sup>

본 논문에서는 H.264에서 사용되는 디블록킹 필터가 파라미터  $\alpha$ 와  $\beta$ 가 0일 때는 필터링 조건에 만족하지 않는 것과 핵심 필터링 부분의 연산을 두 개의 클럭을 이용하여 효율적으로 구현한 저 전력 구조를 제안한다.

본 논문의 II장에서는 H.264 디블록킹 필터의 블록 픽셀 단위와 파라미터  $B_s$ ,  $\alpha$ ,  $\beta$  그리고  $tc_0$ 에 대해 설명하고, III장에서는 제안한 H.264 디블록킹 필터에서 사용된 효율적인 필터링 연산과 모드 결정 블록에 대해 설명하였으며, IV장에서는 시뮬레이션 결과를 통한 기존 논문의 성능비교를 통해 평가한다. 마지막으로 V장에서는 결론을 맺는다.

## II. H.264 디블록킹 필터

H.264의 디블록킹 필터는 먼저 수평방향으로 필터링을 한 다음 수직방향으로 필터링을 하는 순서로 필터링 과정을 실행한다. 디블록킹 필터 처리는 휘도와 색차 성분을 분리해서 처리하며, 매크로 블록을 기본 단위로 하여 처리된다. 블록의 수직 경계 면에서 먼저 필터 처리가 되는데, 매크로 블록 내부의 왼쪽에서 오른쪽으로, 수평적인 블록 경계 면이 위에서 아래의 순서로 필터 된다. 휘도의 디블록킹 필터 처리는 4개의 16화소 블록 경계에 실행되고, 색차 성분은 2개의 8화소 블록 경계에 실행된다. 그림 1은 매크로블록에서 필터 되는 휘도와 색차 블록의 수직과 수평 경계를 나타내고, 그림 2는 필터 처리되는 기본 픽셀 단위를 수직과 수평경계를 중심으로 보여준다. 그림에서 보는 것과 같이 필터처리를 하기 위해 정해진 기본 화소의 단위는  $p_3 \sim p_0$ ,  $q_0 \sim q_3$ 로 이루어져 있으며, 화소들은  $4 \times 4$  블록의 경계를 기준으로 정해지

며, 필터된 결과는  $p_2 \sim p_0$ ,  $q_0 \sim q_2$ 인 화소 값만이 대응되는 화소로 바꾸어 출력된다.  $4 \times 4$  블록의 필터링은 해당 블록의 코딩 모드, 계수, 블록 경계 값에 따라 필터링의 경계 강도(boundary strength,  $B_s$ )가 그림 3과 같이 결정된다. 결정된 경계 강도에 따라 필터링 정도가 달라지는데,  $B_s = 0$ 인 경우는 필터링이 적용되지 않는 no filtering 블록,  $B_s = 1 \sim 3$ 인 경우는 최소 2개의 픽셀, 최대 4개의 픽셀이 필터링 되는 standard filtering 블록,  $B_s = 4$ 는 최소 2개의 픽셀, 최대 6개의 픽셀이 필터링 되는 가장 강력한 strong filtering 블록이 각각 실행된다. 핵심 필터링 과정에서 필요한 파라미터  $\alpha(0 \sim 255, 8bit)$ ,  $\beta(0 \sim 18, 5bit)$  그리고  $tc_0(0 \sim 15, 5bit)$ 는 그림 4, 5와 같다. <sup>[5][6][7]</sup>

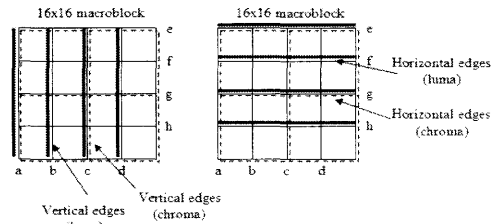


그림 1. 매크로블록에서 필터되는 블록의 수직, 수평 경계  
Fig 1. Vertical and Horizontal boundary of blocks filtered in Macroblocks

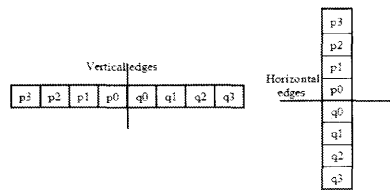


그림 2. 필터처리되는 기본 화소 단위(수직, 수평경계)  
Fig 2. Basic pixel unit format for filtering process (Vertical, Horizontal boundary)

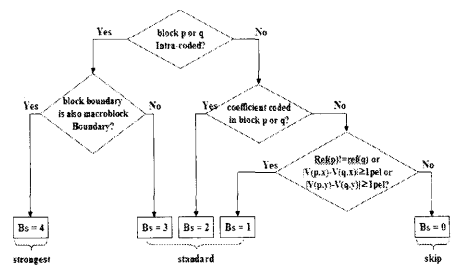


그림 3. 경계 강도 결정 흐름도  
Fig 3. bs(boundary strength) decision flow

Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
$\alpha$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4
$\beta$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2

Index	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
$\alpha$	5	6	7	8	9	10	12	13	15	17	20	22	25	28	32	36	40	45
$\beta$	2	3	3	3	3	4	4	4	6	6	7	7	8	8	9	9	10	10

Index	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
$\alpha$	50	56	63	71	80	90	101	113	127	144	162	182	203	226	255	255
$\beta$	11	11	12	12	13	13	14	14	15	15	16	16	17	17	18	18

그림 4. indexA와 indexB에 따른 alpha, beta 파라미터  
Fig 4. Parameters alpha and beta depending on indexA and indexB

IndexA	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
bs=1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bs=2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bs=3,4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

IndexA	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
bs=1	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	2	2
bs=2	0	0	0	1	1	1	1	1	1	1	1	1	1	1	2	2	2	3
bs=3,4	1	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	4	4

IndexA	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	
bs=1	2	3	3	3	4	4	4	4	5	6	6	7	8	9	10	11	13
bs=2	3	3	4	4	5	5	6	7	8	8	10	11	12	13	15	17	
bs=3,4	4	5	6	6	7	8	9	10	11	13	14	16	18	20	23	25	

그림 5. indexA에 따른 tc0 파라미터  
Fig 5. Parameter tc0 depending on indexA

### III. 제안하는 H.264 디블록킹 필터 구조

#### 1. 제안하는 디블록킹 필터 블록도

그림 6은 제안하는 저전력 H.264 디블록킹 필터의 블록도를 보여준다. 모든 블록은 하드웨어 설계 언어인 Verilog HDL을 사용하여 기술하였고, behavioral로 기술하여 회로 합성을 통해서 구현하였다. 최종 단계에서는 p, q 화소 값을 필터링하여 블로킹 현상(blocking artifacts)가 제거된 값을 출력시킨다. 우선 bs\_decision에서는 p, q 블록의 코드된 모드와 코드된 계수, 블록의 경계값, p, q 블록의 참조 프레임, 움직임 벡터 값에 따라 bs 값을 결정한다. parameter\_calculation과 pre\_calculation에서 필터링 연산에 필요한 변수를 각각 alpha, beta, tc0와 apq, ap, aq, bp, bq, 그리고 b0를 생성한다. mode\_selection에서는 생성한 변수를 이용하여 필터링의 종류를 결정한다. 다음의 각각의 no\_filtering, standard\_filtering, strong\_filtering모드는 BS(boundary strength)

에 따라 결정되는 부분이다. mode\_selection에서 결정된 모드에 따라 p\_q\_mux에서 필터링한 p, q 화소 값을 최종 출력한다. 메인 클럭은 108MHz이며, 사용된 클럭은 108MHz와 13.5MHz로서 최종 출력인 p\_q\_mux의 출력은 13.5MHz로 출력된다. 본 논문에서 제안하는 핵심 필터링 블록과 모드 결정 블록은 다음 절에서 자세히 설명하겠다.

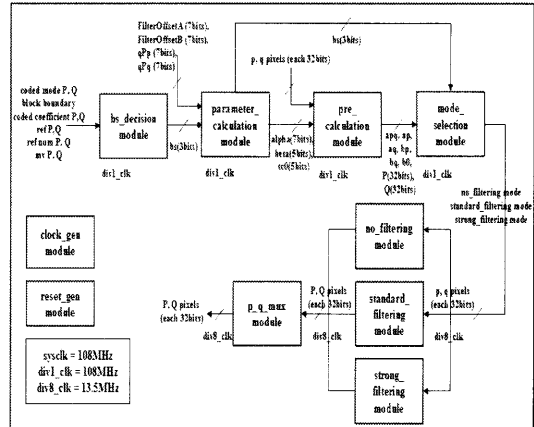


그림 6. 제안하는 H.264 디블록킹 필터 블록도  
Fig 6. Proposed H.264 deblocking filter block diagram

#### 2. 모드 결정 블록

모드 결정 블록은 필터링 연산을 하기 전에 필터링 조건에 해당하는 경우를 결정하여 해당하는 모드에 맞게 필터링 연산을 결정하는 모듈이다. 식 (1)은 필터링 조건식을 보여준다.

$$bs \neq 0 \ \&\& \ bp = |p2-p0| < \beta \ \&\& \ bq = |q2-q0| < \beta \ \&\& \ b0 = |p0-q0| < (\alpha/4+2) \quad (1)$$

그림 4에서 보면 indexA와 indexB에 따라 파라미터  $\alpha$ 와  $\beta$ 가 결정되는데, 조건식에서 보면  $\alpha$ 와  $\beta$ 가 0일 때는 필터링 조건식에 만족하지 않는다. 이에 따라 indexA또는 indexB가 16보다 작을 경우에 연산량이 많은 strong, standard 블록 대신 필터링 하지 않는 블록을 실행함으로써 연산량의 감소효과를 가져올 수 있다.

#### 3. 제안한 Bs = 4 일 때의 필터링 구조

H.264 디블록킹 필터의 핵심 필터링 블록 중의 하나인

strong filtering 블록으로  $B_s = 4$ 이고, 모드 결정 블록에서 필터링 조건에 만족하였을 때, 식 (2), (3)과 같이 최소 2개의 픽셀, 최대 6개의 픽셀을 필터링하는 블록이다. 입력 픽셀은  $p_3, p_2, p_1, p_0$ 이며, 출력 픽셀은  $p'_3, p'_2, p'_1, p'_0$ 의 형식으로 표기한다.

$$\begin{aligned} &\text{if luma and } ap=|p_2-p_0|<\beta \text{ and } apq=|p_0-q_0|<((\alpha/4)+2) \\ p'_0 &= (p_2 + 2p_1 + 2p_0 + 2q_0 + q_1 + 4) >> 3 \\ p'_1 &= (p_2 + p_1 + p_0 + q_0 + 2) >> 2 \\ p'_2 &= (2p_3 + 3p_2 + p_1 + p_0 + q_0 + 4) >> 3 \\ \text{else} \\ p'_0 &= (2p_1 + p_0 + q_1 + 2) >> 2 \end{aligned} \quad (2)$$

$$\begin{aligned} &\text{if luma and } aq=|q_2-q_0|<\beta \text{ and } apq=|p_0-q_0|<((\alpha/4)+2) \\ q'_0 &= (p_1 + 2p_0 + 2q_0 + 2q_1 + q_2 + 4) >> 3 \\ q'_1 &= (p_0 + q_0 + q_1 + q_2 + 2) >> 2 \\ q'_2 &= (2q_3 + 3q_2 + q_1 + q_0 + p_0 + 4) >> 3 \\ \text{else} \\ q'_0 &= (2q_1 + q_0 + p_1 + 2) >> 2 \end{aligned} \quad (3)$$

strong filtering 블록을 실행하기 위한 식을 보면, 모든 필터계수가 1, 2, 또는 3이므로 덧셈연산만으로 설계가 가능하다.  $p'_0 \sim p'_2$  픽셀과  $q'_0 \sim q'_2$  픽셀을 살펴보면, 각각 서로 다른 값을 가지고 있지만 같은 구조를 가지고 있는 것을 확인할 수 있다. 이러한 같은 구조를 갖는 특징을 이용하여 클럭의 posedge와 negedge를 이용하여  $p'_0 \sim p'_2$ 와  $q'_0 \sim q'_2$  픽셀의 연산을 한 클럭에서 동시에 수행할 수 있다. 이것을 토대로 설계된 블록은, 기존 논문에서 사용되는 mux와 demux 블록을 사용하지 않게 됨으로써 전체적인 연산량의 감소효과를 가져오게 된다. 이러한 연산량의 감소효과는 저전력의 효과로 이어지게 된다.

#### 4. 제안한 $B_s = 1 \sim 3$ 일 때의 필터링 구조

H.264 디블록킹 필터의 핵심 필터링 블록 중의 다른 하나로  $B_s = 1 \sim 3$ 이고, 모드 결정 블록에서 필터링 조건에 만족하면, 식 (4), (5)과 같은 기본 필터 연산과 식 (6), (7)과 같은 조건 필터 연산으로 구성되며, 최소 2개의 픽셀, 최대 4개의 픽셀을 필터링하는 블록이다.

$\Delta$ 는 픽셀을 조절하는 값으로써, 픽셀값의 블러링을 막기 위해 식 (5)와 같이 연산하며, 기본 필터 연산으로 모든 경우에 연산을 한다.

$$\begin{aligned} p'_0 &= p_0 + \Delta q'_0 = q_0 - \Delta \\ \Delta &= \text{Min}(\text{Max}(-tc, ((q_0-p_0) < 2 + (p_1-q_1) + 4) >> 3), tc) \end{aligned} \quad (4)$$

$$\begin{aligned} tc &= tc_0 + ((ap < \beta)?1:0) + ((aq < \beta)?1:0) \\ \text{for luma } tc &= tc_0 + 1 \quad \text{for chroma} \end{aligned} \quad (5)$$

standard filtering 블록을 실행하기 위한 식을 보면,  $p'_1$  픽셀과  $q'_1$  픽셀을 살펴보면, 서로 다른 값을 가지고 있지만 같은 구조를 가지고 있는 것을 확인할 수 있다. strong filtering 블록과 마찬가지로 이러한 같은 구조를 갖는 특징을 이용하여 클럭의 posedge와 negedge를 이용하여  $p'_1$ 와  $q'_1$  픽셀의 연산을 동시에 수행할 수 있다. standard filtering 블록은 위에서 설명한 strong filtering 블록과 같은 연산량 감소효과를 가져오게 된다.

$$\begin{aligned} &\text{if luma and } ap = |p_2-p_0|<\beta \\ p'_1 &= p_1 + \text{Min}(\text{Max}(-tc_0, (p_2 + ((p_0+q_0+1) >> 1) - 2p_1) >> 1, tc_0)) \\ &>> 1, tc_0) \\ \text{else} \\ p'_1 &= p_1 \end{aligned} \quad (6)$$

$$\begin{aligned} &\text{if luma and } aq = |q_2-q_0|<\beta \\ q'_1 &= q_1 + \text{Min}(\text{Max}(-tc_0, (p_2 + ((p_0+q_0+1) >> 1) - 2q_1) >> 1, tc_0)) \\ &>> 1, tc_0) \\ \text{else} \\ q'_1 &= q_1 \end{aligned} \quad (7)$$

## IV. 시뮬레이션 결과 및 고찰

본 절에서는 Verilog-HDL을 이용한 시뮬레이션과 FPGA 장비를 이용한 시뮬레이션 2가지 방법으로 실험 결과를 고찰하였다.

### 1. Verilog-HDL 시뮬레이션

이 절에서는 III장에서 제안한 전체 디블록킹 필터를 Verilog-HDL로 설계 후, 시뮬레이션 결과와 기존 구조와 비교함으로써 효과를 검증한다. 제안한 H.264 디블록킹 필터는 삼성 0.35 $\mu$ m 표준 셀 라이브러리 공정을 사용하여 설계하였다. 각 블록별 모듈과 TOP 모듈에 대한 시뮬레이션은 Synopsys사의 design compiler를 이용한 게이트 합성을 수행하였다.

그림 7은 제안한 H.264 디블록킹 필터의 전체 블록의

시물레이션 결과이다. 결과로부터 각 블록에서 정상 동작함을 확인하였다. 그림 8은 Auto P&R툴인 Apollo를 이용한 레이아웃으로 전체 칩 코어 면적은 5mm x 5mm이며, QFP 208핀 pad 정보를 사용하였으며, 3.3V 전원 전압으로, 전체 블록의 테스트를 위해 108MHz 주파수를 이용하였다.

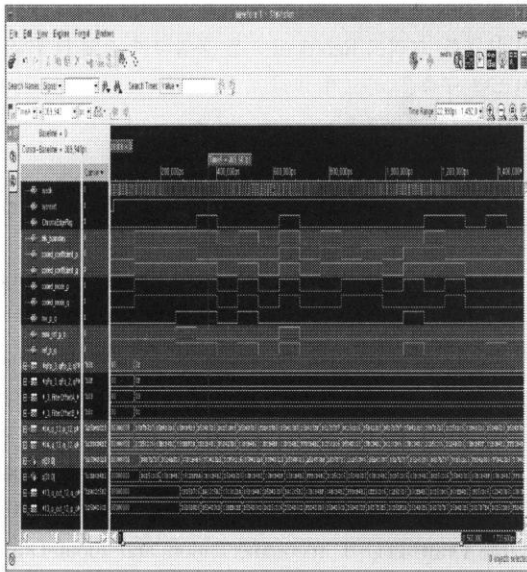


그림 7. 제안한 H.264 디블록킹 필터의 Verilog-HDL 시물레이션 결과

Fig 7. Verilog-HDL simulation result of the proposed H.264 deblocking filter

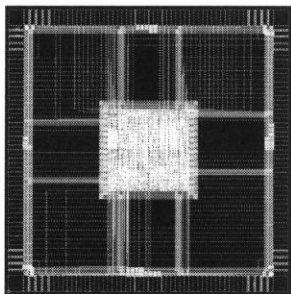


그림 8. 제안한 H.264 디블록킹 필터의 레이아웃

Fig 8. Layout of the proposed H.264 deblocking filter using Apollo tool

## 2. FPGA 구현 및 시물레이션

이 절에서는 설계된 디블록킹 필터를 FPGA장비에 구현하여 정상적으로 동작함을 확인하였다. 사용된 FPGA 장비는 휴인스사의 EDA-PRO 장비와 한백전자의

HBE-Combo가 사용되었으며, FPGA 칩은 Xilinx사의 Virtex2 xc2v4000를 사용하였다. Xilinx ISE 9.1i의 project navigator를 사용하여 합성 후 시물레이션으로 정상적으로 동작함을 확인하였다. 그림 9는 FPGA Device를 이용한 테스트 시스템을 보여준다.

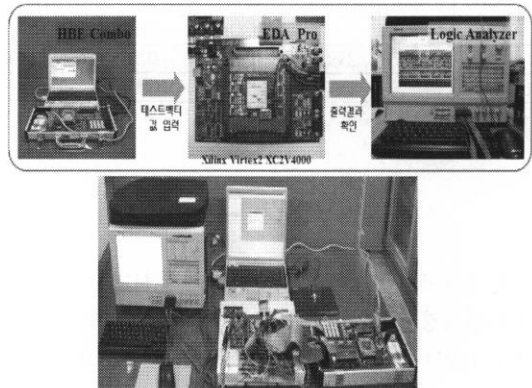


그림 9. FPGA장비를 이용한 테스트 시스템

Fig 9. Test-system using FPGA(EDA\_Pro and HBE-Combo)

표 1. 기존구조와 제안구조와의 핵심 필터링 모듈의 게이트 수 비교

Table 1. The gate count comparison of filtering modules of conventional structure with thoes of proposed structure

구분	기존구조 1	기존구조 2	제안구조
bs=4	5,349	4,257	3,732 (87.66%)
bs=1~3	4,780	3,746	3,403 (90.84%)
total	10,698	8,003	7,135 (89.15%)

표 1은 같은 조건에서 기존 필터링 구조와 제안한 필터링 구조와 비교한 결과를 나타낸다. 표 1에서 보듯이 디블록킹 필터의 strong, standard filtering 블록에 대한 제안 구조의 코딩 후 합성 결과, 기존 구조들과 비교하여 bs = 4일 때, 기존 구조 1과 2에 비해 각각 약 30%와 12.4% 감소하였고 bs = 1~3일 때는 29%와 1.1% 정도 감소 효과를 나타내었다. 전체 모듈로 비교해 보면, 기존 구조 1에 비해 33.31%, 기존 구조 2에 비해 10.85%의 게이트 감소효과를 보여준다.

Dual-port RAM (64x32bit)을 사용하여 하나의 16x16 block이 258 클럭에 동작하고, 최종 출력은 13.5MHz로 P, Q 블록이 출력된다. QCIF, CIF, CCIR 601 사이즈의 화면이 각각 1초당 528.54, 132.13, 33.03 프레임을 처리됨으로

서 실시간으로 처리 가능함을 보여준다. 표 2는 픽셀 사이즈의 크기에 따른 성능평가를 나타낸 것이다.

표 2. QCIF, CIF, and CCIR601형식에 대한 성능 평가  
Table 2. Performance estimations for QCIF, CIF, and CCIR601 image formats

구분	pixel size	Clks/16x16	MBs/frame	sec/frame	frames/sec
QCIF	176x144	258Clks	99MBs	0.001892	528.54
CIF	352x288	258Clks	396MBs	0.007568	132.13
CCIR601	704x576	258Clks	1584MBs	0.030272	33.03

## V. 결 론

본 논문에서는 동영상 압축 표준안인 H.264에서 사용될 디블록킹 필터를 저 전력으로 설계한 구조를 제안하였다. H.264 디블록킹 필터는 핵심 연산 부분인 필터링부, 변수 결정 및 모드 결정부로 구성되어 있으며, 본 논문에서는 기존 논문의 필터링 연산 모듈인 strong, standard filtering 모듈을 각각 12.34%, 9.16%의 게이트를 감소시켜 설계하고, 모드 결정부에서 필터링 조건을 바꾸어서 필터링하지 않는 경우를 미리 추출함으로써 저 전력으로 구현이 가능하였다.<sup>[8]</sup>

전체 디블록킹 필터를 Verilog-HDL로 설계하였으며, 전체 블록의 게이트 수는 Xilinx ISE 9.1i의 project navigator를 이용한 합성 후 게이트 수는 16601개, Synopsys 社의 design compiler를 이용한 합성 후 전체 블록 게이트의 수는 8506의 결과를 확인하였다. 시뮬레이션과 테스트 결과에서 원하는 클럭 주파수인 108MHz에서 정상 동작함을 확인할 수 있었다. 또한 픽셀 사이즈의 크기에 따른 성능 평가로서 QCIF, CIF 그리고 CCIR601 화면 형식에서 각각 1초당 528.54, 132.13, 33.03 프레임이 처리됨으로서 실시간으로 처리 가능함을 보여준다.

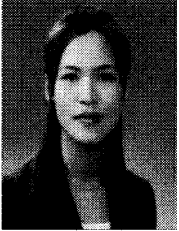
본 논문에서는 제안한 H.264 디블록킹 필터 전체 블록을 FPGA장비에 다운로드 후 Logic Analyzer를 통해 시뮬레이션 결과를 확인하였다. 확인 결과 모든 블록에서 정상 동작함을 알 수 있었다. 각 블록은 기존 모듈에 비해 적은 연산 구조를 가지며, 이는 게이트 감소 효과 및 전력 감소 효과를 가져 온다. 따라서, 저 전력의 멀티미디어 제품에 이용되리라고 생각된다. 향후, 움직임 보상 추정 및 보상, 변환 및 양자화, 역변환 및 역양자화, 가변길이 부호화, 그리고 Reordering 을 통합하여 전체 H.264의 시스템 구축을 하고자 한다.

## 참 고 문 헌

- [1] ITU-T Rec. H.264, "Advanced Video Coding for Generic Audio Visual Services", 2005.
- [2] ITU-T Recommendation H.261, "Video codec for Audiovisual Services at p X 64 kbit/s", March 1993
- [3] ISO/IEC 14496-10:2003, "Coding of Audiovisual Objects-Part10 : Advanced Video Coding", 2003
- [4] Draft ITU-T recommendation and Final Draft Inter-national Standard Of Joint Video Specification (ITU-T Rec. H.264/ISO/IEC 14496-10 AVC), Mar. 2003.
- [5] Shen-Yu Shih 외 2명 "A Near Optimal Deblocking Filter for H.264 Advanced Video Coding" IEEE 2006
- [6] 호요성, H.264 표준의 이해. 두양사 2007
- [7] 호요성, 김승환, H.264/AVC 표준의 소스코드 분석 두양사, 2006
- [8] 장영범, 오세만, 박진수, 한규훈, 김수홍, "H.264용 디블록킹 필터의 저전력 구조" 전자공학회 논문지, 제43권 SP편 제3호 2006. 5

저자 소개

허 정 화(정회원)



- 2001년 : 세명대학교 정보통신학과 학사 졸업
- 2003년 : 세명대학교 전산정보학과 석사 졸업
- 2006년 : 세명대학교 전산정보학과 박사 수료

<주관심분야 : ASIC 설계, 신호처리, Serial ATA ADC/DAC>

박 상 우(정회원)



- 2006년 : 세명대학교 정보통신학과 학사 졸업
- 2006년 : 세명대학교 전산정보학과 석사 졸업

<주관심분야 : ASIC설계, DSP, H.264, FPGA>

박 상 봉(정회원)



- 1985년 : 광운대 전자재료공학과 학사 졸업
- 1987년 : 고려대 전자공학 석사 졸업
- 1992년 : 고려대 전자공학과 공학박사 졸업
- 1992년 ~ 1999년 : 삼성전자 선임연구원

- 1999년 ~ 현재 : 세명대학교 정보통신학과 부교수
- 2000년 ~ 현재 : 읊니미디어(주) ASIC팀 기술고문
- 2000년 ~ 현재 : @lab(주) Digital 설계팀 기술고문

<주관심분야 : Digital TV, Serial ATA>