고성능 HEVC 복호기를 위한 효과적인 In-loop Filter 하드웨어 설계

박승용*, 조현표*, 박재하*, 강병익**, 류광기*
*한밭대학교 정보통신전문대학원
**건양대학교 의료IT공학과
e-mail: {srrr.kr, chohyunpyo77, parkjeaha1}@gmail.com,
bikang@konyang.ac.kr, kkryoo@hanbat.ac.kr

The Hardware Design of Effective In-loop Filter for High Performance HEVC Decoder

Seungyong Park*, Hyunpyo Cho*, Jaeha Park*, Byungik Kang**, and Kwangki Ryoo*
*Graduate School of Information and Communication, Hanbat National University
**Medical IT Engineering, Konyang University

요 약

본 논문에서는 고성능 HEVC(High Efficiency Video Coding) 복호기 설계를 위한 효율적인 in-loop filter의 하드웨어 구조 설계에 대해 기술한다. in-loop filter는 deblocking filter와 SAO로 구성되며, 블록 단위 영상 압축 및 양자화 등에서 발생하는 정보의 손실을 보상하는 기술이다. 하지만 HEVC는 64x64 블록 크기까지 화소 단위 연산을 수행하기 때문에 높은 연산시간 및 연산량이 요구된다. 따라서 본 논문에서 제안하는 in-loop filter의 deblocking filter 모듈과 SAO 모듈은 최소 연산 단위인 8x8 블록 연산기로 구성하여 하드웨어 면적을 최소화하였다. 또한 SAO에서는 8x8 블록의 연산 결과를 내부레지스터에 저장하는 구조로 64x64 블록 크기를 지원하도록 설계하여 연산시간 및 연산량을 최소화 하였다. 제안하는 하드웨어 구조는 Verilog HDL로 설계하였으며, TSMC 칩 공정 180nm 셀 라이브러리로 합성한 결과 동작 주파수는 270MHz이고, 전체 게이트 수는 48.9k 이다.

1. 서론

최근 디지털 방송기술과 디스플레이 기기 등의 발전으로 초고해상도 비디오 품질에 대한 소비자의 요구가 꾸준히 증가하고 있으며, 지상파 방송사에서도 2012년 10월부터 3개월간 4K급 초고해상도 영상에 대한 실험방송을 실시하였다. 이에 따라 기존의 영상 크기를 넘어선 4K~8K급 초고해상도 영상에 대한 필요성이 대두되고 있다[1,2]. 그러나 기존의 영상 압축 표준인 H.264/AVC로 초고해상도 영상의 데이터를 저장하거나 현재의 통신채널로 전송하기에는 상당한 비용이 발생한다. 이에 따라 국제 비디오 부호화 표준화 단체인 JCT-VC(Joint Collaborative Team on Video Coding)는 2010년 4월 독일 Dresden에서 개최된 표준화 회의를 시작으로 차세대 영상 압축 표준인 HEVC(High Efficiency Video Coding) 표준 제정 프로젝트를 시작하였으며, 2013년 1월에 HEVC 표준화를 완료하였다. HEVC 표준은 기존의 H.264/AVC 표준과 같은 움직임 보상 변환부호화 압축 기술에 기반을 하며, 보다 향상된 다양한 요소 기술을 적용함으로써 압축 효율을 극대화하고 있다[3].

H.264/AVC 표준은 in-loop filter 기술에서 영상의 블록화 현상을 제거하기 위한 deblocking filter만을 포함하였지만, HEVC에서는 deblocking filter, 양자화 등의 손실 압축에서 발생하는 정보의 손실을 보상하기 위한 SAO(Sample Adaptive Offset) 및 ALF(Adaptive Loop Filter)와 같은 새로운 기술을 추가적으로 포함하여 주관적 화질 뿐만 아니라 압축 효율을 향상시키는 구조를 가지고 있다. HEVC의 in-loop filter에서 deblocking filter 기술은 블록 기반 영상압축 시스템에서 발생하는 영상의 블록화 현상을 제거하고, SAO 기술은 영상 압축 과정에서 발생하는 원본 영상과 복원 영상 간의 왜곡을 화소 단위의 offset을 통해 보상한다. HEVC의 in-loop filter는 deblocking filter와 SAO를 통해 주관적 화질 및 부호화 효율을 같이 향상시키는 기능을 수행한다. 하지만 HEVC에서 처리하는 최대 블록 크기는 64x64로 화소 단위 연산을 수행하는 in-loop filter에서는 높은 연산량과 연산시간을 차지한다[4].

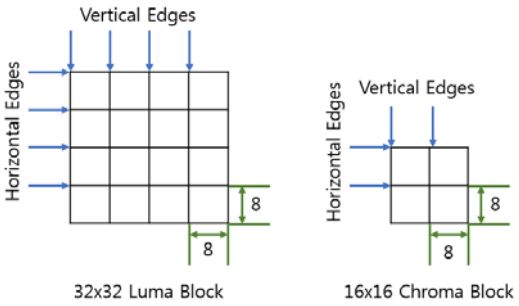
본 논문에서는 in-loop filter의 높은 연산량과 연산시간, 하드웨어 설계시 면적을 최소화할 수 있는 하드웨어 구조를 제안한다. 제안하는 하드웨어 구조는 deblocking filter와 SAO의 연산 블록 단위를 8x8 연산기 구조로 구성하였다. 또한 SAO는 64x64 블록 크기를 8x8 연산기 구조로 지원하기 위해 내부 레지스터를 이용하여 화소를 저장하는 구조로 구현하였다. 제안하는 deblocking filter 하드웨어 구조는 16x16 블록을 입력 받아 8x8 블록 크기로 블록화 현상을 제거하여 출력한다. 제안하는 SAO 하드웨어 구조는 deblocking filter에서 출력되는 8x8 블록 크기

를 입력받아 offset type을 연산하여 offset을 적용시킨다. 그 결과 in-loop filter 내부 모듈인 deblocking filter와 SAO의 8x8 연산기 단위 파이프라인 구조 설계로 하드웨어 면적 및 연산시간을 최소화 하였다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 in-loop filter의 기법을 기술하며, 3장에서는 제안하는 in-loop filter 하드웨어 구조를 기술한다. 4장에서는 하드웨어 합성 결과를 기술하며, 마지막으로 5장에서는 결론으로 끝을 맺는다.

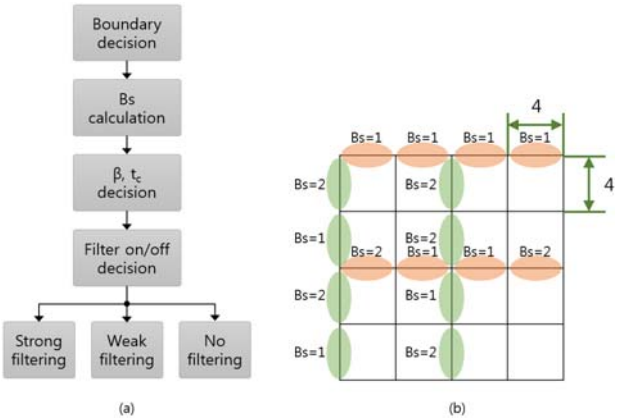
2. In-loop filter

HEVC main profile의 in-loop filter는 deblocking filter와 SAO로 구성된다. Deblocking filter는 블록 단위 영상 압축으로 생기는 블록화 현상을 보상하는 기능을 수행한다. 그림 1은 deblocking filter의 필터링 유닛이며, 최소 필터링 유닛은 8x8 블록이다.



(그림 1) Deblocking filter의 필터링 유닛

그림2의 (a)는 deblocking filter의 필터링 과정을 나타낸다. 그림 2의 (b)는 블록 경계의 특성에 따라 적응적인 강도를 구별하기 위해 Bs(Boundary strength) 단위를 나타낸다. Bs는 8x8 블록 경계에서 4개의 화소 단위로 적용한다.



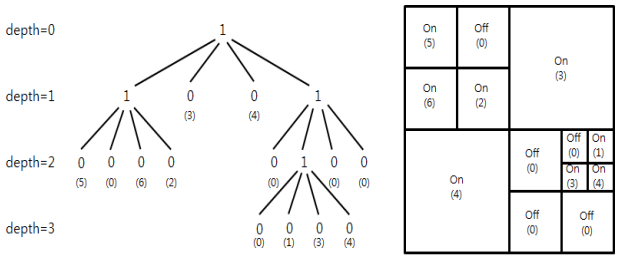
(그림 2) HEVC Deblocking filter의 알고리즘
(a) 필터링 과정 (b) Bs 단위

SAO는 양자화 등의 부호화 과정을 통해 발생하는 원본 영상과 복원 영상간의 왜곡을 식(1)과 같이 화소(Sample) 단위로 offset을 구하며, 식(2)와 같이 부/복호화기가 동일한 방법으로 복원 화소의 왜곡을 보상하는 기능을 수행한다.

$$offset = \frac{\sum_{n=0}^{N-1}(rc_n - rec_n)}{N} \quad (1)$$

$$rec'_n = rec_n + offset \ \ where \ \ n = 0,1,\cdots,N-1 \quad (2)$$

HEVC의 SAO는 그림 3과 같이 quadtree 기반의 CU 분할 방법과 동일한 방식으로 LCU(Largest CU)블록을 0-depth에서 3-depth까지 하위 블록으로 분할하여, 각 하위 블록마다 다른 SAO type이 적용될 수 있다.



(그림 3) SAO 블록분할 구조 및 SAO type 적용 예

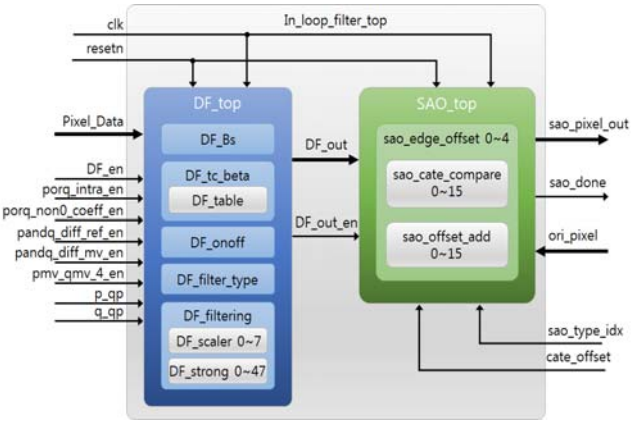
분할된 각 하위 블록에 적용 가능한 SAO type은 표 1과 같으며, SAO를 적용하지 않는 방법, EO(Edge Offset) 및 BO(Band Offset)으로 구분할 수 있다. EO 및 BO은 해당 화소의 카테고리를 구분하기 위한 방법에 따라 EO은 0-degree, 90-degree, 135-degree, 45-degree 총 4가지 방향으로 구분되며, BO은 화소 밝기 값을 이용한 방법으로 전체 화소 값을 32개의 구간으로 분류한 뒤, 각각의 밴드에 해당하는 오프셋을 결정한다. 이 중에서 율-왜곡 측면에서 최적인 연속된 4개의 오프셋을 결정하여 그 위치를 가리키는 band_position과 사용된 4개의 오프셋을 전송한다.

<표 1> SAO type 정보

sao_type_idx	SAO type to used		Number of categories
0	None		0
1	Edge Offset	0-degree	4
2		90-degree	4
3		135-degree	4
4		45-degree	4
5	Band Offset	4 consecutive offset	4

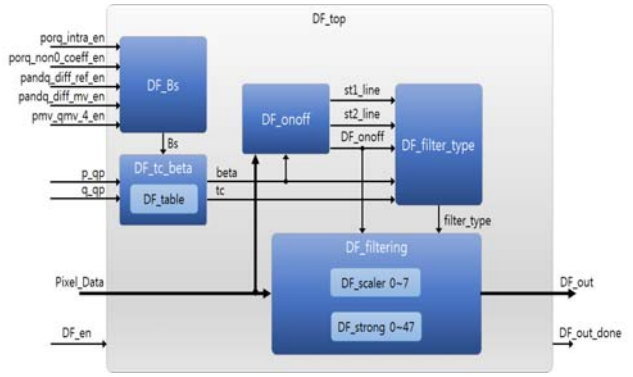
3. 제안하는 in-loop filter 하드웨어 구조

본 논문에서 제안하는 in-loop filter 하드웨어 구조는 그림 4와 같다. 제안하는 in-loop filter는 deblocking filter 기능을 수행하는 DF_top 모듈과 SAO 기능을 수행하는 SAO_top 모듈로 구성된다.



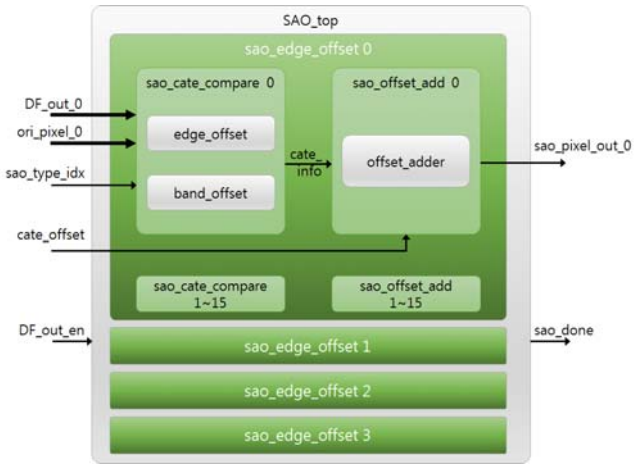
(그림 4) 제안하는 in-loop filter 하드웨어 구조

제안하는 in-loop filter는 deblocking filter와 SAO에서 처리되는 최소 블록 크기를 고려하여 8x8 블록 크기의 연산기 구조로 설계하였다. 또한, 8x8 블록 단위로 deblocking filter와 SAO의 파이프라인 처리가 가능하다. SAO는 LCU인 64x64 블록 크기까지 지원하기 위해 내부 레지스터를 이용하여 상위 블록 연산에 필요한 화소를 저장하는 구조로 설계하였다. 그림 5는 제안하는 deblocking filter 하드웨어 구조를 나타낸다. DF_top 모듈은 16x16 블록 크기의 화소값을 Pixel Data 신호로 입력 받는다. 또한, 필터링의 세기를 결정하기위한 제어신호를 입력받으며, 필터링된 8x8 블록 크기를 DF_out 신호를 통해 SAO_top 모듈로 출력하는 기능을 수행한다. DF_top 모듈은 DF_Bs, DF_tc_beta, DF_onoff, DF_filter_type, DF_filtering 하위 모듈로 구성된다.



(그림 5) 제안하는 deblocking filter 하드웨어 구조

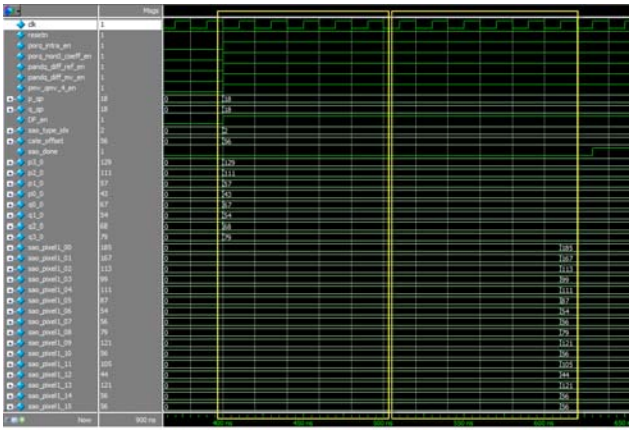
DF_Bs 모듈은 필터링의 강도인 Bs를 결정하는 기능을 수행하며, DF_tc_beta 모듈은 QP 값으로 필터링에 필요한 tc, β 값을 선택하여 출력하는 기능을 수행한다. DF_onoff 모듈은 필터링의 적용 여부를 판별하는 기능을 수행하며, DF_filter_type 모듈은 strong filter와 week filter를 선택하는 기능을 수행한다. DF_filtering 모듈은 deblocking filter의 필터링을 수행한다. 그림 6은 제안하는 SAO 하드웨어 구조를 나타낸다. SAO_top 모듈은 원본 영상 데이터인 ori_pixel 신호와 8x8 블록 크기의 필터링된 영상 데이터를 DF_out_0 신호로부터 입력받는다. 또한, SAO_top 모듈의 시작과 유효한 DF_out 신호를 나타내는 DF_out_en을 입력받는다. SAO_top 모듈은 SAO_type_idx 신호와 cate_offset 신호를 입력받아 오프셋을 적용하며, sao_pixel_out 신호로 오프셋이 적용된 화소 데이터를 출력한다. 또한, sao_done 신호를 통해 SAO_top 모듈의 종료를 출력한다. SAO_top 모듈은 sao_edge_offset 모듈 4개로 구성되며, sao_edge_offset 모듈은 sao_cate_compare 모듈 16개와 sao_offset_add 모듈 16개로 구성된다.



(그림 6) 제안하는 SAO 하드웨어 구조

sao_edge_offset 모듈은 4x4 블록 크기에 해당하는 연산을 수행하며, sao_cate_compare 모듈을 통해 SAO의 카테고리를 결정한다. sao_offset_add 모듈은 sao_cate_compare 모듈에서 정해진 SAO의 카테고리에 따라 오프셋을 덧셈하여 출력한다.

본 논문에서는 HEVC 표준 참조 소프트웨어인 HM-10.0rc1[5]에서 데이터를 추출하였고, 제안한 하드웨어 구조를 통해 시뮬레이션 한 결과 참조 소프트웨어에서 추출한 데이터와 비교하여 정상적으로 동작함을 확인하였다. 그림 7은 제안하는 in-loop filter의 시뮬레이션 결과이다. DF_top 모듈과 SAO_top 모듈에서 각각 5 사이클 소요되며, in-loop filter는 파이프라인 구조로 구성된다. 제안하는 하드웨어 구조는 LCU를 처리하는데 325 사이클이 소요된다. 이는 in-loop filter 하드웨어 구조에 파이프라인을 적용하지 않은 경우 640 사이클이 소요되는 것에 비해 약 2배의 수행 사이클을 감소시켰다.



(그림 7) 제안하는 in-loop filter 시뮬레이션 결과

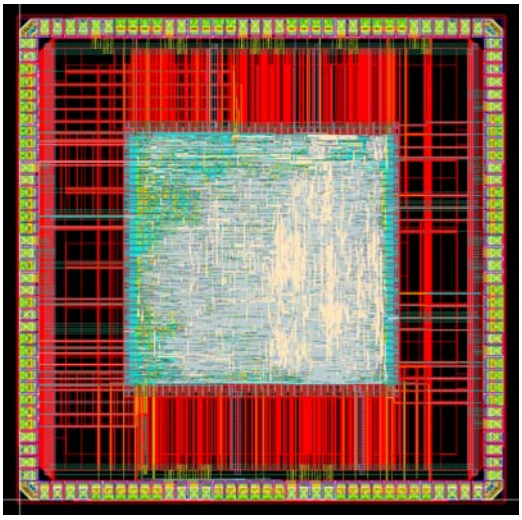
4. 하드웨어 합성 결과

본 논문에서 제안하는 in-loop filter 하드웨어 구조는 Verilog HDL로 설계하였다. 하드웨어 합성은 IDEC에서 지원하는 CAD tool을 사용하였으며, TSMC 180nm 셀 라이브러리로 합성하였다. 표 2는 제안하는 in-loop filter 하드웨어 구조의 합성 결과를 나타낸다.

<표 2> 제안하는 in-loop filter 하드웨어 합성 결과

구분	결과
공정	TSMC 180nm
동작 주파수	270MHz
LCU 당 인코딩 사이클 수	325
게이트 (k)	48.9k

그림 8은 Virtuoso Layout Editing 칩 설계 결과 화면이다.



(그림 8) Virtuoso Layout Editing 칩 설계 결과 화면

5. 결론

본 논문에서는 HEVC in-loop filter의 연산시간과 연산량, 하드웨어 면적을 최소화하기 위해 8x8 블록 연산기를 사용하였고, SAO에서는 LCU인 64x64 블록 크기를 지원하기 위해 내부 레지스터를 사용하여 8x8 블록 경계의 화소들을 저장하는 방식을 채택하였다. 또한, deblocking filter와 SAO에서 파이프라인 구조로 설계하기 위해 8x8 블록 연산기를 사용하여 연산 시간을 최소화하였다.

제안하는 하드웨어 구조를 TSMC 180nm 셀 라이브러리로 합성한 결과 최대 동작 주파수는 270MHz이고, 게이트 수는 48.9k이다. 또한, 하나의 LCU를 처리하는데 325 사이클이 소요된다.

감사의 글

본 논문은 교육부와 한국연구재단의 지역혁신인력양성사업 및 미래창조과학부 출연금으로 수행한 ETRI SW-SoC 융합 R&BD 센터와의 공동 연구의 결과입니다.

참고문헌

[1] Sung-Ho Bae, Jaeil Kim, Munchurl Kim, Sukhee Cho, and Jin Soo Choi, "Assessments of Subjective Video Quality on HEVC-Encoded 4K-UHD Video for Beyond-HDTV Broadcasting Services," IEEE Trans. Broadcast., Vol. 59, Issue 2, pp. 209-222, Jun. 2013.

[2] 임중곤, 김병선, 함상진, 전성호, "지상파 4K UHDTV 제1차 실험방송 결과와 향후 추진 계획," 방송공학회지, 제18권, 제2호, pp. 7-26, 2013.

[3] G. J. Sullivan, J. R. Ohm, W. J. Han and T. Wiegand, "Overview of the High Efficiency Video Coding(HEVC) Standard," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, Dec. 2012.

[4] B. Bross, W.-J. Han, G. J. Sullivan, J.-R. Ohm, and T. Wiegand, "High efficiency video coding (HEVC) text specification draft 8," Document of Joint Collaborative Team on Video Coding, JCTVC-J1003, July, 2012.

[5] "HM10.1: High efficiency video coding HEVC test model 10.1." [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/HM-10.1rc1/