

249 Mpixels/sec 하드웨어 HEVC 디코더의 가변 크기 블록 움직임 보상 모듈 구현

조승현, 변경진, 엄낙웅

한국전자통신연구원

shcho@etri.re.kr, kjbyun@etri.re.kr, nweum@etri.re.kr

Implementation of a Variable-sized Block Motion Compensation Module for 249-Mpixels/sec Hardware HEVC Decoders

Seunghyun Cho Kyungjin Byun Nak-Woong Eum

Electronics and Telecommunications Research Institute, Daejeon, Korea

요 약

본 논문에서는 하드웨어 HEVC 디코더의 움직임 보상 모듈의 구조를 제안한다. 제안된 구조를 갖는 움직임 보상 모듈은 하드웨어 처리 싸이클 수와 내부메모리 크기를 감소시키기 위해 하나의 코딩 유닛을 그보다 작은 여러 개의 블록으로 분할하여 처리할 수 있다. 제안된 움직임 보상 구조는 캐시를 통해 외부 메모리에 접근하여 참조 픽처를 로딩하는 단계와 보간 필터를 거쳐 예측 샘플을 생성하는 단계로 내부-파이프라인을 구성하며 코딩 유닛의 크기에 따라 내부-파이프라인에서 처리할 블록의 크기를 결정한다. 본 논문에서는 코딩 유닛 분할의 기준이 되는 블록 크기를 결정하기 위한 절충사항에 대해서도 논의한다. 제안된 구조의 효율성을 판단하기 위해 구현된 움직임 보상 모듈을 RTL 시뮬레이션 및 FPGA 보드 검증을 통해 테스트 하였으며, SoC 로 제작될 경우 초당 249 Mpixel 을 처리하여 4K-UHD 시퀀스의 실시간 디코딩이 가능한 것으로 판단되었다.

1. 서론

ISO/IEC MPEG 과 ITU-T VCEG 이 공동으로 구성된 JCT-VC 의해 제정된 국제 비디오 압축 규격인 HEVC [1]는 기존의 H.264/AVC 에 비해 압축효율이 두 배에 달해 차세대 비디오 표준기술로 주목 받고 있다 [2]. 압축효율의 개선을 위해 HEVC 에는 코딩 구조의 큰 변화가 있었는데, 코딩 트리 유닛(Coding Tree Unit, CTU) 내부에 쿼드트리 형태로 분할이 가능한 코딩 유닛(Coding Unit, CU)을 두고 이를 다시 예측 유닛(Prediction Unit, PU)과 변환 유닛(Transform Unit, TU)으로 각각 분할할 수 있도록 한 것이다. 이러한 코딩 구조의 변화에 따라 HEVC 에서는 기존 보다 큰 크기의 블록에 대한 예측 및 변환을 유연하게 지원할 수 있게 되었다.

일반적으로, 전용 하드웨어로 구현한 비디오 디코더는 범용 CPU 상에서 소프트웨어로 구현한 디코더에 비해 처리성능뿐만 아니라 전력소모 면에서도 유리하다. 따라서, 4K/8K-UHD 영상과 같은 대용량 비디오 데이터의 실시간 처리를 위해 전용 하드웨어 기반의 디코더 구현과 관련된 연구가 활발히 진행되고 있다 [3]~[6]. 하드웨어 비디오 디코더를 구성하는 모듈 중 움직임 보상(Motion Compensation, MC) 모듈은 전체 디코더에서 요구되는 대부분의 메모리 대역폭의 대부분을

차지하며, 외부 메모리 접근에 많은 싸이클이 소요된다. 이 때문에 MC 모듈의 설계에는 빠른 접근이 가능한 비디오 데이터 캐시를 구현하기 위한 노력들 [4]~[5]이 필요하다. 그러나, 캐시를 구현한 경우에도 예측 블록의 크기 및 움직임 벡터의 크기와 방향에 따라 MC 에 소요되는 싸이클 수는 블록 별로 큰 차이가 나게 되며, 추가적으로 캐시와 보간 필터 간 참조 샘플 전달을 위해 상당히 큰 버퍼가 필요하게 된다. 특히, 다양한 크기의 CU 및 PU 를 지원하는 HEVC 의 경우, 이러한 문제점이 가중되어 하드웨어 디코더의 성능 및 비용 측면에서 걸림돌이 될 수 있다. 본 논문에서는 CTU 단위로 병렬처리를 하는 하드웨어 HEVC 디코더에서 전술한 문제점들을 해결하기 위한 방법을 제안하고 실험을 통해 효율성을 확인한다.

본 논문의 구성은 다음과 같다. 2 절에서는 가변 크기 블록 MC 모듈의 구조와 동작방식에 대해 살펴본 후, 3 절에서는 제안된 구조의 성능을 실험을 통해서 확인한다. 마지막으로 4 절에서는 본 논문에 대한 결론을 맺는다.

2. 가변 크기 블록 움직임 보상 모듈의 구조와 동작방식

하드웨어 기반의 비디오 디코더는 일반적으로 디코딩의 각 과정을 모듈 별로 구현하여 파이프라인을 구성하고 기존 표준의 매크로블록과 같이 코딩 구조의 기본 단위가 되는 블록을 각 모듈에서 병렬로 처리함으로써 태스크 수준의

본 연구는 산업통상자원부 및 한국산업기술평가위원회의 산업융합 원천기술개발사업(정보통신)의 일환으로 수행하였음. [10039214, 초고해상도 비디오 코덱 SoC]

병렬성을 이용한 처리속도 향상을 얻는다. 본 절에서는 CTU 단위로 병렬 처리하는 하드웨어 HEVC 디코더를 구성하기 위한 MC 모듈의 구조 및 동작 방식에 대해 설명한다.

MC 모듈에서 다양한 크기의 PU 처리에 따른 처리 사이클 및 내부 버퍼의 증가를 억제하기 위해서는 내부 파이프라인 구성이 필수적이다. 그림 1에 제안하는 가변 크기 블록 MC 모듈의 구조와 외부 연결을 나타냈다. 제안된 구조는 MC

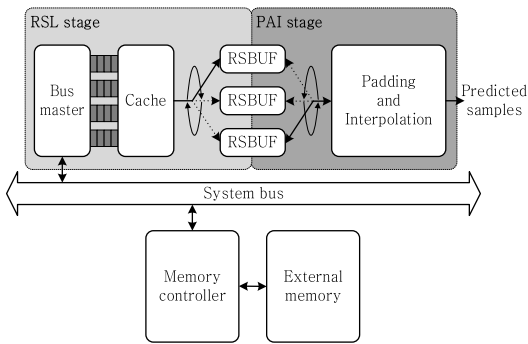


그림 1. 가변 크기 블록 처리 MC 모듈의 구조와 외부 연결

모듈을 참조 샘플을 읽는 RSL 단계와 패딩 및 보간을 처리하는 PAI 단계로 구분하여 내부-파이프라인화 하고 이들 사이에 참조 샘플을 전달하기 위한 복수개의 RSBUF를 두었다. RSL 단계에는 4-way set associative 캐시 [4]~[5]를 구현하였으며, PAI 단계에는 휘도 샘플과 색차 샘플 보간의 공유가 가능한 단일 구조 필터 [6]를 수평과 수직 방향으로 각각 네 개씩 구현하였다. 제안하는 MC 모듈은 입력된 한 CTU 내 CU를 차례로 방문하여 PU들을 처리하는데, PU 크기의 편차에 따른 각 내부-파이프라인 단계의 처리지연을 최소화하기 위해 PU를 다시 MxM 크기의 MC 처리 유닛(Processing Unit for MC, PUMC)을 기준으로 분할처리한다. PU의 크기를 WxH라고 할 때, W, H가 모두 M과 같거나 작은 경우 각 내부-파이프라인 단계를 통해 처리되는 MC 처리 블록(Processing Block for MC, PBMC)의 크기는 그대로 WxH가 된다. 그러나, W와 H 중 어느 하나라도 M을 초과하는 경우에는 PBMC 크기가 PUMC 크기 이하가 되도록 분할하게 된다. 그림 2에 16x16 PUMC를 사용할 경우 32x32 CU의 PartMode에 대한 PBMC 분할의 예를 나타냈다. 그림 2에서 굵은 실선은 PartMode에 따른 CU 내 PU 분할을, 얇은 점선은 16x16 PUMC를 나타낸다. 그림 2-(a)의 32x32 PU는 네 개의 16x16 PBMC로 분할하여 처리하고 그림 2-(b)의 두 16x32 PU는 각각 16x16 PBMC로 분할하여 처리하며 그림 2-(c)의 8x32 PU는 두 개의 8x16 PBMC ①, ④로 24x32 PU는 두 개의 8x16 PBMC ②, ⑤와 두 개의 16x16 PBMC ③, ⑥으로 각각 분할하여 처리한다.

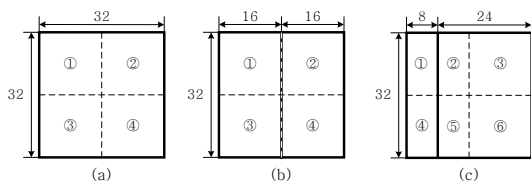


그림 2. 16x16 PUMC를 사용할 경우, 32x32 CU의 PartMode에 대한 PBMC 분할의 예: (a) PART₂Nx2N; (b) PART_Nx2N; (c) PART_nLx2N.

제안된 MC 모듈의 구조에서 PUMC의 크기가 클 수록

RSBUF로 로딩되는 샘플의 중첩이 줄어들고 PAI 단계의 연속적인 처리가 늘어나 전체 MC 처리 사이클 수는 줄어들지만, RSBUF의 크기는 비례적으로 증가한다. 반대로 PUMC의 크기가 작을수록 MC 처리 사이클 수는 증가하는 반면 RSBUF의 크기는 감소한다. 따라서, MC 처리 사이클 수와 RSBUF의 크기 사이에 절충이 필요하며 이를 결정하기 위한 실험의 결과에 대해 다음 절에서 논의한다.

3. 실험결과

제안된 구조를 갖는 가변 크기 블록 MC 모듈을 Verilog HDL을 사용하여 설계하였으며, 효율성을 검토하기 위해 RTL 시뮬레이션과 FPGA 보드를 통한 실험을 실시하였다. 실험은 32x32, 16x16, 8x8 PUMC 크기에 대해 각각 수행 되었으며 HM-13.0 [7]의 입의 접근(random access) 설정을 사용하여 인코딩한 비트스트림을 사용하였다. 단, 64x64 PUMC를 사용하는 것은 하드웨어 HEVC 디코더에 파이프라인 한 단계를 추가하는 것과 같이 실험에서 제외 하였다. 각 크기의 PUMC에 대응되는 RSBUF는 단일포트 SRAM으로 구현하였는데, 두 개의 RSBUF만을 사용하는 경우에도 세 개의 RSBUF를 사용한 경우와 비슷한 경향성이 확인되었으나, 각 PUMC 크기에 대해 높은 평균 MC 처리 사이클 수를 보였다. 그림 3은 PUMC 크기에 따른 비교실험의 결과를 보이고 있다. 그림 3-(a)에 *Kimono*, *ParkScene*, *Cactus*, *BasketballDrive*, *BQTerrace* 영상에서 측정된 평균 MC 처리 사이클 수를 나타냈다. 8x8 PUMC를 사용하는 경우에 비해 16x16과 32x32 PUMC는 각각 30.5%와 38.8%의 평균 MC 처리 사이클 수의 감소효과가 있었다. 그림 3-(b)에는 RSBUF의 크기를 나타냈는데, 32x32 PUMC를 사용하는 경우에 비해 16x16과 8x8 PUMC는 각각 52.2%와 77.4%의 RSBUF 크기를 절약할 수 있었다.

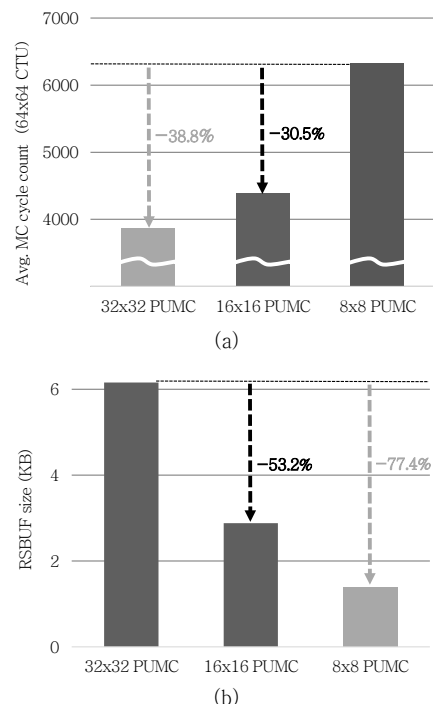


그림 3. PUMC 크기에 따른 비교실험의 결과: (a) 평균 MC 처리 사이클 수; (b) RSBUF 크기.

그림 3 에 보인 실험결과를 통해 본 연구에서는 16×16 PUMC 가 제안된 구조를 갖는 MC 모듈의 성능과 하드웨어 비용간의 절충을 위한 최적의 PUMC 크기라는 결론에 도달하였다. 구현된 가변 크기 블록 MC 모듈은 약 270 MHz 의 동작 주파수에서 249 Mpixels/sec 의 처리성능을 보였으며, HEVC 로 압축된 4K-UHD 영상의 실시간 처리가 가능한 것으로 판단되었다. 아울러, 본 논문에서는 MC 모듈의 내부만을 조명하였으나, 하드웨어 HEVC 구현 시 CTU 별 MC 처리 사이클 수의 편차를 줄이기 위해서는 디코더 파이프라인상에서 MC 모듈의 입력과 출력에 추가적인 버퍼를 삽입하는 방법이 사용될 수 있다.

4. 결론

본 논문에서는 CTU 단위로 병렬처리를 하는 하드웨어 HEVC 디코더를 구성하기 위한 가변 크기 블록 MC 모듈의 구조가 제안되었다. 제안된 구조는 MC 과정을 두 단계의 내부-파이프라인으로 구분하며 각 단계에서 처리되는 블록의 최대 크기에 제한을 두고 가변 하도록 함으로써 MC 모듈의 처리 성능과 하드웨어 비용을 효과적으로 절충할 수 있다. 제안된 구조로 구현한 가변 크기 블록 MC 모듈은 실험을 통해 4K-UHD 영상의 실시간 처리가 가능한 것으로 판단되었다.

참고문헌

- [1] B. Bross, W.-J. Han, G. J. Sullivan, J.-R. Ohm, and T. Wiegand, *High Efficiency Video Coding (HEVC) Text Specification Draft 10*, document JCTVC-L1003, ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC), Mar. 2013.
- [2] G. J. Sullivan, J.-R. Ohm, W.-J. Han and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Systems for Video Technol.*, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.
- [3] M. Tikekar, C.-T. Huang, C. Juvekar, V. Sze, and A. Chandrakasan, "A 249-Mpixel/s HEVC Video-Decoder Chip for 4K Ultra-HD Applications," *IEEE Journal of Solid-State Circuits*, vol. 49, Jan. 2014.
- [4] S. Wang, D. Zhou, and S. Goto, "Motion compensation architecture for 8K UHDTV HEVC decoder," *Proc. ICME 2014*, pp. 1-6, July 2014.
- [5] J. Zhou, D. Zhou, G. He, and S. Goto, "Cache based motion compensation architecture for quad-HD H.264/AVC video decoder," *IEICE Trans. Electron.*, Vol. E94-C, No. 4, pp. 439-447, Apr. 2011.
- [6] Z. Guo, D. Zhou, S. Goto, "An optimized MC interpolation architecture for HEVC," *Proc. ICASSP*, 2012, pp. 1117-1120.
- [7] HM Reference Software 13.0 [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware