논문 16-41-11-25 The Journal of Korean Institute of Communications and Information Sciences '16-11 Vol.41 No.11
http://dx.doi.org/10.7840/kics.2016.41.11.1502

HEVC 용 고속 인트라 예측 VLSI 구현

조 현 수◆, 홍 유 표°, 장 한 별*

High-Speed Intra Prediction VLSI Implementation for HEVC

Hyeonsu Jo◆, Youpyo Hong°, Hanbeyoul Jang*

요 약

HEVC (High Efficiency Video Coding)는 최근에 제안된 비디오 압축 표준으로서 이전의 비디오 압축 표준보다 두 배 이상의 부호화 효율을 가진다. 다양한 종류의 인트라 예측 블록과 모드는 HEVC의 높은 압축 성능과 연산 복잡도 증가의 주요 요인이다. 본 논문은 파이프라인과 인터리빙 기술을 사용하여 하드웨어 자원의 요구조건을 줄이는 반면 효율과 성능은 향상시킨 HEVC 용 인트라 예측 하드웨어 구조를 제시한다.

Key Words : HEVC, Intra prediction, Hardware, Architecture, VLSI, Pipeline

ABSTRACT

HEVC (High Efficiency Video Coding) is a recently proposed video compression standard that has a two times greater coding efficiency than previous video compression standards. The key factors of high compression performance and increasement of computational complexity are the various types of block partitions and modes of intra prediction in HEVC. This paper presents an intra prediction hardware architecture for HEVC utilizing pipelining and interleaving techniques to increase the efficiency and performance while reducing the requirement for hardware resources.

Ⅰ. 서 론

HEVC 인트라 예측은 현재 예측 단위의 원본 픽셀과 주변에 디코딩 된 예측 단위의 참조 픽셀간의 차이를 사용하는 코딩 툴이다. 이러한 주변 블록 간 데이터 의존성은 H.264/AVC 인트라 예측 성능의 장애 요인이었고, HEVC 인트라 예측에서 또한 중요한 디자인 이슈이다. 더욱이 HEVC 인트라 예측은 H.264/AVC와 같은 이전 표준에 비해 다양한 블록 사이즈와 모드를 처리하도록 확장되었다.

종래에는 인트라 예측의 복잡 도를 줄이기 위한 방법으로 이미지 손상을 감수하고 인트라 예측 모드를 줄이거나[1] 트랜스폼의 연산을 간단히 하는 방법을 사용하였다[2]. 최근에는 연구 주제로서 효과적인 HEVC 용 인트라 예측의 하드웨어 구현이 활성화 되고 있다. Fu[3]는 처리 속도와 참조 픽셀을 저장하기 위한 레지스터의 개수를 줄이기 위해 유연한 참조 샘플 선택 기술을 제안했다. Zhenyu[4]는 재구성 가능한 인트라 예측 구조를 위해 가로와 세로 예측에 대한 주변 참조 픽셀의 배열 매핑 구조를 제시하였다. Daniel[5]은 처음으로 모든 예측 단위가 적용된 HEVC 인트라 예측 설계를 제시하였다. 이 연구에서는 처리량을 향상시키기 위해 가로와 세로 모드에 대해 두 독립적인 예측 로직을 사용하였다. Jung[6]은 IDCT와 IDST를 통합한

※ 본 연구는 2016 년도 동국대학교 논문게재장려금 지원으로 이루어졌음.
◆ First Author : Dongguk University Department of Electronic & Electrical Engineering, newbie5136@dongguk.edu, 정회원
° Corresponding Author : Dongguk University Department of Electronic & Electrical Engineering, yhong@dongguk.edu, 종신회원
* Dongguk University Department of Electronic & Electrical Engineering, hjang@dongguk.edu
논문번호 : KICS2016-08-217, Received August 31, 2016; Revised October 10, 2016; Accepted October 10, 2016

구조를 제안하여 하드웨어의 크기를 줄였다.

Andrzej[7]는 4x4 인트라 예측과 나머지 인트라 예측을 위한 듀얼 패스 인트라 예측 구조를 도입하였다. Dam[8]은 블록 간 데이터 의존성 문제를 해결하기 위해 화질의 손실을 감수하고 참조 픽셀로서 재구성된 픽셀 대신 원본 픽셀을 사용하였다.

본 논문에서는 높은 하드웨어 활용도와 성능을 가진 새로운 듀얼 패스 인트라 예측 구조를 제안한다. 전체 인트라 예측 알고리즘의 효과적인 하드웨어 구현을 위해 파이프라인과 블록 간 인터리빙을 사용하였다.

Ⅱ. 제안된 인트라 예측 및 트랜스폼 구조

이전 영상 압축 표준의 인트라 예측을 위한 예측 단위가 4x4, 8x8, 16x16 블록으로 정의된 반면, HEVC의 인트라 예측을 위한 예측 단위는 4x4, 8x8, 16x16, 32x32, 64x64 블록으로 정의되는데 이것은 HEVC 인트라 예측 복잡도의 현저한 증가를 의미한다. 참조 소프트웨어인 HM을 사용하여 일반적인 비디오 영상들을 실험한 결과에 의하면 LCU (Largest Coding Unit)가 32x32 블록인 경우와 64x64 블록인 경우 압축률과 이미지 화질의 큰 변화가 없기 때문에 본 연구에서는 4x4, 8x8, 16x16, 32x32 블록의 인트라 예측 구현을 목표로 한다. 구현에 있어서 최적의 인트라 예측 모드를 찾기 위해 예측 비용의 계산은 인코더 하드웨어에서 가장 널리 쓰이는 방식 중 하나인 SAD (Sum of Absolute Difference)를 사용했다. 적정한 로직 크기를 갖는 동시에 실행 사이클을 줄이기 위해서는 네 종류의 인트라 예측 블록을 위한 SAD PE (Processing Element)의 효율적인 분할과 공유가 필수적이다. Andrzej[7]는 두 개의 로직 패스를 사용했는데 하나는 4x4 블록의 인트라 예측을 위한 것이고 다른 하나는 나머지 블록의 예측을 위한 것으로 가장 빈번히 사용되는 4x4 모드에만 초점을 맞췄다. 그러나 네 종류의 인트라 예측 블록은 동일한 연산 량을 가지기 때문에 두 개의 인트라 예측 블록이 SAD 와 트랜스폼 로직을 동일하게 공유한다면 하드웨어 사용률을 두 배로 증가시킬 수 있다. 이러한 대칭적 분할을 실현하기 위해 다음과 같이 설계하였다.

SAD PE와 트랜스폼 체인 (DST (Discrete Sine Transform), Q (Quantization), IQ (Inverse Quantization), IDST (Inverse Discrete Sine Transform))은 4x4 예측 로직을 위해 순환 방식으로 결합된다. 대부분의 4x4 블록의 인트라 예측을 위해서는 이전에 예측된 4x4 블록의 예측 결과가 필요하다. 4x4 블록의 예측 로직에서 SAD PE와 트랜스폼 로직의 실행 사이클의 일치는 파이프라인 효율을 최대화한다.

만약 4x4와 8x8 블록의 인트라 예측을 위한 SAD PE를 공유한다면 트랜스폼 로직 일정에 따라 4x4와 8x8 블록을 비교한 후에 예측할 다음 블록의 인트라 예측 타이밍이 지연된다. 이는 4x4와 16x16 블록의 인트라 예측을 위한 SAD PE를 공유하는 경우도 마찬가지다.

반면 4x4와 32x32 블록의 인트라 예측을 인터리빙한다면, 32x32 블록의 트랜스폼은 모든 4x4와 32x32 블록의 인트라 SAD 계산이 끝난 후에 시작하기 때문에 32x32 블록의 인트라 예측이 4x4 블록의 인트라 SAD 계산에 영향을 주지 않는다. 제안된 인트라 예측 로직의 구조는 그림 1에서 볼 수 있다. 4x4 와 32x32 블록의 인트라 예측은 9개의 SAD PE 를 공유하고 8x8과 16x16 블록의 인트라 예측은 또 다른 9개의 SAD PE를 공유한다.

기본적으로 SAD 계산은 원본 픽셀과 참조 픽셀의 뺄셈이다. 제안된 디자인에서 하나의 SAD PE는 4개의 뺄셈을 병렬로 수행한다. 그러므로 4x4 블록의 한 예측 모드에 대한 SAD 계산을 끝내기 위해서는 4 사이클이 필요하다. 우리는 4개의 SAD PE를 묶어서 4x4 블록의 한 예측 모드를 1 사이클 만에 끝내도록 설계했다. 예를 들어 SAD PE 0 ~ 3은 모드 2에 대한 SAD 계산을 하고, SAD PE 4 ~ 7은 모드 3에 대한 SAD 계산을 동시에 수행한다. 남은 하나의 SAD PE는 스페셜 모드에 대한 SAD 계산을 한다. 그러므로 mode0, 1, 10, 26의 SAD 계산을 하는데 각각 4 사이클이 소요된다.

최적 모드는 SAD가 가장 작은 모드로 결정된다. SAD 계산 과정에서 원본 픽셀과 참조 픽셀 간 레지듀얼 픽셀이 만들어지고, 레지듀얼 픽셀은 트랜스폼과

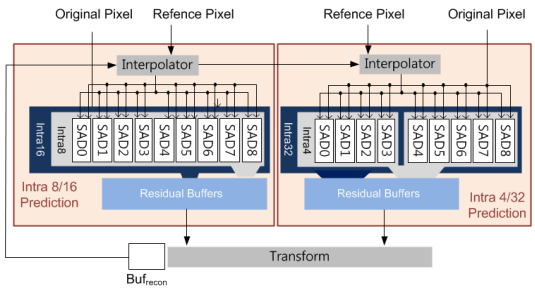

그림 1. 제안된 인트라 예측 로직 구조
Fig. 1. The architecture of proposed intra prediction logic

나머지 인코딩 과정을 위해 사용된다. 가장 작은 예측 비용을 갖는 레지듀얼만 필요하기 때문에 모든 모드의 레지듀얼이 저장되지는 않는다. 일반 모드와 스페셜 모드 간 SAD 계산은 독립적으로 수행되고, 스페셜 모드의 계산은 일반 모드의 계산보다 느리다. 그러므로 두 그룹의 모드 레지듀얼을 저장할 레지듀얼 버퍼는 두 개로 분리되어 있고 이는 그림 1에서 볼 수 있다.

4x4 블록의 최적 모드가 결정되고 나면 최적 모드의 레지듀얼을 이용하여 DST, Q, IQ, IDST를 수행한다. 그림 1에서 볼 수 있듯이 DST 결과는 BUF_{recon}에 저장된다. BUF_{recon}은 재구성된 LCU 블록 데이터를 저장하는 SRAM (Static Random Access Memory)버퍼이다. 만약 4x4와 8x8 블록의 비교 시 8x8 블록이 이긴다면 이전에 저장된 4x4 블록의 인트라 예측으로 재구성된 블록 데이터는 8x8 블록의 인트라 예측으로 재구성된 블록 데이터로 덧씌워진다.

4x4 블록 대비 8x8, 16x16, 32x32 블록의 인트라 예측 일정은 하나의 모드에 대한 SAD 계산을 위해 8개의 SAD PE를 사용하는 것을 제외하고는 거의 동일하다. 8개의 SAD PE는 레지듀얼 버퍼의 크기를 줄이기 위해 함께 사용된다. 4x4 인트라 예측과 달리 8x8, 16x16, 32x32 블록의 예측 시 최적 모드의 결정이 되기 전까지 최적 모드에 대한 레지듀얼이 저장되지 않는 대신 최적 모드 번호가 저장된다. 최적 모드가 결정되면 트랜스폼과 나머지 인코딩 과정을 위해 최적 모드의 레지듀얼이 계산된다.

인트라 예측의 전체 동작에 대한 정확한 타이밍 도는 그림 2에서 볼 수 있다. 타이밍도의 가장 중요한 것은 동일한 그룹의 인트라 블록 간의 인터리빙이다. 예를 들어 같은 로직을 공유하는 4x4와 32x32 블록은 인터리빙 되고, 인터리빙 단위는 4x4 인트라 예측 실행 사이클에 기초하여 결정된다.

전체 인트라 예측 실행 사이클을 증가시키지 않고 BUF_{recon}의 저장 공간을 줄이기 위해서 예측 일정을 약간 조정했다. BUF_{recon}은 32x32 픽셀을 저장할 수 있다. BUF_{recon}에 저장할 데이터는 트랜스폼 유닛의 마지막 단계인 IDCT의 결과이며, 32x32 블록이나 4개의 16x16 블록 또는 다양한 크기의 블록 조합이 될 수 있다. 처음 재구성된 8x8 블록을 저장하는 과정은 다음과 같다.

처음 4개의 4x4 블록과 1개의 8x8 블록의 최적 모드가 결정되면 두 블록의 예측 비용을 비교하여 승자를 결정하고 승자의 재구성된 블록을 BUF_{recon}에 저장한다. 4x4와 8x8 두 블록을 비교하기 전에 이미 만들어진 3개의 재구성된 4x4 블록을 BUF_{recon}에 저장해 둔다. 8x8 블록에 대한 승자가 결정되면 네 번째 4x4 블록의 트랜스폼이나 첫 번째 8x8 블록의 IDCT가 수행된다. 4x4 블록이 이길 경우엔 네 번째 재구성된 4x4 블록을 BUF_{recon}에 추가로 저장하고 8x8 블록이 이길 경우는 재구성된 8x8 블록을 이전에 재구성된 3

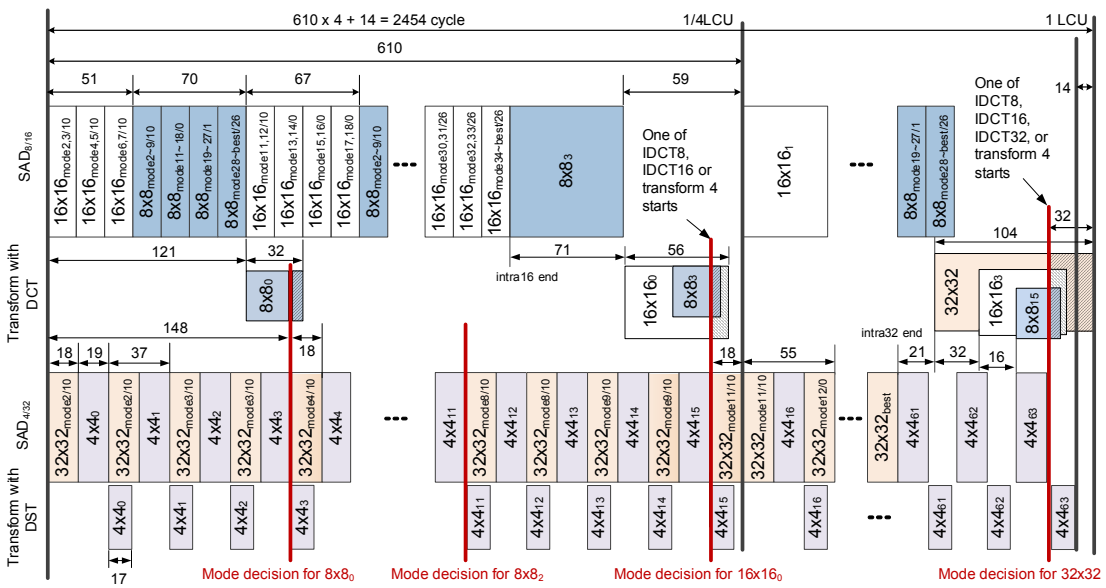

그림 2. 인트라 예측 및 트랜스폼의 전체 타이밍도
Fig. 2. Timing chart for entire intra prediction and transform

개의 4x4 블록이 저장되어 있는 BUF_{recon} 위치에 덮어 쓴다.

가능한 빨리 블록 간 승자를 결정하기 위해서 8x8과 16x16 블록에 대한 인트라 예측 일정을 약간 조정했다. $4x4_4$ 블록의 인트라 예측을 위해서는 $4x4_0$ ~ $4x4_3$ 또는 $8x8_0$ 블록 중 승자에 대한 재구성 블록이 필요하다. 이를 위해 51 클록에 $8x8_0$ 블록의 SAD 계산을 시작하게 하여 $4x4_4$ 블록의 인트라 예측이 지연되지 않도록 하였다.

그림 2에서 빗금 친 부분은 재구성된 블록을 생성하기 위해 IDCT를 수행하는 구간이다. 빗금 친 부분이 오버랩 된 것은 하나의 트랜스폼만 수행되는 것을 나타낸다. 예를 들어 $8x8_0$ ~ $8x8_3$ 블록이 $16x16_0$ 블록을 이겼을 때 $8x8_3$과 $16x16_0$ 블록의 트랜스폼이 오버랩 된 구간에서 $8x8_3$ 블록의 트랜스폼만 수행된다.

SAD PE에 모든 모드가 차례로 배열되어 있지는 않다. 예를 들어 번호가 낮은 모드가 SAD PE 0 ~ 3에 할당된 다른 모드와는 다르게 모드 12 는 SAD PE 0 ~ 3에, 모드 11은 SAD PE 4 ~ 7에 각각 할당되어 있다. 이렇게 부분적으로 불규칙하게 할당한 이유는 관련 모드 간 참조 픽셀을 생성할 때 인터폴레이터 로직을 공유하여 게이트 숫자를 줄이는 동시에 설계의 복잡 도를 줄이기 위해서이다.

스페셜 모드 중에서 planar와 DC 모드는 SAD를 계산하기 전에 참조 픽셀의 덧셈을 해야 한다. 로직의 크기와 예측의 지연을 줄이기 위해 이러한 모드를 위한 참조 픽셀의 덧셈은 다른 모드의 SAD 계산이 수행되는 동안 여러 사이클에 걸쳐 순차 덧셈 기를 사용하여 수행된다. 본 설계에서는 모드 10의 인트라 예측이 먼저 수행되고 모드 0과 모드 1의 인트라 예측이 수행된다.

Ⅲ. 구현 결과

제안된 인트라 예측 로직은 verilog 하드웨어 기술언어를 사용하여 구현되었다. 본 설계의 트랜스폼 부는 Pramod[9]에 제시된 최첨단 DCT와 DST 구조를 도입하여 구현했다.

65 나노 CMOS 기술을 사용하여 합성한 결과 전체 게이트 숫자는 1059K 개로 측정되었고 이에 대한 게이트의 분포는 표 1에서 보여주고 있다. 예측 로직의 크기는 트랜스폼 로직의 크기의 절반 정도이다.

제안된 인트라 예측과 트랜스폼 로직은 최대 251 MHz의 주파수로 동작하고 시뮬레이션을 통해 LCU 당 2454 사이클이 소요됨을 확인했다. 이전의 인트라

표 1. 인트라 예측 구현 비교
Table 1. Comparison of intra prediction implementation

		[5]	[7]	[8]	Proposed
Technology		65nm	130nm	90nm	65nm
Gate count	Pred	37	127	1,524	1059
	Transform	-	-		
Max. Freq. (MHz)		500	200	146	257
Throughput (1080p frame/sec)		13.4	29.4	35	56
Prediction		Partial	Pruned	Modified	Full
Feature		I4~I64	I4~I32	I4~I64	I4~I32
사이클s/LCU		72,520	3,305	2,240	2,454

예측 연구와의 비교는 표 1에서 보여 진다. Daniel[5]과 Andrzej[7]의 설계에는 인트라 예측 로직만 포함되어 있고 트랜스폼 로직은 포함되어 있지 않다. 뿐만 아니라 설계를 단순화하기 위해 모드 중 일부만 적용하거나 프루닝 알고리즘을 사용했고, 그렇게 했음에도 throughput 이 30 프레임/초가 되지 않기 때문에 실시간 처리에 어려움이 있는 반면 제안된 설계의 throughput은 56 프레임/초로 거의 모든 영상의 실시간 처리가 가능하다. Dam[8]은 재구성 과정을 포함한 모든 인트라 예측 모드를 구현했다. 그러나 Dam[8]의 인트라 예측은 두 개의 연속적인 이미지가 필요하기 때문에 실시간 인코더 작업의 실현 가능성이 제한된다. 또한 예측 처리 속도를 향상시키기 위해 일부 블록에서 참조 픽셀 대신 원본 픽셀을 사용하였기 때문에 해당 연구는 HEVC 표준을 완벽하게 준수한 것이 아니다. 이러한 것들을 배제하더라도 Dam[8]의 설계에 비해 본 논문에서 제안된 설계가 사이즈 측면과 throughput 측면에서 모두 우수하다. 따라서 본 논문에 제안된 설계의 성능, 게이트 수, 완성도 측면에서 기존의 설계를 능가한다고 말할 수 있다.

Ⅳ. 결 론

본 논문에서는 HEVC 용 고속 인트라 예측 로직 구현 방법을 제시했다. 4개의 예측 단위 크기에 대한 예측 로직의 분할과 SAD PE의 할당 및 예측과 트랜스폼 간 인터리빙을 통해 풀 HD 급 영상에서 56 프레임/초의 속도로 전체 인트라 모드를 사용한 인코딩을 가능하게 했다. 적정한 로직 사이즈와 더불어 거의 모든 영상에서 실시간 처리를 수행할 수 있다.

References

[1] J. H. Moon and J. K. Han, "Fast intra prediction mode decision using most probable mode in HEVC," in *Proc. KICS Winter Conf.*, pp. 841-843, Yongpyong, Korea, Feb. 2014.

[2] S. H. Yang, H. J. Shim, D. H. Lee, and B. W. Jeon, "Transform skip mode fast decision method for HEVC encoding," *J. KICS*, vol. 39, no. 4, pp. 172-179, Apr. 2014.

[3] F. Li, G. Shi, and F. Wu, "An efficient VLSI architecture for 4x4 intra prediction in the high efficiency video coding (HEVC) standard," *IEEE Int. Conf. Image Process.*, pp. 373-376, Brussels, Sept. 2011.

[4] Z. Liu, D. Wang, H. Zhu, and X. Huang, "41.7BN-Pixels/s reconfigurable intra prediction architecture for HEVC 2560x1600 encoder," *IEEE Int. Conf. Acoustics, Speech and Sign. Process.*, pp. 2634-2638, Vancouver, BC, May 2013.

[5] D. Palomino, F. Sampail, L. Agostini, S. Bampi, and A. Susin, "A memory aware and multiplierless VLSI architecture for the complete intra prediction of the HEVC emerging standard," *IEEE Int. Conf. Image Process.*, pp. 201-204, Orlando, FL, Sept. 2012.

[6] S. K. Jung and S. S. Lee, "Design of merged architecture for 4x4 IDCT/IDST for HEVC," in *Proc. KICS Int. Conf. Commun.*, pp. 918-919, Jeju Island, Korea, Jun. 2015.

[7] A. Abramowski and G. Pastuszak, "A double-path intra prediction architecture for the hardware H.265/HEVC encoder," *Int. Symp. Design and Diagnostics of Electron. Cir. & Syst.*, pp. 27-32, Warsaw, Apr. 2014.

[8] D. M. Tung, T. L. T. Don, and T. T. Anh, "An efficient parallel execution for intra prediction in HEVC Video Encoder," *Int. Conf. Comput., Management and Telecommun.*, pp. 233 - 238, Da Nang, Apr. 2014.

[9] P. K. Meher, S. Y. Park, B. K. Mohanty, K. S. Lim, and C. Yeo, "Efficient integer DCT architectures for HEVC," *IEEE Trans. Cir. and Syst. for Video Technol.*, vol. 24, no. 1, pp. 168-178, Jan. 2014.

조 현 수 (Hyeonsu Jo)

2015년 2월 : 동국대학교 전자공학과 졸업

2015년 3월~현재 : 동국대학교 전자공학과 석사과정

<관심분야> 비디오 코덱 칩 설계

홍 유 표 (Youpyo Hong)

1991년 2월 : 연세대학교 전지공학과 학사

1993년 5월 : University of Southern California 전기공학과 석사

1998년 8월 : University of Southern California 컴퓨터공학과 박사

1998년 7월~1999 2월 : Synopsys, Hillsboro, Senior Engineer

1999년 3월~현재 : 동국대학교 전자전기공학부 전자공학전공 교수

<관심분야> 멀티미디어 칩 설계, SOC 설계

장 한 별 (Hanbeyoul Jang)

2015년 8월 : 중부대학교 정보통신학과 졸업

2015년 9월~현재 : 동국대학교 전자공학과 석사과정

<관심분야> 비디오 코덱 칩 설계