ISSN: 1226-7244 (Print)
ISSN: 2288-243X (Online)
논문번호 15-03-14

j.inst.Korean.electr.electron.eng.Vol.19,No.3,378~384,September 2015
http://dx.doi.org/10.7471/ikeee.2015.19.3.378

곱셈기를 재사용하는 16×16 HEVC 코어 역변환기 설계

16×16 HEVC Inverse Core Transform Architecture Using Multiplier Reuse

이 종 배*, 이 성 수*★

Jong-Bae Lee*, Seongsoo Lee*★

Abstract

In conventional HEVC inverse core transform architectures, extra n×n inverse transform block is added to 2n×2n inverse transform block, and it operates as one 2n×2n inverse transform block or two n×n inverse transform blocks. Thus, same number of pixels are processed in the same time, but it suffers from increased hardware size due to extra n×n inverse transform block. To avoid this problem, a novel 8×8 HEVC inverse core transform architecture was proposed to eliminate extra 4×4 inverse transform block based on multiplier reuse. This paper extends this approach and proposes a novel HEVC 16×16 inverse core transform architecture. Its frame processing time is same in 4×4, 8×8, and 16×16 inverse core transforms, and reduces gate counts by 13%.

요 약

기존의 HEVC 코어 역변환기에서는 동일한 시간에 동일한 수의 화소를 처리하기 위해서 2n×2n 역변환기에 여분의 n×n 역변환기를 추가하여 한 개의 2n×2n 역변환기 또는 두 개의 n×n 역변환기로 동작하게 하였으나 여분의 n×n 역변환기 때문에 하드웨어 크기가 증가하는 단점이 있다. 이러한 문제점을 해결하기 위해 곱셈기를 재사용하여 여분의 4×4 역변환기를 없앤 새로운 8×8 HEVC 코어 역변환기 구조가 제안되었으며, 본 논문에서는 이를 확장한 16×16 HEVC 코어 역변환기 구조를 제안한다. 제안하는 16×16 HEVC 역변환기는 4×4 역변환, 8×8 코어 역변환, 16×16 코어 역변환에서 프레임 처리 시간이 모두 동일하며, 여분의 역변환기를 사용하는 아키텍처에 비해 게이트 수를 13% 줄일 수 있다.

Key words : HEVC, inverse core transform, multiplier reuse, small hardware, low power

Ⅰ. 서론

최근 들어 UHD(Ultra High Definition)와 같은 초고화질, 초고해상도 영상의 상용 서비스가 시작됨에 따라 4k급, 8k급 영상의 처리 필요성이 대두되기 시작하였다. 이에 따라 기존의 동영상 압축 국제 표준인 H.264/AVC (Advanced Video Coding)[1]로는 이러한 영상을 수용할 수 없는 문제에 봉착하였으며, 특히 4k급, 8k급 영상의 처리는 기존의 압축 표준보다 훨씬 더 높은 압축률을 필요로 한다. 이와 같은 문제를 해결하고자 ISO/IEC와 ITU-T는 공동으로 H.264/AVC에 비해 2배 향상된 압축률을 목표로 하는 차세대 비디오 코덱 기술인 HEVC (High Efficiency Video Coding)[2]을 최종 발표하였고, 현재 많은 연

* School of Electronic Engineering, Soongsil University
★ Corresponding author: sslee@ssu.ac.kr, 02-820-0692
※ Acknowledgment
"This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2010-0025041)."
Manuscript received Aug. 6, 2015; revised Aug. 26, 2015 ; accepted Aug. 27. 2015

구팀들이 HEVC의 하드웨어 구현에 대해 활발한 연구를 진행하고 있다.

HEVC의 핵심 프로세스 중의 하나인 코어 역변환(core inverse transform)[3][4]은 이전의 H.264/AVC와 상당히 다른 성질을 가지고 있다. 먼저, H.264에서는 홀수 부분과 짝수 부분 모두에 버터플라이 구조(butterfly architecture)가 적용되는데 비해 HEVC에서는 짝수 부분만 버터플라이 구조를 적용할 수 있다. 또한 HEVC에서는 재귀 구조 (recursive architecture)가 적용되기 때문에 상위 크기 블록의 짝수 부분과 하위 크기 블록 전체의 계수가 동일하다[5]. 이러한 성질 때문에 HEVC에서는 하드웨어 구현 시에 몇 가지 문제가 발생한다[5]-[7].

일반적으로 H.264/AVC에서는 n×n 역변환기 2개에 버터플라이 덧셈기를 결합하여 2n×2n 역변환기를 구성할 수 있다. 이 경우, 2n×2n 역변환기를 2개의 n×n 역변환기로 동작시킬 수 있으며, 같은 크기의 화소 영역을 n×n 역변환할 때와 2n×2n 역변환할 때 걸리는 시간이 같다. 이에 반하여 HEVC에서는 2n×2n 역변환기의 짝수 부분은 n×n 역변환기와 동일하지만 홀수 부분은 다르기 때문에, 2n×2n 역변환기를 오직 1개의 n×n 역변환기로만 동작시킬 수 있다. 따라서 같은 크기의 화소 영역을 n×n 역변환할 때 걸리는 시간이 2n×2n 역변환할 때 걸리는 시간의 2배가 된다[5]. 이 경우, 속도가 느린 n×n 역변환에 맞춰 시스템 처리 능력이 낮아지는 문제가 발생한다.

기존의 HEVC 코어 역변환기[8]에서는 2n×2n 역변환기에 여분의 n×n 역변환기를 추가하는 방법을 제시하였다. 다만 이 구조에서는 여분의 n×n 역변환기 때문에 하드웨어 크기가 증가한다. 또한 2n×2n 역변환을 수행할 때 n×n 역변환 하드웨어가 놀게 되고, n×n 역변환을 수행할 때에는 2n×2n 역변환의 홀수 부분 하드웨어가 놀게 된다[5].

[8]의 문제점을 해결하기 위해서 [5]에서는 여분의 n×n 역변환기를 추가하는 대신에 2n×2n 역변환의 홀수 부분 하드웨어를 수정하여 이 하드웨어가 n×n 역변환도 함께 수행할 수 있도록 하였다. 이 경우, 2n×2n 역변환의 홀수 부분 하드웨어가 다소 커지는 대신에 여분의 n×n 역변환기가 필요없기 때문에 전체적으로 하드웨어 크기는 작아진다. [5]에서는 4×4, 8×8 역변환을 수행할 수 있는 하드웨어까지만 제안되었는데, 게이트 수가 [8]에 비해 12%까지 감소하였다.

본 논문에서는 [5]를 16×16 역변환까지 확장한 HEVC 코어 역변환기를 제안한다. 제안하는 아키텍처는 곱셈기를 재사용하는 아키텍처를 채택하여 상위 크기 블록의 홀수 부분 역변환 하드웨어가 하위 크기 블록의 역변환을 겸용으로 수행할 수 있도록 설계되었다. 또한 하나의 하드웨어로 4×4, 8×8, 16×16 역변

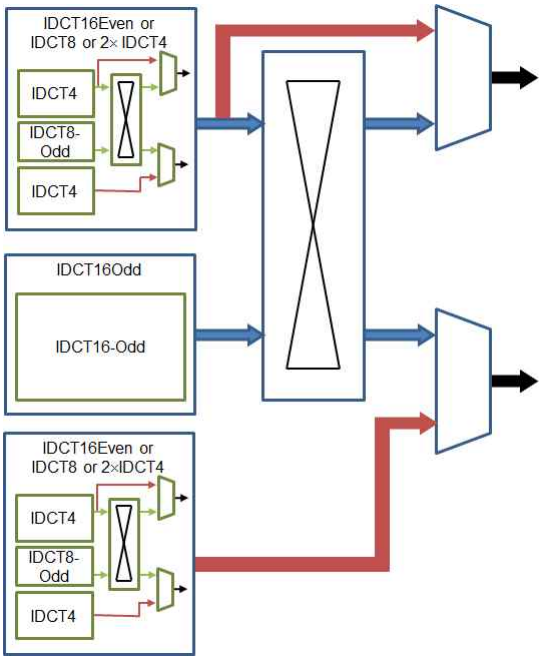


Fig. 1. Conventional 16×16 HEVC core inverse transform architecture based on [8].

그림 1. [8]에 기반한 기존 16×16 HEVC 코어 역변환기 아키텍처.

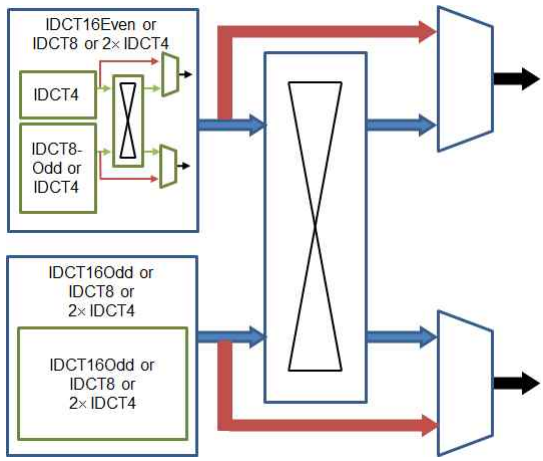
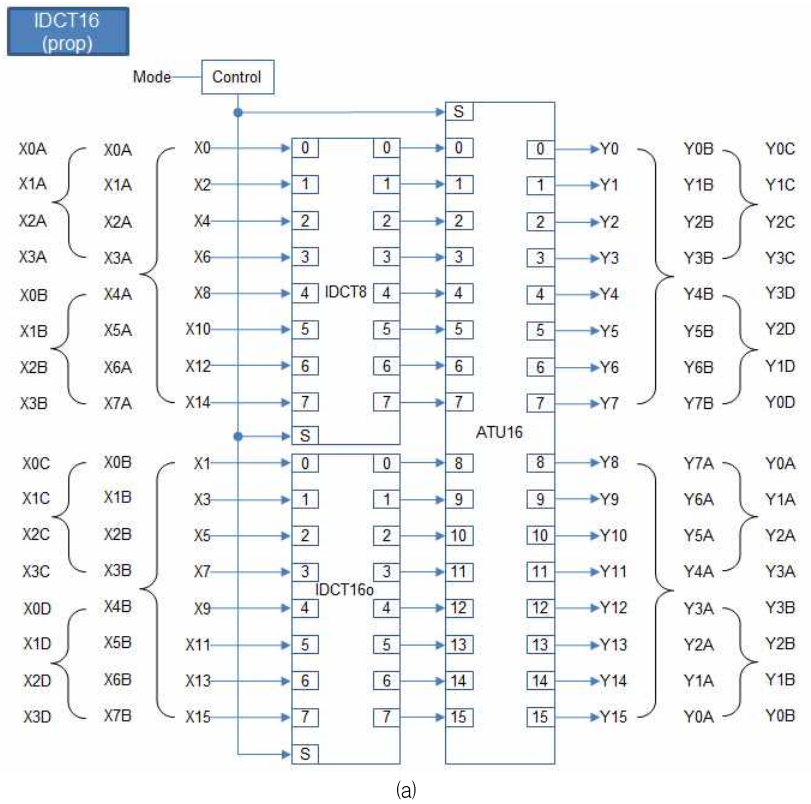


Fig. 2. Proposed 16×16 HEVC core inverse transform architecture.

그림 2. 제안하는 16×16 HEVC 코어 역변환기 아키텍처.



(a)

Fig. 3. Design of the proposed inverse core transform. (a) Top block.
그림 3. 제안하는 코어 역변환기의 설계. (a) 최상위 블록.

환을 수행할 수 있으며 같은 크기의 화소 영역을 역변환할 때 걸리는 시간이 모두 동일하다.

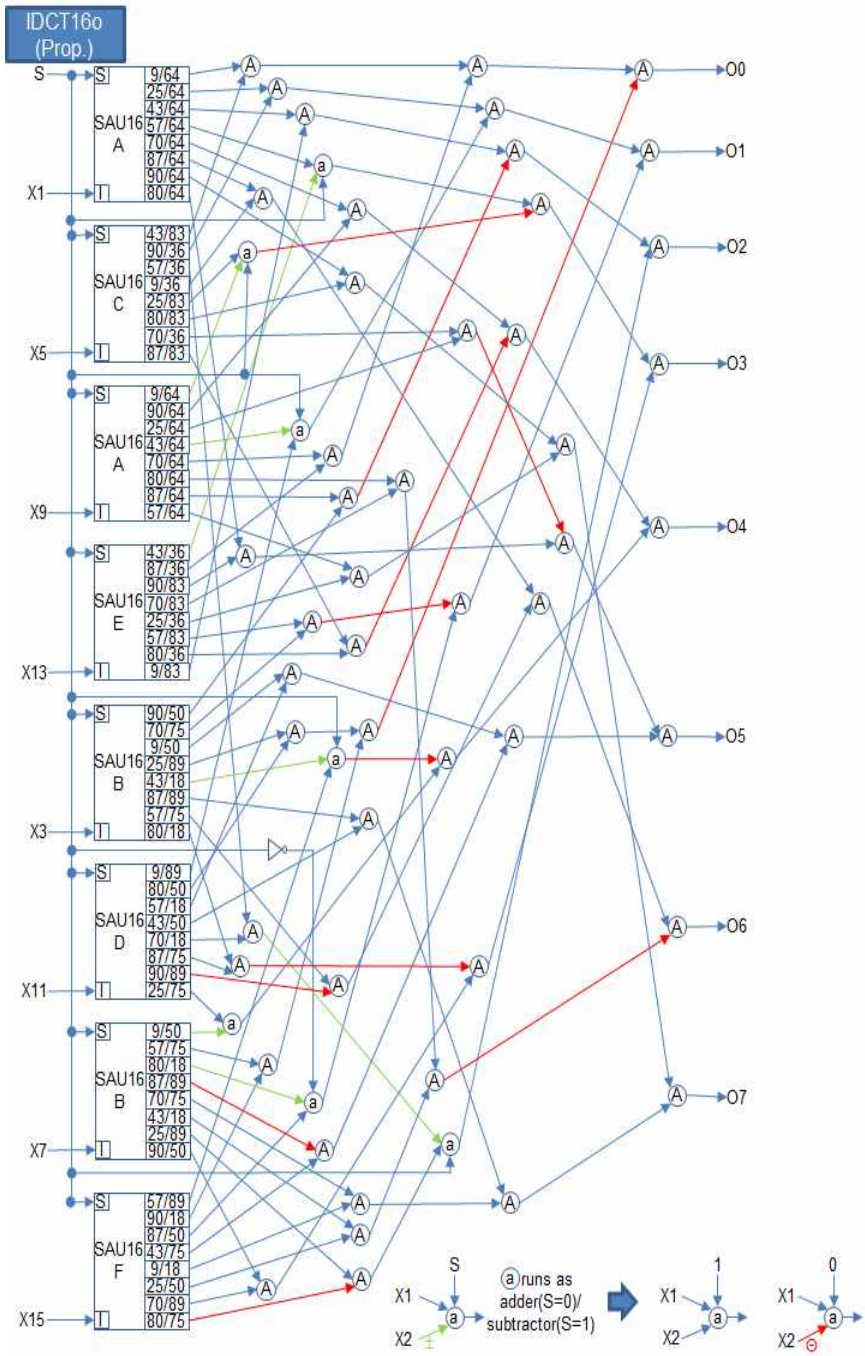
II. 제안하는 코어 역변환기 구조

[8]에 기반한 기존 아키텍처는 그림 1과 같이 4개의 4×4 역변환기, 2개의 8×8 홀수 역변환기, 1개의 16×16 홀수 역변환기, 이들을 연결하는 버터플라이 덧셈기 및 멀티플렉서가 필요하지만, [5]에 기반하여 본 논문에서 제안한 아키텍처는 그림 2와 같이 1개의 4×4 역변환기, 1개의 수정된 8×8 홀수 역변환기 (4×4 역변환기 1개로도 겸용 동작), 1개의 수정된 16×16 홀수 역변환기 (4×4 역변환기 2개, 8×8 역변환기 1개로도 겸용 동작), 이들을 연결하는 버터플라이 덧셈기 및 멀티플렉서로 구성된다. 따라서 그림 2의 아키텍처는 1개의 16×16 역변환기로 동작하거나 2개의 8×8 역변환기로 동작하거나 4개의 4×4 역변환기로 동작할 수 있다.

제안하는 아키텍처를 설계할 때 가장 어려운 점은 하나의 하드웨어로 두 개 이상의 변환을 겸용으로 수행할 수 있도록 하는 것이다. 본 논문에서는 HEVC 코어 역변환을 수학적으로 분석하여 곱셈기를 최대한 재사용할 수 있도록 설계하였다.

제안하는 역변환기의 구조는 그림 3(a)와 같으며, 그림 2에 설명한 것처럼 8×8 역변환(16×16 역변환 짝수 부분과 동일)을 수행하는 IDCT8, 하나의 하드웨어로 16×16 역변환의 홀수 부분 수행과 8×8 역변환 전체를 수행하고, 4×4 역변환 전체를 2번 수행 할 수 있는 어레이 곱셈기인 IDCT16o와 버터플라이 및 MUX 동작을 수행하는 블록 ATU16으로 크게 구성되어 있으며, 마지막으로 모드 입력 값에 따라 각 부에 제어 신호를 보내는 주는 Control 블록으로 구성되어 있다. 각 블록은 다음과 같이 동작 한다.

IDCT8 블록은 [5]의 그림 4에서 이미 설계된 블록과 동일하며, 모드 값이 1이면 8×8역변환 전체(16×16 역변환 짝수 부분)를 수행하며 모드 값이 0이면 4×4



(b)

Fig. 3. Design of the proposed inverse core transform. (cont'd) (b) IDCT16o.
그림 3. 제안하는 코어 역변환기의 설계. (계속) (b) IDCT16o.

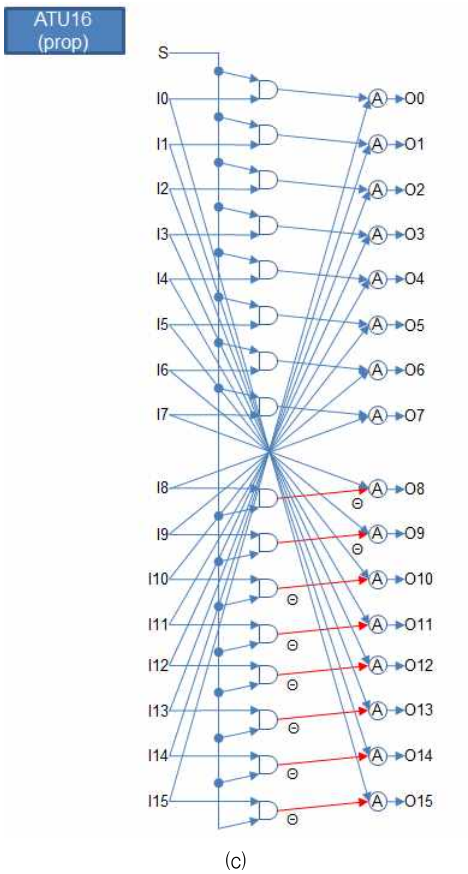


Fig. 3. Design of the proposed inverse core transform. (cont'd) (c) ATU16.
그림 3. 제안하는 코어 역변환기의 설계. (계속) (c) ATU16.

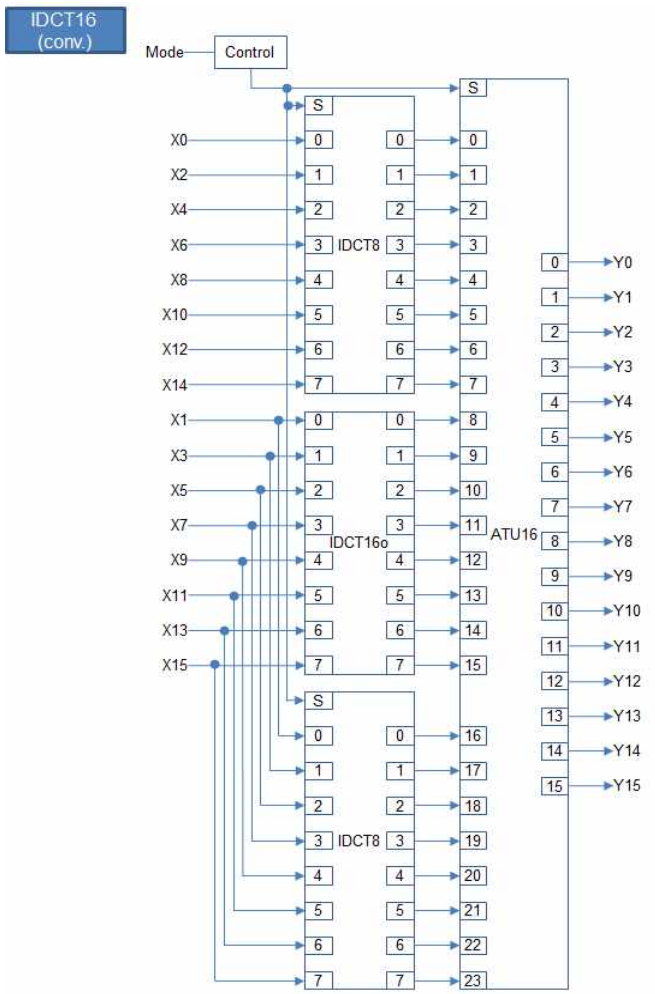


Fig. 4. Design of the conventional inverse core transform based on [8].
그림 4. [8]에 기반한 기존 코어 역변환기의 설계.

역변환을 두 번 수행하게 된다.

그림 3(b)는 IDCT16o 블록의 구조이고 어레이 곱셈 연산을 수행하게 된다. 또한 계수 생성기인 SAU16A, SAU16B, SAU16C, SAU16D, SAU16E. SAU16F를 포함한다. 계수 생성기 SAU16은 2bit 모드 값이 00일 때는 4×4 역변환 계수인 {36,64,83}을 생성하고 모드 값이 01일 때 8×8 역변환 계수인 {18,36,50,64,75,83,89}을 생성하며 모드 값이 10일 때 16×16 역변환의 홀수 부분 계수인 {9,25,43,57, 70,80,87,90}을 생성하게 된다. 생성된 계수 값들은 IDCT16o 블록의 어레이 곱셈을 통하여 각 역변환에 해당하는 출력 값을 계산한다.

그림 3(c)는 ATU16 블록을 나타내며 기존의 구조에서는 MUX를 사용하였지만 제안하는 구조는 AND 게이트로 대체하여 MUX의 기능은 유지하지만 하드웨어 면적을 줄였다. 모드 값이 1이면 16×16 코어 역변환으로 동작해 버터플라이 연산이 수행이 되고, 모드 값이 0이면 블록의 입력 값이 바이패스되어 출력

으로 나가는데, 버터플라이 구조를 가지고 있기 때문에 그림 3(a)에서 보는 것과 같이 출력의 순서가 바뀌게 된다.

Control 블록은 2비트 모드 값 입력에 따라 각 블록을 제어하며, 00일 때는 4×4 역변환 모드, 01일 때 8×8 역변환 모드, 10일 때 16×16 역변환 모드로 동작할 수 있게 각 블록에 제어 신호를 출력하여 준다.

III. 구현 및 검증

그림 4는 제안하는 16×16 HEVC 코어 역변환기와 하드웨어 크기를 비교하기 위해 [8]의 구조를 기반으로 하여 16×16 HEVC 코어 역변환기를 설계한 것이다. 이 코어 역변환기는 그림 1에 설명한 것처럼 8×8 코어 역변환 (16×16 코어 역변환의 짝수 부분과 동일)을 수행하는 IDCT8를 두 개 사용하며, 16×16 코어 역변환의 홀수 부분을 처리하는 어레이 곱셈기인 IDCT16o, 버터플라이 및 MUX 동작을 수행하는 ATU16의 세 부분으로 구성되어 있다. 입력 모드 값에 따라 Control 블록은 각 블록을 제어하게 되며, 00일 때 4x4 역변환을 네 번, 01일 때 8x8 역변환 두 번과 10일 때 16x16 역변환을 처리하게 된다. 이 구조는 상위의 역변환을 처리할 때, 추가로 구성된 하위 블록이 동작을 하지 않게 된다. 즉 유휴 하드웨어가 일부 존재하여 제안하는 구조에 비해 하드웨어가 커지게 된다.

본 논문에서는 제안한 아키텍처를 검증하고 하드웨어 크기를 비교하기 위해 그림 3과 그림 4를 동일한 0.18μm 공정 라이브러리를 사용하여 설계하였고, IDEC에서 제공한 EDA Tool인 Design Vision 툴을 이용하여 합성을 진행하였다.

Table 1. Synthesis results.
표 1. 합성 결과.

	Proposed (Fig. 3)	Conventional [8] (Fig. 4)	Gain (%)
Size (gates)	40299	46336	13
Speed (MHz)	300	300	-

표 1은 제안하는 구조 (그림 3)와 기존 구조[8] (그림 4)의 합성 결과를 비교한 것이다. 두 설계 모두 300MHz에서 동작하여 속도는 동일하나 2-input NAND로 환산한 게이트 수는 제안한 구조가 40299, 기존의 구조가 46336게이트로, 제안하는 구조가 13% 가량 크기가 작아진 것을 알 수 있다.

IV. 결론

본 논문에서는 기존 HEVC 코어 역변환기의 하드웨어 크기를 줄이기 위해서 곱셈기를 재사용하여 여분의 4×4 역변환기를 없앤 새로운 16×16 HEVC 코어 역변환기 구조를 제안한다. 제안하는 16×16 HEVC 역변환기는 4×4 역변환, 8×8 코어 역변환, 16×16 코어 역변환에서 프레임 처리 시간이 모두 동일하며, 여분의 역변환기를 사용하는 아키텍처에 비해 게이트 수를 13% 줄일 수 있다.

References

[1] T. Wiegand, G. Sullivan, G. Bjontgaard, and A. Luthra, “Overview of the H.264/AVC Video Coding Standard”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560-576, Jul. 2003.

[2] G. Sullivan, J. Ohm, W. Han, and T. Wiegand, “Overview of the High Efficiency Video Coding (HEVC) Standard”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[3] A. Fuldseth, G. Bjøntegaard, and M. Budagavi, “CE10: Core Transform Design for HEVC,” JCTVC-G495, Nov. 2011.

[4] M. Budagavi and V. Sze, “Unified Forward+Inverse Transform Architecture for HEVC”, Proceedings of IEEE Internation Conference on ICIP, pp. 209-212, Oct. 2012.

[5] J. Lee and S. Lee, “8×8 HEVC Inverse Core Transform Architecture Using Multiplier Reuse”, Journal of IKEEE, vol. 17, no. 4, pp. 570-578, Dec. 2013.

[6] S. Han, W. Nam, and S. Lee, “Design of Low-Area HEVC Core Transform Architecture”, Journal of IKEEE, vol. 17, no. 2, pp. 119-128, Jun. 2013.

[7] J. Park, W. Nam, S. Han, and S. Lee, “2-D Large Inverse Transform (16x16, 32x32) for HEVC (High Efficiency Video Coding)”, Journal of Semiconductor Technology and Science, vol. 12, no. 2, pp. 203-211, Jun. 2012.

[8] P. Meher, S. Park, B. Mohanty, K, Lim, and C. Yeo, “Efficient Integer DCT Architecture for HEVC”, IEEE Transactions on Circuits and Systems for Video Technology, vol. 24, no. 1, pp.

168-178, Jan. 2013.

BIOGRAPHY

Jong-Bae Lee (Member)

2011 : BS degree in Electronic Engineering, Hanseo University.
2014 : MS degree in Electronic Engineering, Soongsil University.
2014~Now : Ph.D candidate in Electronic Engineering, Soongsil University

<Main Interest> HEVC, Low-Power SoC Design, Multimedia SoC Design, Battery Management

Seongsoo Lee(Life Member)

1991 : BS degree in Electronic Engineering, Seoul National University.
1993 : MS degree in Electronic Engineering, Seoul National University.
1998 : PhD degree in Electrical Engineering, Seoul National University.
1998~2000 : Research Associate, University of Tokyo
2000~2002 : Research Professor, Ewha Womans University
2002~Now : Professor in School of Electronic Engineering, Soongsil University

<Main Interest> HEVC, Low-Power SoC Design, Multimedia SoC Design, Battery Management