
H.264/AVC 복호기를 위한 고성능 연산처리 인트라 예측기 설계

김선철* · 류광기**

A Design of High Performance Operation Intra Predictor for H.264/AVC Decoder

Xianzhe Jin* · Kwangki Ryoo**

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

요 약

본 논문에서는 H.264/AVC 복호기를 위한 고성능 연산처리 인트라 예측기를 제안한다. 기존의 인트라 예측기는 4x4블록에 적용되는 17개의 예측모드를 효율적으로 연산하기 위해 공통 연산기를 사용하였다. 하지만 기존의 공통 연산기는 한 픽셀에 적용되는 연산 수식을 분석하여 설계되었기 때문에, 16개 픽셀의 4x4 블록을 연산하기 위해 4개의 공통 연산기를 사용하며 4 사이클을 소요한다. 본 논문에서는 병렬 연산을 위한 T3(Three Type Transform) 연산기를 제안한다. 제안하는 T3 연산기는 17개의 인트라 예측 모드를 3가지 형태로 나누어, 4x4 블록의 16개 픽셀에 적용되는 연산 수식을 한 사이클에 처리한다. 제안하는 인트라 예측기와 기존의 인트라 예측기의 예측 수행 사이클을 각 모드 별로 비교한 결과, 제안하는 인트라 예측기가 평균 58.95%의 향상된 결과를 얻었다.

ABSTRACT

This paper proposes a parallel operation intra predictor for H.264/AVC decoder. In previous intra predictor design, common operation units were designed for 17 prediction modes in order to compute more effectively. However, it was designed by analyzing the equation applied to one pixel. So, there are four operation units for computing 16 pixels in a 4x4 block and they need four cycles. In this paper, the proposed intra predictor contains T3(Three Type Transform) operation unit for parallel operation. It divides 17 modes into 3 types to calculate 16 pixels of a 4x4 block in only one cycle and needs 16 cycles minimum in 16x16 block. As the result of the experiment, in terms of processing cycle, the performance of proposed intra predictor is 58.95% higher than the previous one.

키워드

H.264/AVC, 인트라 예측, 병렬 연산, 공통연산기

Key word

H.264/AVC, Intra Prediction, Parallel Operation, Common Operation Unit

* 정회원: 한밭대학교 정보통신전문대학원

** 중신회원: 한밭대학교 정보통신공학과 (교신저자, kkryoo@hanbat.ac.kr)

접수일자 : 2012. 05. 09

심사완료일자 : 2012. 07. 20

Open Access <http://dx.doi.org/10.6109/jkiice.2012.16.11.2503>

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

I. 서 론

H.264/AVC는 ISO의 MPEG 연구기관과 ITU의 연구 그룹인 VCEG에 의해 개발되었으며, 동일한 화질을 기준으로 기존 압축 표준인 MPEG-2 보다 2배 높은 압축률과 MPEG-4 대비 40% 높은 압축률을 제공한다[1]. 하지만 새로운 알고리즘의 도입으로 연산의 복잡도가 증가하여, 많은 양의 연산 처리가 요구된다[2]. 따라서 H.264/AVC의 부호기 및 복호기는 실시간처리를 위하여 하드웨어로 설계되고 있으며[3-5], 설계되는 하드웨어 구조 및 알고리즘에 따라 그 성능이 좌우되기 때문에 세부 알고리즘을 효율적으로 구현하는 기술이 중요하다.

인트라 예측은 휘도 영상을 위한 4x4 블록과, 16x16 블록, 색차 영상을 위한 8x8 블록의 예측 알고리즘을 제공하기 때문에 연산의 복잡도가 높다. 기존의 인트라 예측기는 인트라 예측모드에 따른 연산 수식을 효율적으로 연산하기 위해 공통 연산기를 설계 하였으며, 하나의 공통 연산기는 1 사이클에 하나의 픽셀을 연산한다. 따라서 기존 예측기는 4개의 공통 연산기를 사용하여 4x4 블록의 16픽셀을 연산하기 때문에 4 사이클이 소요된다 [6-7].

본 논문에서는 17개의 인트라 예측모드의 연산 수식을 분석한 후, 4x4 블록의 16개 픽셀을 한 사이클에 병렬 연산하는 T³(Three Type Transform) 연산기를 이용한 인트라 예측기를 제안한다. 본 논문의 구성은 다음과 같다. 2장에서는 인트라 예측 알고리즘에 대해 기술하고, 3장에서는 기존 연산기에 대해 기술한다. 4장에서는 연산 수식을 분석하여, 5장과 6장에서 T³ 연산기와 인트라 예측기를 제안한다. 7장에서는 실험 결과를 기술하고, 마지막으로 8장에서는 결론을 통해 본 논문을 요약한다.

II. 인트라 예측 알고리즘

인트라 예측은 H.264/AVC의 압축 기법 중 공간적인 중복성을 제거하기 위한 알고리즘으로써, 프레임 내의 인접한 블록들 사이의 픽셀들이 유사한 값을 가지고 있는 것을 기초로 하여, 현재 블록의 픽셀 값들을 인접한 블록의 픽셀 값으로부터 예측한 후 예측된 픽셀 값과 실제 픽셀 값의 오차를 구한 후, 인트라 예측 모드의 정보

와 오차 정보 데이터만을 전송하여 압축이 이루어진다. 17개의 인트라 예측모드는 휘도 4x4블록을 위한 9개의 예측 모드, 휘도 16x16 블록을 위한 4개의 예측 모드, 색차 8x8 블록을 위한 4개의 예측 모드이다.

2.1. 휘도 4x4 블록의 예측 모드

휘도 4x4 블록의 예측 모드는 예측하고자 하는 영상이 복잡할 경우, 휘도 16x16 블록을 보다 세밀하게 예측하기 위해 사용한다. 휘도 4x4블록을 위한 9가지 예측 모드는 그림 1과 같이 8가지의 방향성 예측 모드와 인접한 8개 픽셀의 평균값으로 예측하는 평균치 모드(DC)가 있다.

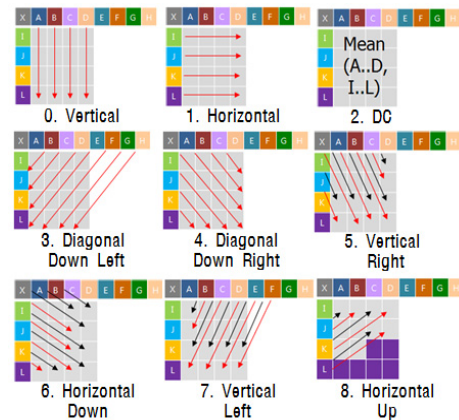


그림 1. 휘도 4x4 블록의 예측모드
Fig. 1 Luma 4x4 prediction mode

휘도 4x4 블록의 예측을 수행할 경우 그림 2와 같이, 4x4 블록의 왼쪽과 왼쪽상단, 위쪽, 오른쪽 상단에 있는 인접한 13개의 픽셀들을 참조하며, 휘도 16x16블록 내에서 휘도 4x4블록의 인트라 예측은 0번부터 15번의 순서로 진행된다[8].

2.2. 휘도 16x16 블록의 예측 모드

휘도 16x16 블록의 예측모드는 매크로블록의 휘도 성분을 한 번의 동작으로 예측하여, 단조로운 영상 부분에서 가장 좋은 효과를 가져 온다. 휘도 16x16 예측 모드는 그림 3과 같이, 수직 모드(Vertical), 수평 모드(Horizontal), 평균치 모드, 평면 모드(Plane)로 4개의 예측 모드가 존재한다.

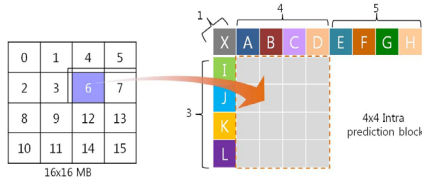


그림 2. 휘도 4x4 블록의 예측 순서와 참조 픽셀
Fig. 2 Luma 4x4 prediction sequence and reference pixel

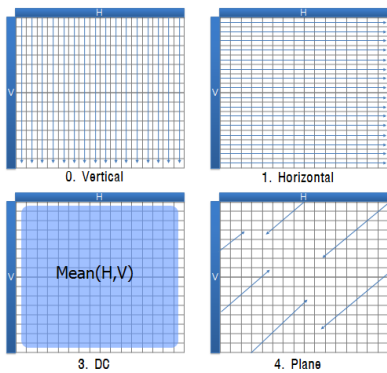


그림 3. 휘도 16x16 블록의 예측모드
Fig. 3 Luma 16x16 prediction mode

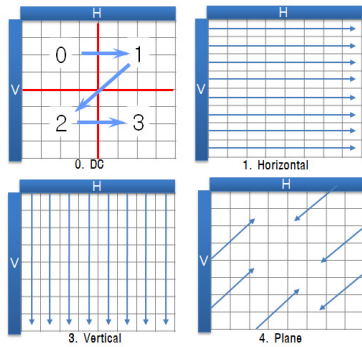


그림 4. 색차 8x8 블록의 예측모드
Fig. 4 Chroma 8x8 prediction mode

2.3. 색차 8x8 블록의 예측 모드

색차 8x8 예측모드는 그림 4와 같이 평균치 모드, 수평 모드, 수직 모드, 평면 모드로 4개의 모드가 있으며, 휘도 16x16 예측 모드와 참조하는 픽셀의 수만 다를 뿐 동일한 알고리즘이 적용된다.

III. 기존의 공통 연산기

기존의 공통 연산기는 한 픽셀에 적용되는 17개의 인트라 예측 모드의 연산 수식을 분석하여 설계되었다 [3-4]. 인트라 예측 모드의 연산 수식은 휘도 4x4 블록의 방향성 예측 모드 연산 수식, 4x4 블록, 8x8 블록, 16x16 블록의 평균치 모드 연산 수식, 8x8 블록, 16x16 블록의 평면 모드 연산 수식으로 나눌 수 있다. 식(1)과 (2)는 휘도 4x4 블록의 방향성 예측모드의 두 가지 연산 수식이다.

$$Dr0 = (P0 + 2 \times P1 + P2 + 2) \gg 2 \quad (1)$$

$$Dr1 = (P0 + P1 + 1) \gg 1 \quad (2)$$

식(3)부터 식(6)까지는 4x4 블록, 8x8 블록, 16x16 블록의 평균치 모드를 위한 연산 수식이다.

$$DC0 = (P0 + P1 + P2 + P3 + 2) \gg 2 \quad (3)$$

$$DC1 = (P0 + P1 + P2 + \dots + P7 + 4) \gg 3 \quad (4)$$

$$DC2 = (P0 + P1 + P2 + \dots + P15 + 8) \gg 4 \quad (5)$$

$$DC3 = (P0 + P1 + P2 + \dots + P31 + 16) \gg 5 \quad (6)$$

식 (7)은 16x16 블록의 평면 모드 연산 수식이다. 8x8 블록의 평면 모드는 16x16 블록의 평면 모드의 연산 수식 안에 포함되므로 생략한다.

$$Pn16 \times 16[x, y] = Clip((a + b \times (x - 7) + c \times (y - 7) + 16) \gg 5), \quad (7)$$

$$\begin{cases} a = 16 \times (p[-1, 15] + p[15, -1]) \\ b = (5 \times H + 32) \gg 6 \\ c = (5 \times V + 32) \gg 6 \\ H = \sum_{x'=0}^7 (x'+1) \times (p[8+x', -1] - p[6-x', -1]) \\ V = \sum_{y'=0}^7 (y'+1) \times (p[-1, 8+y'] - p[-1, 6-y']) \end{cases}$$

[6]의 공통 연산기는 4개의 덧셈기와 1개의 쉬프트 연산기로 구성되며, 하나의 공통 연산기는 한 픽셀의 연산을 1 사이클에 처리한다. 따라서 4개의 공통 연산기를 사용하여 4x4 블록의 16개 픽셀을 연산하기 때문에 4x4 블록을 처리하는데 4 사이클이 소요된다. 즉 인트라 예측을 위해 16개의 덧셈기와 4개의 쉬프트 연산

기를 사용한다. [7]의 공통 연산기는 덧셈기 3개와 2개의 쉬프트 연산기로 구성되며, 평균치 모드 연산과 평면 모드의 복잡한 연산 수식을 빠르게 연산하기 위해 진처리 연산기를 사용한다. 하지만 [6]의 공통 연산기와 동일하게 하나의 공통 연산기는 한 픽셀을 처리하고, 4개의 공통 연산기를 사용하기 때문에 4x4 블록의 16개 픽셀을 처리하는데 4 사이클이 소요된다. 따라서 인트라 예측을 위해 12개+a개의 덧셈기와 (4+β)개의 쉬프트 연산기를 사용한다[6-7].

IV. 연산 수식 분석

인트라 예측의 방향성 예측 모드와 평균치 모드의 덧셈 연산 수식을 분석해 보면, (PX + PY + 1)의 형태의 수식으로 모든 인트라 예측 모드의 연산 수식이 아래와 같이 정리된다.

$$Dr0 = (P0 + P1 + 1) \gg 1 \quad (8)$$

$$Dr1 = (P0 + P1 + 1 + P2 + P3 + 1) \gg 2 \quad (9)$$

$$DC0 = (P0 + P1 + 1 + P2 + P3 + 1) \gg 2 \quad (10)$$

$$DC1 = (P0 + P1 + 1 + P2 + P3 + 1 + P4 + P5 + 1 + P6 + P7 + 1) \gg 3 \quad (11)$$

$$DC2 = (P0 + P1 + 1 + P2 + P3 + 1 + \dots + P12 + P13 + 1 + P14 + P15 + 1) \gg 4 \quad (12)$$

$$DC3 = (P0 + P1 + 1 + P2 + P3 + 1 + \dots + P28 + P29 + 1 + P30 + P31 + 1) \gg 5 \quad (13)$$

즉, 하드웨어로 설계시 덧셈기에 캐리를 입력받기 본 연산(PX + PY + 1)을 처리할 수 있으며, 캐리를 입력받아 더해진 값들을 다시 더하는 형태로 연산 수식을 처리할 수 있다. 즉 기존 공통 연산기에서 사용되었던 반올림을 위한 덧셈기가 따로 필요하지 않게 된다.

평면 모드의 $P_{in16 \times 16}[x,y]$ 수식 또한 Plane 전처리기를 사용하여, Seed값인 $a+b \times (x-7)+c \times (y-8)+16$ 을 입력받으면 그림 5와 같이 처리할 수 있다[9]. 즉, Seed 값에서 c 값을 더해준 후에 b, 2b, 3b 값을 더하여, 첫 번째 행의 연산을 수행한다. 마찬가지로 나머지 행들도 동일하게 seed값에 2c, 3c, 4c 값을 더한 후 b, 2b, 3b 값을 더하여 연산을 수행한다. 따라서 총 16개의 덧셈기와 16개의 쉬프트 연산기를 사용하면, 16개의 픽셀을 한번에 연산할 수 있다. 그 이후의 블록들의 연산은 레지스터에 저장된 이

전 블록의 계산된 픽셀 값들과 b, c 값을 더해줌으로써 1 사이클에 처리 가능하다. 인트라 예측의 17개의 모드를 연산하기 위해 필요한 OU 및 덧셈기, 쉬프트 연산기의 개수는 표 1과 같고, OU(Operation Unit)는 캐리를 입력받는 덧셈기를 의미한다.

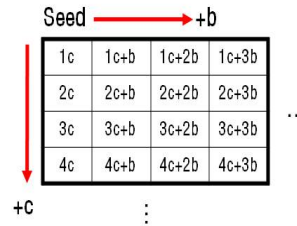


그림 5. 평면모드의 연산 수식 분석
Fig. 5 Decomposition of Plane prediction mode

표 1. 각 인트라 예측모드에서 요구되는 연산기 개수
Table. 1 Number of Operation Unit for Intra Prediction mode

Mode	OU	Adder	Total Adder (OU+Adder)	Shifter
Vertical	0	0	0	0
Horizontal	0	0	0	0
Diagonal down left	8	7	15	7
Diagonal down right	8	7	15	7
Vertical right	7	6	13	10
Horizontal down	7	6	13	10
Vertical left	6	5	11	10
Horizontal Up	4	3	7	6
DC0	2	1	3	1
DC1	4	3	7	1
DC2	8	7	15	1
DC3	16	15	31	1
Plane	0	16	16	16

방향성 예측 모드를 처리하기 위해 요구되는 연산 장치 개수는 OU 8개, 덧셈기 7개, 쉬프트 연산기 10개이며, 또한 Plane 모드에서는 덧셈기 16개, 쉬프트 연산기 16개가 요구된다. 평균치 모드의 경우 DC3 모드를 제외한 모든 예측 모드는 8개의 OU, 7개의 덧셈기, 1개의 쉬프트 연산기를 사용하면 연산 처리가 1 사이클에 가능하다.

V. T³(Three Type Transform) 연산기

본 논문에서 제안하는 T³(Three Type Transform) 연산기 구조는 그림 6과 같다. T³ 연산기는 8개의 OU, 8개의 덧셈기, 16개의 쉬프트 연산기로 구성되어 있으며, 4x4 블록의 16개 픽셀을 한 사이클에 병렬 연산한다. Normal 모드, 평균치 모드, 평면 모드로 동작한다. 내부에 Three Type Transform Switch가 내장되어 있어, 세 가지 모드에 따라 내부 구조가 바뀌게 된다.

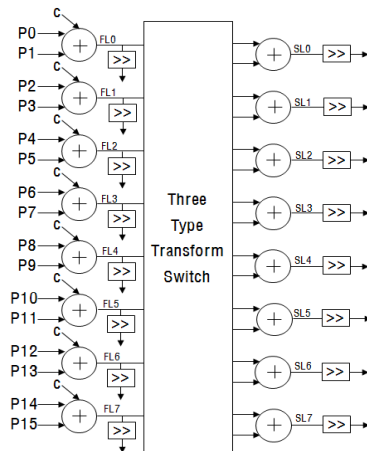


그림 6. T³(Three Type Transform) 연산기
Fig. 6 T³(Three Type Transform) Operation Unit

5.1. T³ 연산기의 Normal 모드의 내부 구조

T³ 연산기의 Normal 모드는 휘도 4x4 블록의 방향성 예측모드 Diagonal Down Left, Diagonal Down Right, Vertical Right, Horizontal Down, Vertical Left, Horizontal Up 모드를 연산하기 위해 설계 되었으며, 15개의 덧셈기와 15개의 쉬프트 연산기로 구성되어, 표 1에서 제시된 방향성 예측 모드에서 요구되어지는 기본 연산장치의 수를 만족한다. Normal 모드의 내부 구조는 그림 7과 같다.

Normal 모드는 FL(First Level)에서 8개의 OU 결과 값을 출력하며, SL(Second Level)는 연속된 2개의 OU 결과 값을 각각 더하여 7개의 덧셈 결과 값을 출력한다. 예를 들어 Vertical Left 예측모드를 수행할 경우, 그림 1에서 설명한 4x4 블록의 인접한 블록 A~G를 P0~11에 순서대로 입력한다. FL0~FL5에서 (A+B+1), (B+C+1), (C+D+1), (D+E+1), (E+F+1), (F+G+1) 값을 출력하고, SL0~SL5에서는 FL에서 출력된 2개의 값들을 각각 더

한 (A+2B+C+2), (B+2C+D+2), (C+2D+E+2), (D+2E+F+2), (E+2F+G+2) 값을 출력한다. 쉬프트 연산기들은 FL과 SL에서 출력된 값들을 Vertical Left 예측모드에서 요구하는 10개의 연산 값으로 변환한다.

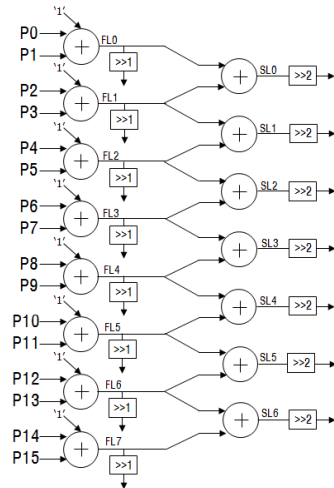


그림 7. T³ 연산기의 Normal 모드 내부 구조
Fig. 7 Architecture of T³ Operation Unit in Normal Mode

5.2. T³ 연산기의 평균치 모드의 내부 구조

T³ 연산기의 평균치 모드는 인트라 예측 모드에서 사용되는 4가지의 DC0, DC1, DC2, DC3 모드를 연산하며, 평균치 모드의 내부 구조는 그림 8과 같다. 평균치 모드의 경우 14개의 덧셈기와 6개의 쉬프트 연산기가 사용된다. 평균치 모드는 더해지는 픽셀 수에 따라 4가지로 분류되기 때문에, 처음에 입력된 OU 결과 값을 계속해서 더해주는 과정을 반복한다. 연산 수식 DC0은 FL0, FL1을 더한 SL0의 출력 값과 SL0에 연결된 길이가 2인 오른쪽 쉬프트 연산기를 이용하여 처리된다. 연산 수식 DC1은 FL0, FL1을 더한 SL0과 FL2, FL3을 더한 SL1을 이용하여 SL1을 얻은 후 길이가 3인 오른쪽 쉬프트 연산을 수행하여 처리된다. 연산 수식 DC2는 P0~P15 값과 캐리 값을 입력하여 SL0, SL2, SL4, SL6 값을 얻은 후 레지스터에 저장하고, 다음 사이클에 저장된 레지스터 값들을 P0~P3에 입력하여 SL0을 구한 뒤 오른쪽으로 4번 쉬프트 연산을 함으로써 처리된다.

연산 수식 DC3의 경우 P0~15와 캐리 값을 입력하여 SL1과 SL5 값을 얻은 후 레지스터에 저장한다. 다음 사이클에 P16~P31과 캐리 값과 입력하여 얻어진 SL1과

SL5를 레지스터에 저장한다. 레지스터 저장된 4개의 값을 P0~P3에 입력하여 SL0을 구한 뒤 오른쪽으로 5번 쉬프트 연산을 수행함으로써 처리된다.

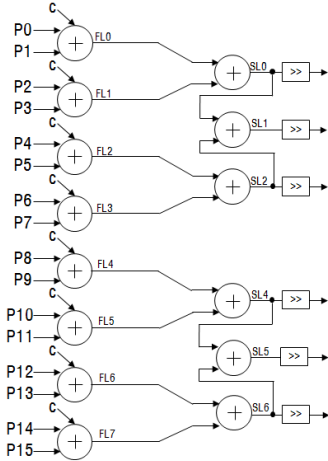


그림 8. T^3 연산기의 평균치 모드 내부 구조
Fig. 8 Architecture of T^3 Operation Unit in DC Mode

5.3. T^3 연산기의 평면 모드의 내부 구조

평면 모드는 그림 9와 같이 16개의 덧셈기와 쉬프트 연산기로 구성되며, seed값을 구한 후 c값과 b값을 차례대로 더하여 연산한다. 4x4 블록의 연산 결과 중 가장 오른쪽 픽셀 4개는 레지스터에 저장되어 다음 4x4블록을 연산할 때 이용된다.

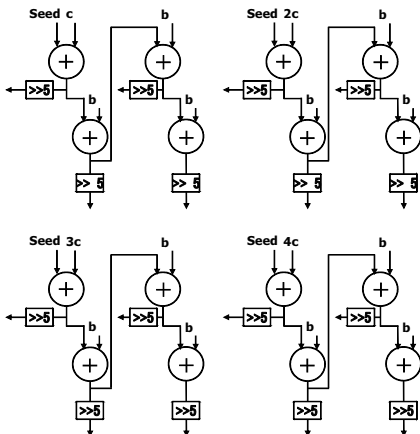


그림 9. T^3 연산기의 평면 모드 내부 구조
Fig. 9 Architecture of T^3 Operation Unit Plane Mode

연산 과정은 seed값에 c값을 더하여, 그림 5의 첫 번째 4x4 블록 (0,0) 픽셀을 연산한다. 그 후 b 값을 차례로 더하여 (0,1), (0,2), (0,2) 픽셀 값을 구한다. 마찬가지로 seed 값에 2c와 3c, 4c값을 더하여, (1,0), (2,0), (3,0) 픽셀 값을 구한 후, b 값을 차례로 더하여 (1,1), (1,2), (1,3), (2,1), (2,2), (2,3), (3,1), (3,2), (3,3)을 연산 하여, 4x4블록의 16개 픽셀 값을 연산한다.

VI. 인트라 예측기 구조

본 논문에서 제안한 인트라 예측기는 H.264/AVC 복호기를 위한 것으로서, 인트라 예측 모드의 정보와 오차 데이터 값을 이용해 연산을 수행한다. 인트라 예측기의 내부 구조는 그림 10과 같다. 인트라 예측기는 인트라 예측 제어기, 메모리, 레지스터 파일, T^3 연산 제어기, T^3 연산기, Plane 전처리 연산기, 4개의 레지스터로 구성된다.

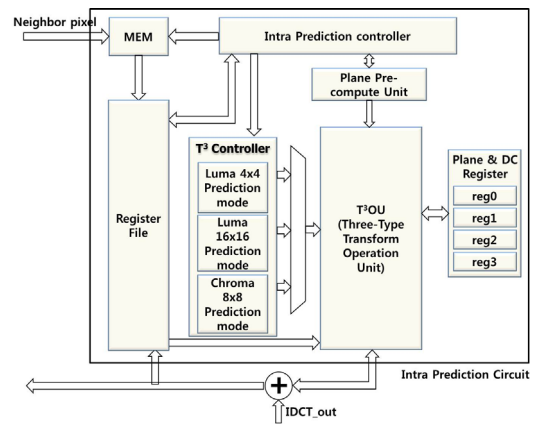


그림 10. 인트라 예측기 구조
Fig. 10 Architecture of Intra Predictor

인트라 예측 정보는 인트라 예측 제어기로 전송되고, 매크로 블록 단위의 인접한 픽셀 데이터는 메모리에 저장하며, 4x4블록 단위의 픽셀 데이터는 레지스터 파일에 저장한다. 인트라 예측 제어기는 T^3 연산 제어기를 통해 T^3 연산기를 제어하며, 인접한 픽셀 값들을 레지스터 파일로부터 가져와 연산을 수행한다. 또한 평면 모

드가 발생 했을 경우, Plane 전처리 연산기를 이용해 Seed값을 T³ 연산기에 제공해 준다. 4개의 레지스터는 평면 모드와 평균치 모드를 연산할 때 사용되며, T³ 연산기를 통해 연산된 결과는 IDCT에서 출력된 오차 데이터와 더해져서 인접한 픽셀 데이터를 생성한다. 생성된 픽셀 데이터는 다시 인트라 예측기의 메모리에 저장되고 레지스터 파일로 전송되기 때문에 T³ 연산기에 연속적으로 제공된다.

VII. 실험 결과

본 논문에서 제안하는 인트라 예측기는 Verilog HDL로 설계되었고 highway, coastguard, foreman, mother-daughter, trevor 등 영상을 이용하여 테스트 벡터를 추출 하였다. 제안하는 예측기의 검증을 위해 IDEC에서 지원하는 Xilinx사의 통합 설계도구인 ISE 9.2i와 Virtex-4 FPGA 디바이스를 사용하였다. ISE 9.2i를 이용하여 합성한 결과 제안하는 예측기는 59,297개의 게이트를 사용하고, 102MHz에서 동작한다. 표 2는 제안하는 인트라 예측기와 기존의 인트라 예측기들과 비교하고 분석한 결과를 나타낸 것으로, 기존 예측기들과 비교 했을 때 서로 다른 공정기술을 적용하였으므로 게이트 수를 직접적으로 비교 할 수 없다. 하지만 사이클 수를 비교할 경우, 제안하는 인트라 예측기의 성능은 기존 예측기들 대비 평균 58.95%의 향상된 결과를 얻는다.

VIII. 결 론

본 논문에서는 H.264/AVC 복호기를 위한 T³ 연산기가 내장된 인트라 예측기를 제안한다. 제안하는 T³ 연산기는 16개의 덧셈기와 16개의 쉬프트 연산기로 구성되며, 한 픽셀의 예측 연산을 수행하는 기존의 공통 연산기와는 달리 4x4 블록의 16개 픽셀 전체를 한 사이클에 처리한다. 또한 T³ 연산기는 Three Type Transform Switch가 내장되어 있어 Normal 모드, 평균치 모드, 평면 모드를 수행할 때 각 모드에 최적화된 연산기로 동작한다. 제안하는 인트라 예측기는 인트라 예측 제어기, 메모리, 레지

스터 파일, T³ 연산 제어기, T³ 연산기, Plane 전처리 연산기, 평균치 모드와 Plane 모드를 위한 4개의 레지스터로 구성된다.

인트라 예측 제어기는 레지스터 파일과 Plane 전처리 연산기, 내부 메모리를 제어하여 정해진 사이클 주기로 인트라 예측 모드와 인접한 픽셀 값을 T³ 연산기에 전달한다. Vertex-4 FPGA 디바이스에 제안하는 인트라 예측기를 구현한 결과, 동작 주파수는 102.8MHz, 게이트의 수는 59,297개이다. 제안하는 인트라 예측기와 기존의 인트라 예측 회로의 예측 수행 사이클 수를 비교하고 분석한 결과 평균 58.95%의 향상된 결과를 얻었다.

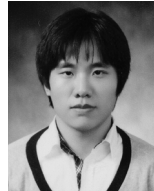
표 2. 인트라 예측기의 예측 연산 사이클 비교
Table. 2 Comparison of Intra Predictor computation cycle

Mode		Clock Cycles		
		[6]	[7]	본논문
공정기술		0.13um	0.18um	FPGA
주파수(MHz)		100.9	100	102.8
게이트 수		9,464	154,611	59,297
4x4	Vertical	4	1	1
	Horizontal	4	1	1
	DC	6	2	1
	Diagonal down left	4	4	1
	Diagonal down right	4	4	1
	Vertical right	4	4	1
	Horizontal down	4	4	1
	Vertical left	4	4	1
16x16	Horizontal up	4	4	1
	Vertical	64	16	16
	Horizontal	64	16	16
	DC	78	20	18
8x8	Plane	279	71	23
	Vertical	16	4	4
	Horizontal	16	4	4
	DC	18	8	4
Plane		87	21	9
Average		38.82	11.06	6.06

참고문헌

- [1] Joint Video Team(JVT), Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification. ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, March 2005.
- [2] 이찬호, "H.264 복호기를 위한 스케일러블 인트라 예측기 구조 설계," 대한전자공학회논문지, 제45권, 제11호, pp. 77-82, 2008년 11월.
- [3] 차기종, 류광기, "DCT 기반 인트라 예측 인코더를 위한 효율적인 하드웨어 설계," 한국정보통신학회 논문지, v.16, no.4, pp.765-770, 2012년 4월.
- [4] 서기범, "H.264 하이프로파일 인트라 프레임 부호화 기 설계," 한국정보통신학회논문지, v.15, no.11, pp.2285-2291, 2011년 11월.
- [5] 서기범, 이혜윤, 이용주, 김호의, "H.264 High-Profile Intra Prediction 모듈 설계," 한국정보통신학회논문지, v.12, no.11, pp.2045-2049, 2008년 11월.
- [6] 심재오, 이선영, 조경순, "공통연산부를 공유하는 H.264 디코더용 인트라 예측 회로 설계," 전자공학 회논문지, 제45권 SD편, 제9호, pp. 103-109, 2008년 9월.
- [7] 김옥, 류광기, "H.264/AVC 복호기를 위한 효율적인 인트라 예측기 하드웨어 구조," 전자공학회 논문지, 제47권 SD편 제5호, pp. 24-30, 2010년 5월.
- [8] 정제창, H.264/AVC 비디오 압축 표준, 홍릉과학출판사, 2005년 10월.
- [9] Yu-Wen Huang, Bing-Yu Hsieh, Tung-Chien Chen and Liang-Gee Chen, "Analysis, Fast Algorithm, and VLSI Architecture Design for H.264/AVC Intra Frame Coder," *IEEE Transactions on Circuits and Systems For Video Technology*, Vol. 15, No. 3, pp. 378-401, March 2005.

저자소개



김선철(Xianzhe Jin)

2007년 2월 배재대학교
정보통신공학과 공학사
2009년 2월 한밭대학교
정보통신공학과 공학석사

2009년 3월 ~ 현재 한밭대학교 정보통신공학과
박사과정

※관심분야: SoC 플랫폼 설계, 영상코덱 설계



류광기(Kwangki Ryoo)

1986년 2월 한양대학교 전자공학과
공학사

1988년 2월 한양대학교 전자공학과
공학석사

2000년 2월 한양대학교 전자공학과 공학박사
1991년~1994년 육군사관학교 교수부 전자공학과
전임강사

2000년~2002년 ETRI 시스템IC설계팀 선임연구원
2010년~2011년 Univ of Texas at Dallas 방문교수
2003년~현재 한밭대학교 정보통신공학과 교수

※관심분야: SoC 플랫폼 설계 및 검증, 하드웨어/
소프트웨어 통합설계 및 검증, 멀티미디어 코덱 설계