

# H.264/AVC 디코더용 인트라 예측기의 효율적인 하드웨어 구현

김형호(金亨浩), 유기원(俞基源)

三星電子 株式會社 디지털미디어 研究所

전화: (031) 200-3476 / 팩스: (031) 200-3147

## An Efficient Hardware Architecture of Intra Prediction in H.264/AVC Decoder

Hyung Ho Kim and Ki Won Yoo

Digital Media R&D center, Samsung Electronics Co., Ltd., Republic of Korea

E-mail: {lian.kim, ykiwon}@samsung.com

### Abstract

H.264/AVC is the upcoming video coding standard of ITU-T H.264 and ISO MPEG-4 AVC. The new standard can achieve a significant improvement up to 50% in compression ratio compared to MPEG-4 advanced simple profile. In this paper, we propose the novel intra prediction scheme to speed up intra prediction process in H.264/AVC decoder and show the hardware architecture for it. The proposed scheme uses the concurrent processing of the 4x4 intra prediction, which is based on that some 4x4 block pairs in a 16x16 luma block can be processed concurrently. The proposed scheme can reduce intra prediction time by 33 %.

### I. 서론

H.264/AVC는 ITU-T H.264와 ISO MPEG-4 Part10 AVC 그룹의 협력으로 구성된 JVT(Joint Video Team)에 의해 진행 중인 새로운 비디오코딩 표준으로, 2004년 3월 FDIS로 제정될 예정이다. 새로운 표준은 비디오의 코딩 성능과 비트스트림의 네트워크 적응력 향상을 목적으로 한다. 이를 위해 인트라/인터 예측방법 개선, 변동 블록 크기, 개선된 정수 변환/양자화, Deblocking 필터, MBAFF, CAVLC, CABAC, NAL의 새로운 코딩 방법이 추가 및 향상되었다. 이러한 개선된 코딩 방법으로 인해 H.264/AVC는 MPEG-4 advanced simple profile에 비해 동일한 화질에서 50%까지 비트량이 감소한다고 알려져 있다 [1]. 이런 장점으로, 현재 국내 지상파 디지털 멀티미디어 방송(Digital Multimedia Broadcasting: DMB) 비디오 규격으로 채택되어 있다. H.264/AVC의 인트라 예측과정은 휘도신호와 색차

신호에 대해 구별되어 진행된다. 영상 부호화 과정에서 인트라 예측은 매크로블록 또는 매크로블록 내의 16개 4x4블록들을 대상으로 한다. 인트라 예측 모드의 결정 과정은 각 모드 별로 대상 블록의 인접 화소들에서 계산된 예측 블록과 대상 블록의 잔류값(residue)의 절대합을 비용으로 하여, 최소의 비용을 갖는 모드를 찾는 것이다. 영상 복호화 과정에서 인트라 예측은 모드 정보와 인접 화소들에서 예측 블록을 생성한 후, 비트스트림에서 입력된 잔류 블록의 값을 더하여 블록을 재구성하는 것이다.

본 논문에서는 H.264/AVC의 인트라 예측 과정의 하드웨어 구현과 관련하여 처리 속도를 향상할 수 있는 블록 처리방법과 이를 구현하기 위한 복호기의 하드웨어 구조를 제안한다.

통상의 인트라 예측 과정에서 매크로블록은 16개의 4x4 블록으로 나누어질 수 있으며, 각 4x4 블록의 예측 과정에서 예측 블록의 생성을 위해서는 인접한 매크로블록 또는 현재 매크로블록 내의 인접 블록의 화소 정보를 필요로 한다. 매크로블록의 인트라 예측 과정에서 인트라4x4 모드인 경우에는 인트라16x16 모드와 같은 모든 블록들의 동시처리가 불가능하며, 표준에서 정의된 주사 순서에 따라 16단계의 순차적인 처리를 해야 된다 [2]. 제안하는 인트라 예측은 이러한 과정을 10단계로 축소할 수 있는 블록 처리 순서를 제안한다. 이것은 인트라4x4 모드에서 각 블록의 예측 블록을 발생하기 위해 필요한 인접 정보가 얻어지는 시점을 체크하여, 블록의 처리 단계별로 일부 블록들의 예측 블록이 동시에 계산될 수 있다는 관찰에 기반하여, 이를 하드웨어 구현에 적용하였다.

본 논문은 다음 장에서 H.264/AVC에서 정의되어 있는 인트라 예측 방법에 대해 설명하며, 3장에서는 본 논문에서 제안하는 인트라 예측의 하드웨어 구현 과정에서 속도를 향상한 방법을 설명한다. 4장에서는 구현된 하드웨어의 구조를 보이며, 5장에서는 4장에서

보인 하드웨어 구현 환경과 결과를 설명하며, 끝으로 6장에서 결론과 향후 연구를 제시한다.

## II. H.264/AVC의 인트라 예측

그림 1에서는 H.264/AVC의 인트라 예측 과정을 보인다. H.264/AVC의 인트라 예측은 휘도와 색차 신호를 입력으로 하며, 휘도 신호에 대해서는 4개의 16×16 예측과 9개의 4×4 예측이 있으며, 두 개의 색차 신호에 대해서는 휘도 신호의 16×16 예측과 유사한 4개의 8×8 예측이 있다. 이러한 인트라 예측 모드들은 인트라 예측 대상 블록을 기준으로 동일 프레임 내의 인접 화소들을 참조 화소로 이용하여 예측 블록의 계산에 사용한다.

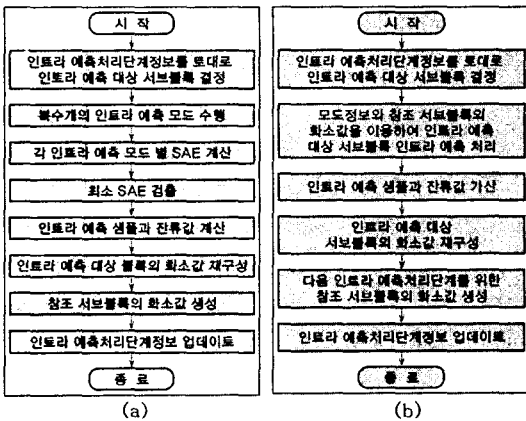


그림 1. 인트라 예측과정 (a) 부호화 (b) 복호화

그림 2에서는 인트라 예측에 필요한 인접 화소들을 보인다. 인접 화소들은 대상 블록의 왼쪽과 위쪽에 인접한 블록의 화소들에서 얻어진다. 모드 선택 과정에서 특정 모드의 예측에 필요한 인접 화소가 존재하지 않을 경우에는 그 모드에 대한 예측은 수행되지 않는다. 예측 과정에서 인접 화소가 존재하지 않는 경우는 대상 블록이 프레임의 왼쪽이나 위쪽의 가장자리에 놓여 있거나 인접 화소를 포함한 매크로블록의 주소값이 현재 매크로블록의 주소값보다 클 때이다.

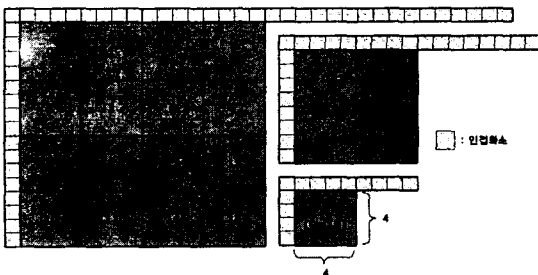


그림 2. 인트라 예측에 필요한 인접 화소들

인트라 예측 과정에서 16×16과 8×8 예측의 경우 예측 블록의 계산에 필요한 인접 화소들이 인접한 매크로블록에 놓여 있기 때문에 독립적인 처리가 가능하다. 그러나, 4×4 예측의 경우에는 각 블록이 동일한 매크로블록 내의 인접한 블록의 화소 정보를 필요로 하기 때문에 16개의 4×4 블록들에 대한 순차적인 처리가 요구된다.

그림 3에서는 휘도 신호에 대한 9개의 4×4 예측 모드를 보인다. A~M은 예측 블록의 발생에 필요한 인접 화소들이다.

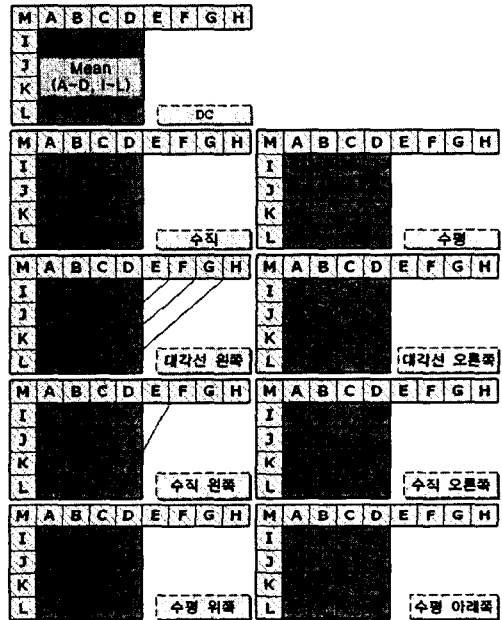


그림 3. 휘도신호에 대한 인트라 4×4 예측모드

## III. 제안하는 인트라 예측 처리방법

매크로블록에 대한 인트라 예측 과정에서 인트라 4×4 예측은 순차적인 처리 때문에 전체 예측 과정의 체중이 된다. 본 논문에서는 예측해야 될 블록을 결정하는 시점에서 예측에 필요한 인접 정보를 분석하여 각 블록이 처리되어야 하는 시점을 지정하여 처리속도를 향상할 수 있는 방법을 제안한다.

표 1은 각 4×4 블록의 예측을 위해 필요한 동일 매크로블록 내의 인접 블록을 보인다. 이때의 색인번호는 그림 2에서 나타낸 것을 따른다. 표 1로부터 2번째 블록과 4번째 블록은 블록 1의 예측이 완료되는 시점에서 동시 처리가 가능하며, 이 외에도 {3,5}, {6,8}, {7, 9}, {10,12}, {11,13}의 블록들 간에도 동시 처리가 가능하다.

표 1. 4x4 블록의 예측에 필요한 인접 블록

대상 4x4 블록의 색인번호	대상 블록의 예측을 위해 필요한 인접 블록의 색인번호	대상 4x4 블록의 색인번호	대상 블록의 예측을 위해 필요한 인접 블록의 색인번호
0	...	8	2, 3
1	0	9	2, 3, 6, 8
2	0, 1	10	8, 9
3	0, 1, 2	11	8, 9, 10
4	1	12	3, 6, 7, 9
5	4	13	6, 7, 12
6	1, 3, 4	14	9, 11, 12, 13
7	4, 5, 6	15	12, 13, 14

표 2에서는 이러한 분석에 근거하여 인트라 4x4 예측모드에서 블록들의 처리 순서를 보인다. 제안된 방법은 하드웨어 구현에서 동시처리를 이용하여 10단계에서 하나의 매크로블록을 처리한다.

표 2. 예측 단계에 따른 블록의 처리 순서

인트라 4x4 예측 처리 단계	예측 대상 블록	인트라 4x4 예측 처리 단계	예측 대상 블록
1	0	6	7, 9
2	1	7	10, 12
3	2, 4	8	11, 13
4	3, 5	9	14
5	6, 8	10	15

#### IV. 하드웨어 구조

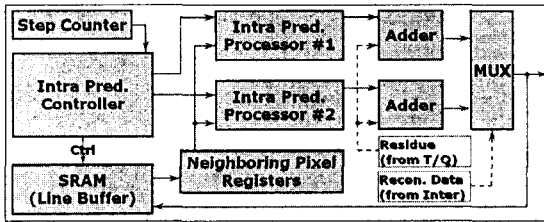


그림 4. 인트라 예측을 위한 하드웨어 블록도

그림.4는 제안된 하드웨어 구조 블록도를 나타낸다. 서론에서 설명한 것처럼 인트라 예측에는 휘도16x16, 휘도4x4, 색차 단위로 모드가 존재한다. 그러나 위 구조는 4x4블록 단위로만 예측 수행할 수 있다. 이는 4x4 블록을 16번 수행하면 매크로블록이 되고 4번 수행하면 색차 블록 크기만큼 수행하는 것과 동일한 효과를 낼 수 있다는 데에서 착안하였다.

인트라 예측일 때 위 예측기는 대기 상태로 놓여있지만 MUX 블록과 인접 화소 레지스터 블록은 인트라/인트라 예측과 상관없이 항상 재구성된 화소를 저장해야 한다.

하드웨어 구조의 동작 순서를 살펴보면 다음과 같다. 먼저 스텝 카운터가 인트라 4x4 예측이 수행되는 단계를 결정하여, 컨트롤러에 전달한다. 컨트롤러는 각 단계에서 처리해야 하는 블록을 선정하며, 각 인트라 예측 처리기에 동작 신호와 해당 블록의 색인 번호를 전달한다. 인트라 예측 처리기는 지정된 블록에 필요한 인접 화소 정보를 이용하여 인트라 예측을 수행한다. 예측값은 덧셈기에서 잔류값과 더해지는데 이것이 재구성된 화소가 된다. 재구성된 화소는 인트라

예측기의 출력입과 동시에 다음 스텝에 필요한 인접 화소가 되며, 또한 다음 라인 매크로블록에 필요한 인접 화소이기도 하다. 그러므로 재구성된 화소는 출력됨과 동시에 인접 화소 저장기와 라인버퍼에 저장된다. 이와 같은 과정을 거쳐서 예측이 수행된다. 하드웨어 블록들의 역할과 기능을 세부적으로 살펴보자.

#### IV-1. 인트라 컨트롤러

인트라 모드일 때 예측에 필요한 정보에 맞게 제어 하는 블록이다. 또한 인접 화소의 이용가능여부에 따라 예측결과가 달라지는 모드가 있기 때문에 이것을 판별해야 한다. 휘도16x16 모드와 색차 모드일 경우에는 주변 매크로 블록에 관한 이용 가능여부가, 휘도4x4모드는 주변 4x4 블록에 관한 것이 필요하다. 그래서 인트라 컨트롤러에서 이 정보들을 다른 블록들로 출력해줘야 한다.

위 구조는 빠른 연산을 위해 외부 메모리를 사용하지 않으므로 각 스텝에서 필요한 인접 화소들을 내부 메모리에 저장한다. 이 내부 메모리의 주소를 생성하는 것 역시 컨트롤러의 몫이다.

#### IV-2. 스텝 계산기

제안하는 구조는 휘도4x4 모드를 연산하는 것을 기본으로 설계하였다. 그렇기 때문에 같은 구조로 4x4블록과 매크로블록을 모두 계산할 수 있다. 이와 같은 이유로 휘도4x4 모드나 휘도16x16 모드 역시 스텝계산기에 의해 정해진 계산 순서대로 예측 연산된다.

#### IV-3. 인트라 처리기

인접 화소를 이용해 실제 인트라 예측을 수행하는 블록이다. 내부는 휘도, 색차전용 예측기로 나뉘어지며 예측모드에 맞게 연산된다. 휘도4x4모드와 휘도16x16 모드를 계산하기 위하여 각각 처리기가 따로 있는 게 아니라 제안하는 하드웨어만으로 모든 인트라 예측을 계산한다. 이를 위하여 병렬 처리 구조가 적용된다. 만약 본 논문에서 제안하는 구조를 사용하지 않는다면 하나의 매크로블록을 예측하는 데 391클록이 소비된다. 그러나 위 구조를 사용하면 265클록이 소비되어, 시간적으로 33%의 감소 효과가 있다. 병렬 처리를 위해 같은 구조 2개가 필요하다.

인트라 예측을 4x4 처리유형을 4가지로 나눠서 하드웨어로 구현하였다. 첫째, DC 모드, 둘째 Plane 모드, 셋째 수직, 수평 모드, 넷째 위 세가지 유형을 제외한 나머지 모드이다. 이 중에서 수직, 수평 모드는 인접 화소를 단지 임시 저장한 후 출력하기 때문에 따로 구조가 필요 없다. 그렇기 때문에 3가지를 하드웨어로 구현하였다.

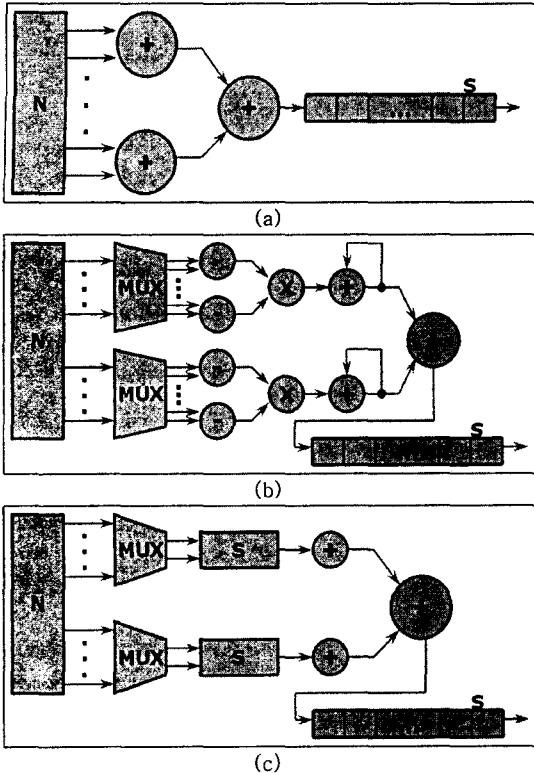


그림 5. 인트라 예측 아키텍처: (a) DC모드 (b) Plane모드 (c) DC, plane, 수직, 수평모드를 제외한 모드 (N: 주변화소 저장용 레지스터, S: 전이 레지스터) 구조의 동작은 인접 화소 이용 가능여부에 따라, (a)에서 수직, 수평 화소별로 더한 결과를 전이하여 인트라 예측을 수행하며, (b)에서는 수직, 수평 화소로 뿔썩한 후 곱한 값을 계속 덧셈한 결과를 전이하며 (c)에서는 수직, 수평 화소를 전이한 후 더한 결과를 다시 전이하여 인트라 예측한다.

IV-4. 주변화소 저장기

인트라 예측 처리기에 필요한 인접 화소를 저장하는 블록이다. 그런데 휘도16×16모드나 휘도4×4 모드, 색차모드가 필요로 하는 참조화소 개수는 다르기 때문에, 휘도16×16 모드를 위해서는 수직 인접 화소 16개를, 휘도4×4 모드를 위해서는 수직 주변화소 4개와 수평 인접 화소 4 개를, 색차 모드를 위해서는 CB,CR 성분이 필요로 하는 수직 인접 화소 8개를 각각 따로 저장할 수 있는 구조를 선택하였다.

IV -5. mux

P,B 슬라이스에는 인터/인트라 매크로 블록이 혼재해 있기 때문에 비록 인터모드로 예측되었다 할지라도 다음 인트라 예측을 위해 인터 예측값을 인접 화소로 사용해야 할 경우도 발생한다. 이를 위해 인터 예측값을 입력 받고자 mux를 사용하였다.

IV -6. 라인 버퍼

인트라 예측을 해야 할 블록이 수평 라인이 바뀌었을 경우를 위해 영상 폭에 해당하는 재구성된 값을 저장시켜야 한다. 라인버퍼는 SRAM으로 사용하여 현재 수평라인에 해당하는 화소를 저장하여 인트라 예측에 필요한 것을 사용할 수 있게 하였다.

V. 구현 및 결과

표 3은 하드웨어 스펙, 표4는 일반구조와 제안된 구조의 인트라 예측 시간을 하나의 매크로블록 기본으로 비교한 것이다. 인트라 예측시간이 33% 정도 절약되었다. 병렬처리 구조이기 때문에 인트라 예측 구조 크기가 약 40%정도 커졌지만 빠른 연산처리를 중요시하는 실시간 재생 하드웨어에서는 시간이 중요하기 때문에 본 논문에서 제안하는 구조에 적용하였다.

표 3. 하드웨어 스펙

GPU	ARM 920T
외부 메모리	64 MB(SDRAM)
시스템 버스 클럭	27MHz
FPGA	Xilinx Vertex2 XCV6000
FPGA 클럭	27MHz

표 4. 결과 비교

	일반 구조	제안된 구조
참조화소 크기(FPGA 점유율)	59840 LUTs(88%)	61883(91%)
인트라 예측기 크기(FPGA 점유율)	6873 LUTs(10%)	9765 LUTs(14%)
인트라 예측 시간(클럭 기준)	391 클럭	265 클럭

VI. 결론

본 논문에서는 H.264 인트라 예측을 복호화하는 하드웨어 구조를 구현하여 인트라 예측에 약 33%의 시간을 절약할 수 있었다. 또한, 현재 세계 최초 지상파 디지털미디어 방송 수신기에 구현되었다. 앞으로 H.264를 채택하여 디지털 방송을 추진중인 중국형 디지털 TV와 디지털 캠코더, PDA 와 같은 개인용 멀티미디어 휴대장치에 응용할 수 있다. 또한 인터넷을 이용한 VOD서비스나 자동차를 위한 항법장치, 텔레메틱스에도 응용할 여지가 충분히 있다. 이를 위해서 고품질 방송에도 적용될 수 있게 영상크기를 고려한 설계가 진행되고 있다.

참고문헌

[1] T. Wiegand, G. J. Sullivan, G. Bjntegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560-576, July 2003  
 [2] Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification