
H.264/AVC용 면적 효율적인 인트라 프레임 디코더 설계

정덕영* · 손승일*

Area-efficient Design of Intra Frame Decoder for H.264/AVC

Duck Young Jung* · Seung Il Sonh*

이 논문의 일부는 2006년 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(KRF-2006-521-D00339)

요 약

H.264/AVC는 ITU-T 비디오 코딩 전문가 그룹과 ISO/IEC MPEG 그룹에 의해 제안된 최신 비디오 코딩 표준이다. 최근 DMB와 멀티미디어 장비들이 비디오 압축 표준으로 H.264/AVC를 채택하고 있다.

본 논문에서는 메모리 사용을 최소화하고, 칩 면적을 최소화할 수 있는 H.264/AVC의 인트라 프레임 디코더를 제안한다. 제안한 인트라 프레임 디코더는 하드웨어 설계 언어인 VHDL로 기술하여 model_sim을 사용하여 시뮬레이션을 수행하였다. 그리고 FPGA칩인 XCV1000E에 다운로드하여 칩 레벨에서 설계된 H.264/AVC의 인트라 프레임 디코더를 검증하였다.

ABSTRACT

H.264/AVC is newest video coding standard of the ITU-T Video coding Experts Group and the ISO/IEC Moving Picture Experts Group. Recently H.264/AVC has been adopted as a video compression standard in DMB and multimedia equipments.

In this paper, we propose a H.264/AVC intra frame decoder which can minimize the memory usage and chip size. The proposed intra frame decoder is described in VHDL language and simulated in model_sim. It was verified in chip level by downloading to XCV1000E FPGA chip.

키워드

Intra Frame, CAVLC, VHDL, Zigzag Scan, Inverse Quantiser, Inverse Transform

I. 서 론

최근 영상에 대한 관심이 높아지는 가운데 DMB 핸드폰, PMP, 디지털 TV 등과 같은 영상 관련 제품에 더 많은 영상과 데이터를 효율적으로 전송하고 저장할 수 있는 압축 방식에 대한 연구가 활발히 진행되고 있다. 이와 관련

하여 영상과 관련된 압축 방식중의 하나이며 지상파 DMB와 IT관련 분야의 장비들이 표준으로 채택한 H.264/AVC는 ITU-T의 권고안 H.264와 ISO/IEC의 국제 표준인 144960-10(MPEG-4 part10) Advanced Video Coding에 의해 표준으로 채택되었다.

본 연구에서는 엔트로피코딩(CAVLC) 단계에서 사용

되는 메모리 접근 횟수를 최소화한 알고리즘과 표준안에서 제시한 9가지 Intra 모드를 모두 사용하고 각 단계에서의 수행을 최소화한 결과를 비주얼 C++언어를 이용하여 성능평가를 실시하여 최적화 하고 H.264/AVC 인트라 프레임 디코더의 입력 데이터를 도출하였다. 도출된 결과를 토대로 Xilinx ISE 6.2i 툴을 이용하여, VHDL 언어로 설계하였고 영상의 크기는 176*144의 QCIF를 사용하였다 [1-2].

II. H.264/AVC 인트라 프레임 디코더

2.1. CAVLC 디코더

CAVLC 디코더에서는 총 5단계로 구성되며 첫 번째 단계인 coeff_token 단계는 4개의 테이블 중에서 이웃 블록들의 total_coeff 개수에 따라 선택된 데이터를 이용하여 4x4블록의 '0'이 아닌 계수의 개수와 trailing_ones의 개수를 찾고, trailing_ones 단계에서 trailing_ones 값에 따라 한 비트씩 읽어 0이면 -1, 1이면 +1로 복원한다. 그 다음 level 단계에서는 입력 데이터와 점두사와 점미사로 구성된 각 레벨에 대한 코드를 비교하여 복호화 한다.

total_zeros 단계에서는 total_coeff 값에 따라 선택된 테이블 값과 입력 데이터를 비교하여 total_zeros를 찾는다. 마지막 단계인 run_before 단계에서는 total_zero에서 구한 zero의 개수에 따라 7개의 테이블 중에서 선택하여 입력 데이터와 비교한 결과에 따라 run_before값을 찾아 데이터를 복원한다[3-4].

2.2. 역변환 및 역양자화

역변환에서 pre-스케일링 계수(계수 위치에 따라 a2, ab, b2 값을 포함하고 있는 행렬 E_i로부터)는 반올림 에러를 방지하기 위한 상수스케일링 계수 64와 함께 이 연산에 통합되며, W'는 핵심 역변환 CiTW'Ci에 의해 변환된 스케일링 계수로 (식1)과 같다. 역변환의 출력 값은 스케일링 계수를 제거하기 위해 (식2)를 이용하여 디코더에서 스케일링을 수행한다.

$$W' = Z * Qstep * PF * 64 \quad (식1)$$

$$W = Z * V * 2 \text{floor}(QP/6) \quad (식2)$$

스케일링 수행으로 얻어진 W'와 핵심 역변환 행렬을

이용하여 (식4)와 같이 역변환을 수행하며, 역변환을 통하여 얻어진 X'를 이용하여 (식5)와 같은 Post 스케일링을 수행하고 4x4 오차 샘플의 출력이 생성된다[3][5].

$$X' = CiTW'Ci \quad (식4)$$

$$X'' = (X' + 32) \gg 6 \quad (식5)$$

2.3. 역 Intra 예측 모드

각 모드에서 복원된 휘도 샘플 P는 각각의 4x4 블록 또는 16x16 매크로 블록에 대해 생성되며, 디코더에서는 입력된 예측 모드에 따라 수행하여 복원된 데이터와 현재의 블록을 더한다. 그림1은 각 모드에 대한 예측 방향을 보여주며, 표1은 각 모드에 대한 수행을 보여주고 있다[2][5].

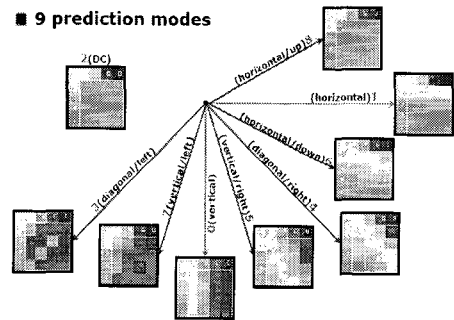


그림 1. 인트라 4x4 예측 모드
Fig. 1. 4x4 intra prediction modes

표 1. 9개의 인트라 4x4 예측모드 샘플 명명
Table. 1 Samples for nine 4x4 intraprediction modes

모드 0 (Vertical)	위쪽 샘플 A,B,C,D가 수직으로 확장
모드 1 (Horizontal)	왼쪽 샘플 I,J,K,L이 수평으로 확장
모드 2 (DC)	P의 모든 샘플은 샘플 A...D 그리고 I...L의 평균으로 예측
모드 3 (Diagonal Down-Left)	샘플은 좌측 하단과 우측 상단 사이의 45° 각도로 보간
모드 4 (Diagonal Down-Right)	샘플은 우측 하단으로 45° 각도로 확장
모드 5 (Vertical-Right)	수직의 오른쪽 약 2.66° 각도로 확장 (넓이/높이=1/2)
모드 6 (Horizontal-Down)	수평의 아래 약 2.66° 각도로 확장
모드 7 (Vertical-Left)	수직의 왼쪽 약 2.66° 각도로 확장 (또는 보간)
모드 8 (Horizontal-Up)	수평의 위쪽 약 2.66° 각도로 보간

III. H.264/AVC Intra frame decoder 설계

본 논문에서 설계한 H.264/AVC 인트라 프레임 디코더는 MC(Motion compensation)과 필터를 제외하고, CAVLC 디코더 모듈, 역변환 및 역양자화 모듈 그리고 역 Intra 예측 모듈로 구성되어 있다.

입력 데이터는 인코더에서 4x4 블록단위로 처리된 bitstream 데이터를 16비트 단위로 전송되며, 전송된 데이터를 입력받아 CAVLC 디코더 모듈에서 변환과 양자화된 12비트의 cavlc_o이란 데이터로 복원한다. 복원된 데이터는 QP 값과 함께 역변환과 역양자화 모듈을 통하여 intra 코딩된 12비트의 데이터로 복원되며, intra decoder에서 intra 모드 값에 따라 영상 데이터로 복원한다. 그림2는 H.264/AVC Intra frame decoder 모듈의 입·출력 인터페이스를 보여 주고 있다.

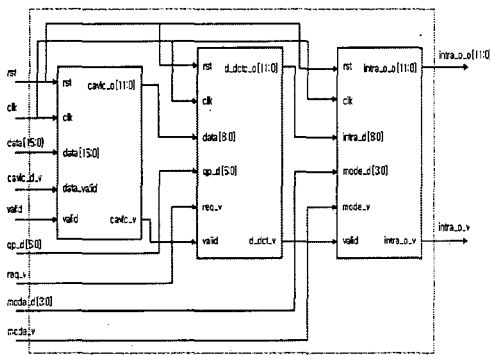


그림 2. 입·출력 인터페이스
Fig. 2. Input and output interfaces

3.1. CAVLC 디코더 설계

CAVLC 디코더는 인코더에서 처리한 순서로 복원하며 이는 coeff_token, trailing_ones, level, total_zeros, run_before의 총 5단계로 이루어져있고 coeff_token과 total_zeros 그리고 run_before에서 사용되는 LUT의 규칙성을 찾아 total_zeros에서 사용하는 LUT의 70%를 제외하고 모든 LUT를 사용하지 않는 알고리즘으로 CAVLC 디코더를 설계하였고 전체 블록도를 그림3에서 보여주고 있다.

슈프터를 통해 16 비트 단위로 입력받은 데이터를 최초 1을 만나기 전까지의 0의 개수를 구하고 그 이후부터 0의 개수와 함께 coeff_token으로 전송하고, 이 단계에서 입력받은 데이터를 이용하여 coeff_token과 trailing_ones를

구하고 이를 trailing_ones 단계와 level 단계로 전송한다. 이 두 단계에서 실제 데이터를 전송하며, total_zeros 단계를 수행하고 그 결과를 run_before에서 입력받아 처리한다. run_before 단계에서의 처리결과로 CAVLC_data 단계에서 CAVLC 복원 데이터를 전송한다. 이 모든 단계의 처리에 관한 제어는 컨트롤러에서 처리한다.

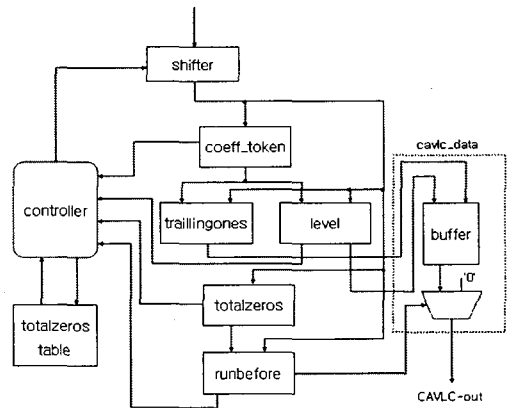


그림 3. 입·출력 신호도
Fig. 3. Signals for inputs and outputs

3.2. 역변환 및 역양자화 설계

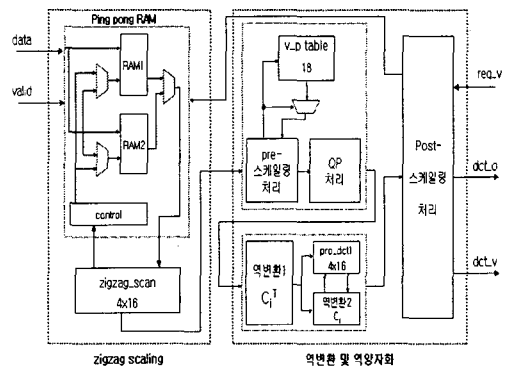


그림 4. 역변환 및 역양자화 전체 블록도
Fig. 4. Block diagram of inverse transform and quantization

CAVLC 디코더에서 수행하여 처리된 데이터를 입력받아 pingpong RAM을 이용하여 역변환과 역양자화 수행시 발생하는 지연으로 인한 데이터 손실을 방지하고, 저장된 데이터를 역변환과 역양자화를 수행하기 위해 zigzag_scan 처리를 통해 재배열 한다. 처리된 데이터를

이용하여 스케일링 처리를 수행하고 역변환을 수행하여 데이터를 복원한다. 역변환과 역양자화에서 연산 처리시 발생하는 지연을 최소화하고 고속으로 연속 처리하기 위해 파이프라인 방식을 사용하였다. 그림4는 역변환 및 역양자화의 전체 블록도를 보여주고 있다.

3.3. 역 인트라 예측모드 설계

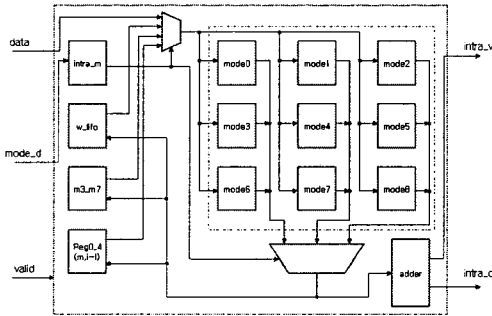


그림 5. 역 인트라 4x4 예측모드 전체 블록도
Fig. 5. Overview of inverse intra 4x4 prediction mode

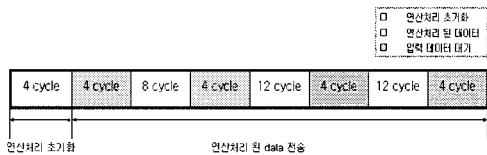


그림 6. 역 인트라 예측모드 수행 사이클
Fig. 6. Execution cycles of inverse intra prediction

역 인트라 4x4 예측모드에서는 전송된 예측모드에 따라 수행하고 수행된 결과와 전송된 데이터를 가산기를 통해서 덧셈을 수행하여 복원된 데이터를 전송한다. 첫 데이터를 입력받고 4 cycle 동안 초기화를 수행한 후 4 cycle 동안 연산 처리된 결과를 전송하고 이후의 처리는 입력 데이터의 대기 시간을 제외하고 입력과 동시에 처리되어 4x4 블록을 처리하는데 52 사이클이 소요된다. 그림5는 역 인트라 4x4 예측 모드의 전체 블록도를 보여주고 있으며, 그림6은 역 인트라 예측모드의 수행 사이클을 보여주고 있다.

IV. 설계 결과

4.1. CAVLC decoder의 메모리와 접근 횟수 비교

본 논문에서 설계된 H.264/AVC 인트라 프레임 디코더의 면적의 효율적 사용 여부와 메모리 접근 횟수의 감소 여부를 판별하기 위해 논문[6],[7]과 메모리 사용 여부를 비교하고, 본 논문에서 제안한 알고리즘과 테이블 참조 알고리즘의 수행 사이클을 비교하였다. 표2에서 보는 바와 같이 논문[6]과 논문[7]에서 사용한 테이블 중에서 본 논문에서는 Totla_zeros의 테이블만 70% 사용하여 면적의 효율적 사용을 확인할 수 있다. 또한 그림7과 같이 테이블을 참조한 알고리즘보다 50~70%정도의 수행 사이클 감소를 통해 고속 처리가 가능함도 알 수 있다[6-8].

표 2. [6]과[7] 그리고 제안한 알고리즘 결과 비교
Table. 2 Result comparison among paper [6], [7] and the proposed algorithm

	Table 사용				RAM 사용	Gates	Critical delay	Frequency (MHz)
	Coef_token	Total_zeros	Level	Run_before				
논문[6]	O	O	O	O	O	6100	8ns	125
논문[7]	O	O	O	O	X	9943(4720)		175
Proposed Design	X	O(70%사용)	X	X	X	1,7389	6,485ns	154,214

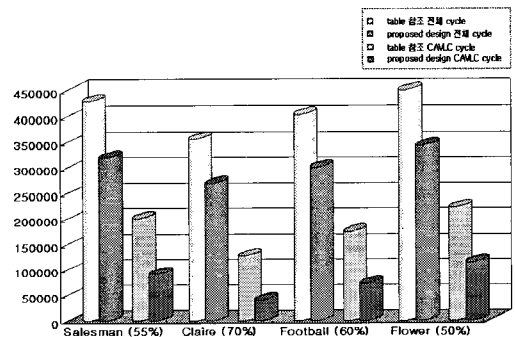


그림 7. 제안한 알고리즘과 테이블 참조 알고리즘 사이클 비교
Fig. 7. Cycle comparison between the proposed and the table reference algorithms

4.3. QP값에 따른 출력영상

C언어를 이용해서 설계된 H.264/AVC 인트라 프레임 디코더의 압축률에 따른 성능을 판별하기 위해 4개의 표준영상에 대해 QP값의 따른 복원된 영상을 확인하였다. 176*144의 표준영상을 QP=4,8,16,24로 압축할 경우에 따라 검증하였다. 그 결과를 그림8에서 보여주고 있다.

또한 통상적으로 30db 이상의 영상은 인간의 시각 특성상 화질 저하로 판단하기 쉽지 않은 영상인데, 표준 영상 4개에 대한 PSNR 비교 결과 QP = 16일 때까지는 30db 이상으로 화질 저하를 느끼지 못하며 QP = 24일때도 26~30db 사이로 많은 화질 저하를 느끼지는 못함을 PSNR을 통해 확인 하였다. 그림9는 이미지별 PSNR 비교 결과를 보여주고 있다.

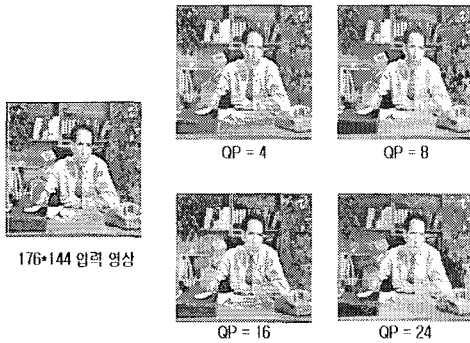


그림 8. 세일즈맨
Fig. 8. salesman

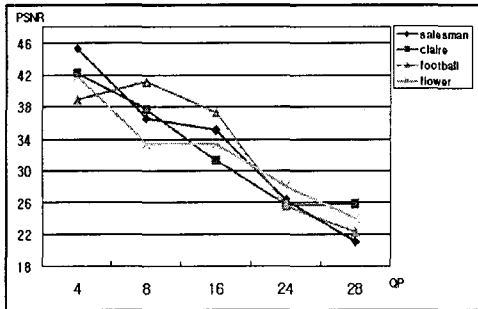


그림 9. 이미지별 PSNR 비교
Fig. 9. PSNR comparison for 4 images

V. 결 론

H.264/AVC 인트라 프레임 디코더에서 사용되는 메모리를 본 논문에서 제안한 알고리즘을 적용하여 설계한 결과 기존 논문[6],[7]과 비교했을 때 면적의 효율적 사용했으며, 제안한 알고리즘을 통해 수행 사이클이 50~70%정도 감소하여 고속처리가 가능함을 알 수 있다. 또한 역양자화와 역변환에서 파이프라인 방식을 사용하여 설계함으로써 연산 처리를 고속으로 순차처리가 가능하게 하였다.

본 논문에서 설계한 H.264/AVC 인트라 프레임 디코더는 적은 칩 면적과 고속 처리를 함으로써 실시간으로 방송하는 지상파 DMB용 핸드폰이나 PMP 그리고 각종 핸드폰에 적용 가능하다.

참고문헌

- [1] T.Wiegand, Smdy of Final Committee Draft of Joint Video Specification Draft 2, Doc. JVT-FIOO d2, Joint Xdeo Team (IVT) of .ISO/IEC MF'EG & ITU-T VCEG Dec. 2002.
- [2] Thomas Wiegand, Gray J. Sullivan, Gisle Bjontegaard, and Ajay Luthra, "Overview of the H.264/AVC Video Coding Standard" IEEE Trans. Circuits and systems for video technology, vol.9, pp. 287-290, July. 2003.
- [3] Iain E.G Richardson, "H.264 and MPEG-4", WILEY, 2003.
- [4] GBjontcgaard and K. Lillcvold. Contest-adaptive VLC(CAVLC)coding of coefficients, Doc.JVT-028, JVT of ISO MPEG & ITU VCEG 3' Meeting, Rairfas. Virginia, USA, May. 2002.
- [5] Joint Video TEam(JVT) of ISO/IEC MPEG & ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 AND ITU-T SG16 Q.6) 8th Meeting: Geneva, Switzer land, 23-27 May, 2003.
- [6] Wu Di, Ga0 Wen, Hu Minazeng, Ji Zhenzhou, "A VLSI Architecture Design of CAVLC DEcoder", IEEE. 2003
- [7] Hsiu-Cheng Chang, Chien-Chang Lin, and Jiun-In Guo, " A Novel Low-Cost High_Performance VLSI Architecture for MPEG-4 AVC/H.264 CAVLC Decoding, IEEE. 2005.
- [8] Yong Ho Moon, Gyu Yeong Kim, jae Ho Kim, Member, IEEE, "An Efficient Decoding of CAVLC in H.264/AVC Video Coding Standard", IEEE. 2005.

저자소개



정 덕 영(Duck-Young Jung)

2005년 한신대학교 정보통신학과
(학사)

2005년~현재 한신대학교 정보통신
학과 석사과정

※관심분야: 영상처리 프로세서 설계, ASIC 설계



손 승 일(Seung-II Sonh)

1989년 연세대학교 전자공학과(학사)

1991년 연세대학교 대학원 전자공학과
(석사)

1998년 연세대학교 대학원 전자공학과 박사

2002년~현재 한신대학교 정보통신학과 부교수

※관심분야: ASIC 설계(네트워크, 영상 칩 설계)