
H.264/AVC 인코더용 파이프라인 방식의 변환 코딩 및 양자화 코어 연구

손승일*

A Study on Pipelined Transform Coding and Quantization Core for H.264/AVC Encoder

Seungil, Sonh*

이 논문은 한신대학교 학술 연구비 지원에 의하여 연구되었음

요 약

H.264/AVC는 부호화되는 잉여 데이터의 유형에 따라 3개의 변환을 사용할 수 있다. 4x4 DCT 변환은 항상 수행되며, 16x16 인트라 모드인 경우에는 추가적으로 휘도 DC 계수에는 4x4 하다마드 변환을 수행하고, 색채 DC 계수에는 2x2 하다마드 변환을 수행한다. 변환 코딩을 완료한 이후에 한층 더한 데이터 압축을 위해 양자화가 수행된다. 본 논문에서는 H.264/AVC에 중요한 역할을 하는 DCT 변환, 하다마드 변환 및 양자화에 대한 하드웨어적인 구현에 대해 연구하였다. 특히 파이프라인 기법을 적용하여 33클럭의 대기 지연시간 이후에는 매 클럭 당 1개의 양자화된 결과를 출력할 수 있는 아키텍처를 제안하였다. 제안한 아키텍처는 Verilog HDL로 코딩되고, Xilinx 7.1i ISE를 사용하여 합성하고 검증하였다. 합성 결과 SPARTAN3S-1000 디바이스에서 동작 주파수는 106MHz이다. 1920x1080 HD 영상 프레임의 경우 최대 33프레임을 처리할 수 있다.

ABSTRACT

H.264/AVC can use three transforms depending on types of residual data which are to be coded. H.264/AVC always executes 4x4 DCT transform. In 16x16 intra mode only, 4x4 Hadamard transform for luma DC coefficients and 2x2 Hadamard transform for chroma DC coefficients are performed additionally. Quantization is carried out to achieve further data compression after transform coding is completed. In this paper, the hardware implementation for DCT transform, Hadamard transform and quantization is studied. Especially, the proposed architecture adopting the pipeline technique can output a quantized result per clock cycle after 33-clock cycle latency. The proposed architecture is coded in Verilog-HDL and synthesized using Xilinx 7.1i ISE tool. The operating frequency is 106MHz at SPARTAN3S-1000. The designed IP can process maximum 33-frame at 1920x1080 HD resolution.

키워드

H.264/AVC, DCT, 하다마드 변환, 양자화, 파이프라인, 인트라 예측

Key word

H.264/AVC, DCT, Hadamard Transform, Quantization, Pipeline, Intra Prediction

* 증신회원 : 한신대학교 (교신저자 : saisonh@hs.ac.kr)

접수일자 : 2011. 08. 10

심사완료일자 : 2011. 09. 22

I. 서 론

H.264/AVC 비디오 압축은 비디오 압축을 위하여 비교적 최근에 개발된 국제표준이며, 전송률-왜곡(Rate-distortion efficiency)에서 기존의 비디오 압축 알고리즘보다 우수한 성능을 보여 다양한 멀티미디어 분야에서 사용되고 있는 기술이다[1].

MPEG-1, H.261, MPEG-2, H.263 및 MPEG-4와 같은 기존의 비디오 코딩 표준안은 잉여 데이터(Residual Data)를 주파수 공간으로 변환하기 위해 8x8 DCT (Discrete Cosine Transform)를 사용하지만, H.264/AVC 표준은 잉여 데이터 변환을 위해 4x4 정수 변환을 사용한다[2]. 부동 소수점 연산을 사용하지 않는 이러한 정수 변환은 8x8 DCT와 거의 유사한 결과를 성취할 수 있도록 고안되었다. 특히 정수 덧셈 및 쉬프트 기능만을 사용하여 변환이 가능하게 함으로써 구현의 용이성을 염두해 두고 개발되었다.

H.264/AVC의 변환에서 한 프레임 내에서의 공간적 압축을 수행하는 인트라 예측 모드 값에 대한 변환을 수행하거나, 프레임 간의 시간적 압축을 수행하는 인터 예측 모드에 따른 차분 값에 대한 변환을 수행한다. 인트라 예측 모드는 4x4 예측 모드와 16x16 예측 모드가 존재하는데, 16x16 인트라 예측 모드에 대해서만 추가적으로 DC 값에 대한 하다마드 변환을 수행한다[3,4].

변환이 완료된 값에 정해진 양자화 테이블이나 공식에 따라 나눗셈을 수행하고 그 결과를 정수 값으로 근사화하는 처리 과정을 양자화라고 한다[5]. 일반적으로 양자화 과정을 거치면 보다 작은 정수 값으로 표현할 수 있기 때문에 더 적은 비트 수로 엔트로피 코딩되므로 부호화 효율이 높아진다. 일반적으로 변환과 양자화는 동시에 처리할 수 있도록 설계되는 것이 더 효율적인 것으로 인식되고 있다.

논문 [2]는 고속의 저비용 변환 및 양자화 알고리즘에 대해 연구하였으며 그 성능은 VGA 프레임을 초당 39 프레임 정도 처리할 수 있는 구조를 제안하였다. 그리고 논문 [3]은 인트라 16x16 모드만을 지원하는 하드웨어 아키텍처를 제안하고 이를 FPGA로 구현하였으나 일반적인 상황에 대한 변환 및 양자화에 대해서는 다루지 않았다. 아울러 논문 [4]는 독특한 가산기를 갖는 다중 변환 아키텍처를 제안하여 4x4 정수변환과 하다마드 변환의 결과 값을 동시에 도출하는 방안을 제안하였지만 구체

적인 아키텍처는 제시되지 않았다.

H.264/AVC 인코더에 대한 간략한 블록도는 그림 1에 나타나 있다[2]. 본 논문에서는 H.264/AVC 인코더에서 중요한 역할을 수행하는 변환 및 양자화에 대해 파이프라인 방식을 사용한 고속화 구현 기법에 대해 연구한다. 변환은 모든 모드에 적용되는 4x4 정수 변환과 16x16 인트라 예측 모드에 대해서만 추가적으로 적용되는 하다마드 변환에 대한 아키텍처를 제안하고 HDL 언어를 사용하여 설계한다. 양자화는 16x16 인트라 예측 모드에서 추가적으로 필요한 DC 4x4 Luma 및 2x2 Chroma 양자화를 수행하는 블록과 그 이외의 모든 모드 및 16x16 인트라 예측 모드의 AC 값들에 대한 양자화를 수행하는 블록으로 구성된다. 그리고 양자화 이후에는 지그재그스캔을 한 값들에 대한 엔트로피 코딩을 수행하여 영상 데이터를 전송한다. H.264/AVC에서 엔트로피 코딩은 CABAC과 CAVLC 중의 하나를 적용하게 된다.

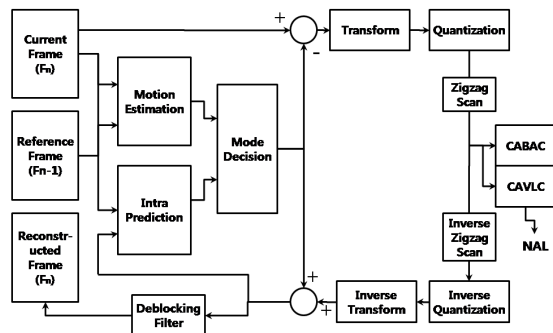


그림 1. H.264/AVC 인코더 블록도
Fig. 1 H.264/AVC Encoder Block Diagram

II. H.264/AVC 변환 및 양자화 알고리즘의 개요

H.264/AVC의 기본적인 변환 및 양자화 과정의 블록도는 그림 2에 나타나 있다[2,7]. 이는 일반적으로 이전에 발표된 표준안과 유사하지만, 16x16 인트라 예측 모드를 사용하는 변환에서는 추가적으로 하다마드 변환을 DC 값들만을 모아서 수행하는 과정이 필요하다. 물론 변환과 양자화 과정이 모두 종료되면 지그재그스캔

하여 엔트로피 코딩 과정이 이어진다. H.264/AVC는 압축률을 높이기 위해서 인트라 모드나 인터 모드 모두가 예측을 수행하며 현재 매크로 블록 값과 재구성된 매크로 블록 값의 차에 대한 변환을 수행한다. 즉, 순방향 DCT 변환으로 들어오는 입력은 현재의 이미지 값에서 예측 값을 빼서 얻어지는 4x4 블록 단위의 잉여 데이터 값들이다. 이러한 변환과 양자화 알고리즘은 매크로 블록 단위로 다수의 블록에 적용되게 되며 결과 값을 지그재그 스캔 방식으로 엔트로피 코딩 블록에 제공하거나 움직임 예측과 보상을 위해 역양자화 및 역 DCT 블록에 전달한다. 그림 2에서 회색 영역은 본 논문에서 다루고자 하는 내용이다.

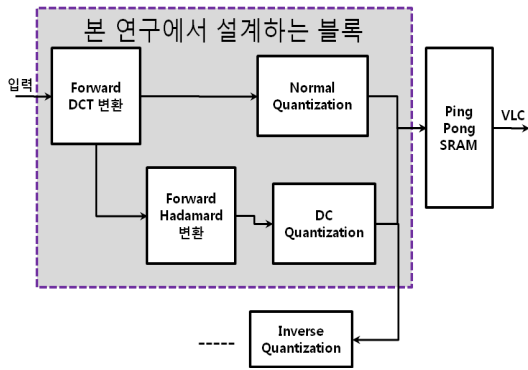


그림 2. H.264/AVC용 변환 및 양자화 블록도
Fig. 2 Block Diagram of Transform and Quantization for H.264/AVC

인코더에서 변환 후 양자화된 계수의 4x4 각 블록은 정의된 지그재그 순서로 16개의 DC를 포함한 AC 계수를 전송하지만, 16x16 인트라 예측 모드로 부호화된 매크로 블록에서는 우선적으로 4x4 Luma DC 계수와 2x2 Chroma DC 계수를 AC 계수에 앞서 변환 후 전송하도록 규정되어 있다[2]. 단, DC 계수 값들만 모든 블록에 대해서는 추가적인 하다마드 변환을 수행해야 한다. 그 이후에 일반적으로 4x4로 양자화된 계수에서 DC 값을 제외한 15개의 AC 계수들을 지그재그 순서로 스캔하여 엔트로피 모듈에 전달한다. 그림 3은 한 매크로 블록 내에서 잉여 값 블록들의 변환을 위한 스캔 순서를 나타내고 있다. 각 4x4 블록에 대한 스캔 순서는 여기서는 생략하기로 하자.

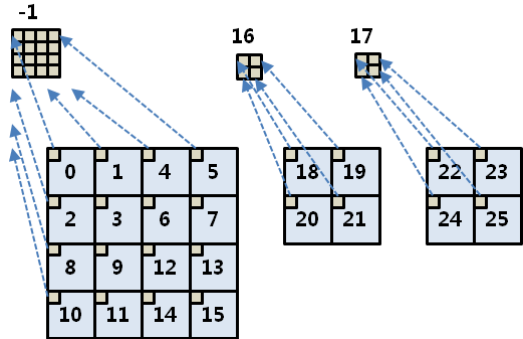


그림 3. 매크로 블록 내에서 잔여 블록들의 스캔 순서
Fig. 3 Scanning Order of Residual Blocks within a Macroblock

H.264/AVC는 변환은 4x4 순방향 정수 DCT 변환과 인트라 16x16 모드에서만 사용되는 Luma 4x4 DC 값에 대한 하다마드 변환 및 Chroma 2x2 DC 값에 대한 하다마드 변환 등 모두 3가지 변환을 필요로 한다[2][4].

입력이 4x4 행렬인 X이며 변환 계수 행렬이 C_f일 때, DCT 변환된 결과 W는 아래와 같다.

$$W = C_f X C_f^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} [X] \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix}$$

그리고 Luma 4x4 하다마드 변환식은 아래와 같이 표현된다. 여기서 W_D는 4x4 입력이며, 변환 계수 행렬은 A_f이며, 변환된 결과는 Y_D가 된다.

$$Y_D = A_f W_D A_f^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} [W_D] \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

아울러 Chroma 2x2 하다마드 변환식은 아래와 같다. 여기서 W_D는 2x2 입력이며, 변환 계수 행렬은 A_f이며, 변환된 결과는 W_{QD}가 된다.

$$W_{QD} = A_f W A_f^T = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} [W_D] \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

H.264/AVC에서의 양자화 과정은 나눗셈과 부동소수점 연산을 사용하지 않고 정수 연산을 사용하여 구현이

가능하도록 제안되었다[8]. 양자화는 DCT 변환이 완료된 이후에 적용되며, 예외적으로 16x16 인트라 예측 모드에 대해서만 추가적인 하다마드 변환 이후에 적용된다.

양자화를 위해서는 양자화 파라미터인 QP 값을 참조하여야 하는데, 0부터 51까지의 값을 가질 수 있다. 양자화 구간의 폭인 QStep 값은 QP 값과 일대일 대응되며 QP가 6 증가할 때마다 QStep은 두 배가 된다. 양자화 과정에서 계산 결과의 쉬프트 양을 결정하는 qbits 값과 f 값은 아래와 같이 표현된다. qbits의 범위는 QP가 0일 때 15가 되며, QP 최대 값인 51일 때 23이 된다. 아울러 f 값은 아래와 같이 인트라 블록일 경우와 인터 블록일 경우에 대해 다른 값을 사용하여 구현된다.

$$qbits = 15 + \text{floor}(QP/6)$$

여기서

$$f = 2^{qbits}/3 \text{ [인트라블록]}$$

$$f = 2^{qbits}/6 \text{ [인터블록]}$$

또한 양자화시에는 MF 값을 사용하는데, 이는 (QP % 6)의 결과에 대응하여 서로 다른 값을 가진다. 즉, MF는 QP 값에 따라 6개의 테이블 값들 중에서 하나를 사용하게 된다.

아래 식에서 Z_{ij} 는 양자화된 결과 값이며, W_{ij} 는 변환 이후의 양자화를 위한 입력 값에 대응된다. MF, f 및 qbits 값에 대해서는 위에서 설명하였다. 그리고 양자화된 결과를 얻기 위해서는 먼저 무비호로 값을 변환하여 양자화를 수행한 후 sign()함수의 결과가 1이면 양자화 결과 값을 음수로 변환하고 그렇지 않으면 양자화 변환된 값을 그대로 사용하면 된다.

$$|Z_{ij}| = (|W_{ij}| \cdot MF + f) \gg qbits$$

$$\text{sign}(Z_{ij}) = \text{sign}(W_{ij})$$

마지막으로 양자화와 관련하여 하다마드 변환이 완료된 DC 값에 대한 양자화는 아래와 같다. 모든 DC 값에 대해 동일한 식을 적용한다. 다만 일반적인 양자화와 구별되는 것은 f 값에 2를 곱한 값을 사용하는 것과 우측 쉬프트 양을 qbits가 아닌 qbits+1 값을 사용한다는 것이다.

$$|Z_{D(i,j)}| = (|Y_{D(i,j)}| \cdot MF_{(0,0)} + 2 \cdot f) \gg (qbits + 1)$$

$$\text{sign}(Z_{D(i,j)}) = \text{sign}(Y_{D(i,j)})$$

III. 제안하는 하드웨어 구조

본 논문의 핵심의 H.264/AVC에서의 변환 코딩과 양자화 하드웨어의 코어를 설계하는 것이다. 먼저 DCT 설계에 대해서 설명하겠다. 통상적으로 2차원 DCT를 2개의 1차원 DCT로 구분하여 설계하는 것이 하드웨어적인 측면에서 유리하다. 따라서 본 논문에서도 2개의 1차원 DCT 기법을 적용하였다. 전에 언급한 바와 같이 H.264/AVC는 DCT를 정수 가산기와 쉬프트만을 사용하여 구현이 가능하다. 본 논문에서 사용한 1st DCT 구조는 그림 4와 같다. 인트라 및 인터 매크로 블록에 대한 4x4 단위의 잉여(잔여) 데이터에 대한 1차원 DCT 변환을 수행한다. 본 논문에서는 외부에서 입력되는 데이터가 매 클럭당 1개의 픽셀 데이터가 유입될 수 있는 구조로 설계하였다. 아울러 그림 4에는 4x4 블록에서 데이터를 읽는 순서에 표시하였다. 행 단위로 데이터가 1차원 DCT 블록에 입력될 수 있고, 먼저 MBDCT_START 신호가 활성화되면 언제든지 DCTEn 신호가 활성화될 때 유효한 DCT 연산을 위한 데이터가 입력된다. 매 4 클럭 사이클마다 ACC0~ACC3 레지스터는 유효한 DCT 값을 가지게 되면 유효한 데이터는 동시에 전치메모리(Transpose Memory)로 전달된다.

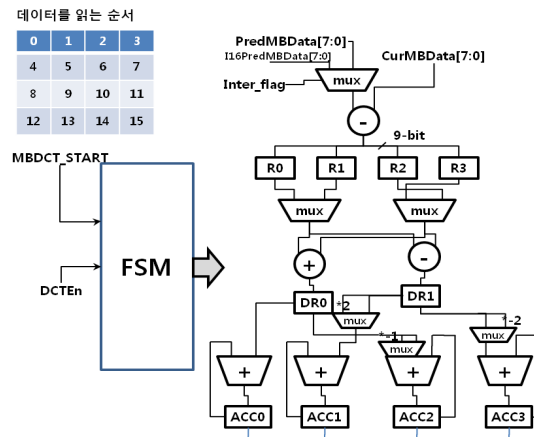


그림 4. H.264/AVC용 첫번째 DCT의 아키텍처
Fig. 4 Architecture of 1st DCT for H.264/AVC

다음은 두 번째 DCT 연산을 수행하는 블록의 아키텍처를 그림 5에서 보여주고 있다. 전치 메모리에서 항상 4x4 DCT 연산을 위해 클럭당 12비트 4개 데이터가 연속

적으로 4회에 걸쳐 입력된다. 따라서 파이프라인의 첫 번째 단계에서는 덧셈 및 뺄셈에 필요한 모든 연산을 수행하여 DR0~DR4 레지스터에 저장하고, 다음 단계에서는 최종적인 DCT 결과를 얻기 위해 덧셈 및 뺄셈 연산을 수행하여 이 결과를 DCTRes0~DCTRes3 레지스터에 저장하여 두 번째 전치 메모리에 저장한다.

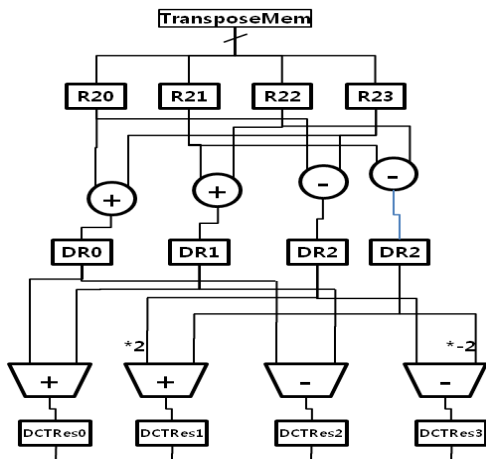


그림 5. H.264/AVC용 두 번째 DCT의 아키텍처
Fig. 5 Architecture of 2nd DCT for H.264/AVC

하다마드 변환은 인트라 16x16 예측 모드에서만 적용되는 변환으로 위에서 설명한 DCT 연산 하드웨어와 유사한 아키텍처를 가지고 계수만 다른 값을 가지므로 특별한 설명은 하지 않을 것이다.

다음은 H.264/AVC에서 2차원 DCT 연산을 완료한 이후에 이루어지는 양자화 기법에 대해 설명하기로 하자. 16x16 인트라 예측을 사용할 경우에 한정하여 하다마드 변환을 수행하고 변환을 완료한 이후에는 5단 파이프라인에 의해 양자화를 수행한다. DCT 완료된 데이터의 값이 음수이면 양수로 보정하고 부호 비트를 파이프라인을 통해 전송한다. 다음은 $|W'_{ij}| \cdot MF$ 곱셈 연산을 수행한다. MF 값은 사전에 ROM 테이블에 저장되며, $QP\%6$ 연산 결과에 따라 다른 MF 값을 사용하게 된다. 곱셈이 완료되면 곱셈 결과에 f 값을 더하는 덧셈 연산을 수행한다. 이전에 언급한 바와 같이 f 값은 인트라 모드에 대해서는 $2^{qbits}/3$ 이고, 인터 모드에 대해서는 $2^{qbits}/6$ 의 값을 가진다. 따라서 qbits의 값이 결정되면 나눗셈기를 사용하여 f 값을 구하면 많은 하드웨어를 차지하므로 간단

히 하드웨어로 구현이 가능하도록 하는 방안을 제안하였다. 이는 다시 설명할 것이다. 덧셈이 이후에는 $(|W'_{ij}| \cdot MF) + f \gg qbits$ 와 관련된 쉬프트 연산을 수행한다. 마지막으로 양자화를 시작할 때 입력된 데이터와 동일한 부호 값을 갖도록 보정하면 최종적으로 DC 양자화된 결과 값을 얻을 수 있다. 또한 16x16 인트라 예측 모드를 제외한 양자화는 그림 6의 우측에 있는 양자화를 수행한다.

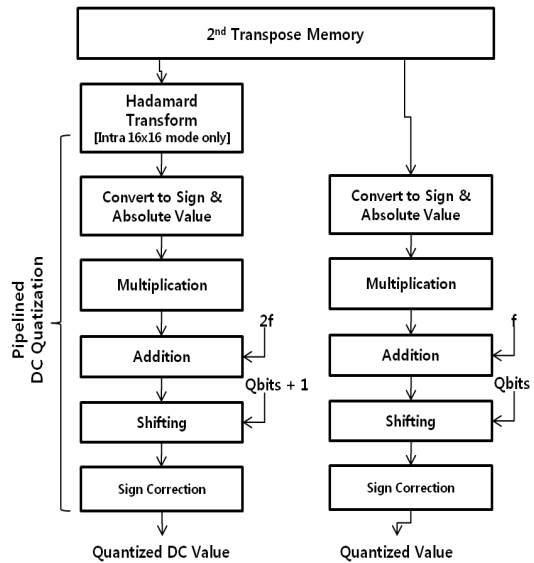


그림 6. H.264/AVC용 양자화 흐름도
Fig. 6 Flow Chart of Quantization for H.264/AVC

DC 양자화 과정과 동일한 방식으로 계산되지만 덧셈에 적용되는 f 값과 쉬프트에 적용되는 qbits 값에 대해서만 차이가 존재한다. 양자화를 병렬로 수행하도록 아키텍처를 구성한 것은 지연없이 연속적인 매크로 블록의 처리가 가능하도록 하기 위해서 제안하였다. 다음은 f 값에 대해서만 좀 더 자세히 설명하면 다음과 같다. f 값을 구하기 위해 나눗셈기를 사용하는 것은 많은 하드웨어를 소요하기 때문에 간략하게 구현하는 방법을 적용하였다. qbits가 가질 수 있는 값의 범위는 15~23까지이다. 결과적으로 $f = 2^{qbits}/3$ [인트라블록], $f = 2^{qbits}/6$ [인터블록]이므로 f 값은 qbits 값에 대응하여 존재한다는 것을 알 수 있다. qbits=19일 때, $f = 2^{19}/3 = 2^{19}/3 = 174763$ 이

므로 이 값을 기준으로 qbits가 19보다 작으면 19와의 차만큼 왼쪽 쉬프트하고, 19보다 크면 우측 쉬프트하면 원하는 f 값을 얻을 수 있다.

IV. 시뮬레이션

본 논문에서는 2가지 경우로 나누어 시뮬레이션을 수행한 과정을 실었다. 첫 번째는 인트라 16x16 예측 모드가 아닌 경우, 즉 일반적인 경우의 시뮬레이션이다. 이 경우에는 4x4 하다마드 변환 및 이에 대한 DC 양자화를 수행하지 않는 경우이다. MBDctStart 신호 이후 DCTEn 신호가 활성화되면 DCT를 위한 유효한 데이터가 입력되는 것을 의미한다. 클럭당 1개의 픽셀이 입력되는 상황에서 4클럭 1회씩 1st DCT 결과가 출력되며, 모두 16개(4x4)의 1-D DCT 완료된 데이터가 수신되면 다시 전치된 데이터에 대해 2nd DCT가 수행된다. 두 번째 DCT 변환된 최종 결과는 양자화 모듈에서 양자화가 수행되면 QuantOutEn 신호가 활성화됨과 동시에 외부 메모리 쓰기 주소인 QWrAddr 신호와 데이터인 QuantDout 신호에 유효한 값이 실리게 된다. 이에 대한 것은 그림 7에 나타나 있다.

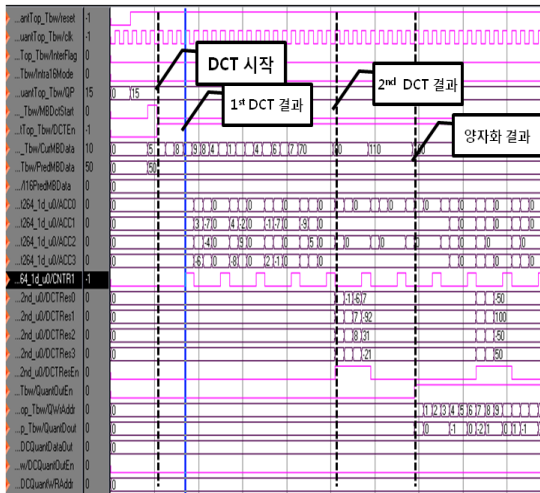


그림 7. 일반적인 DCT 및 양자화 과정 시뮬레이션
Fig. 7 Simulation of conventional DCT and Quantization process

그림 8은 인트라 16x16 예측 모드인 경우 하다마드 변환 및 DC 양자화에 대한 시뮬레이션 파형이다. 그림은 DCT 변환 이후의 하다마드 변환 및 DC 양자화를 수행하는 부분에 대한 설명을 포함하고 있다. 그림에서 보면 알 수 있듯이 DC 양자화의 최종 결과가 AC 계수에 대한 양자화보다 먼저 종료됨을 알 수 있다.

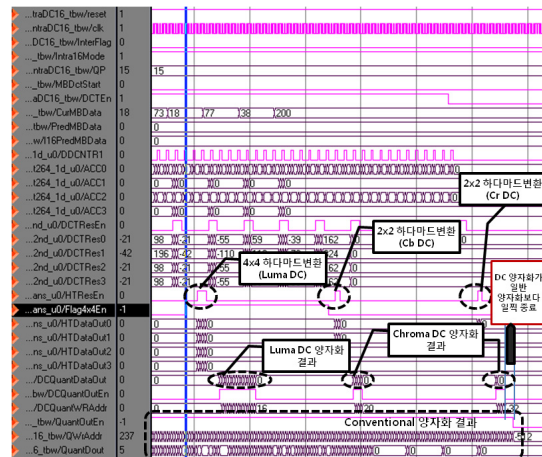


그림 8. 하다마드 변환 및 DC 양자화를 포함한 시뮬레이션
Fig. 8 Simulation including Hadamard transform and DC quantization

이는 DCT 및 양자화에 대한 하드웨어 구현에 있어 파이프라인 구현을 위한 필수 조건이다. 본 논문에서는 하다마드 변환 수행 여부가 설계한 IP의 전체적인 파이프라인 수행에 있어서 문제가 되지 않는 아키텍처임을 알 수 있다. 만약에 하다마드 변환이 존재할 경우 이의 양자화 결과가 기존의(Conventional) 양자화 결과보다 늦게 종료된다면 VLC(Variable Length Coding) 코딩에서 DC 양자화 값을 먼저 처리해야 하는 문제로 인해 매크로 블록간의 연속적인 DCT 변환 및 양자화를 불가능하게 한다. 본 논문에서는 이러한 문제점을 해결할 수 있는 아키텍처를 사용하여 설계하였음을 보여주고 있다.

V. 결과 고찰

본 논문에서 제안한 아키텍처의 타이밍을 분석하면 그림 9와 같이 나타낼 수 있다.

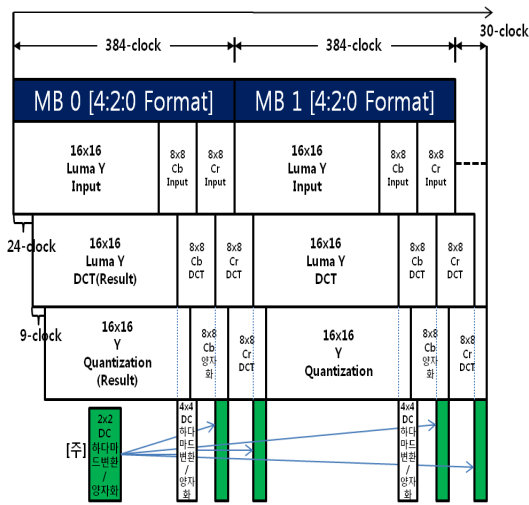


그림 9. 파이프라인 방식의 DCT 및 양자화 타이밍 전략
 Fig. 9 Timing Strategy of Pipelined DCT and Quantization

4:4:2 영상 포맷에서 1개의 매크로 블록은 모두 384개의 픽셀로 구성되어 있으면 처음 256 픽셀은 Y 성분이며, 다음 64개 픽셀은 Cb 성분 그리고 마지막 64 픽셀은 Cr 성분이다. 각 매크로 블록에 대한 DCT 및 양자화 과정을 첫 번째 매크로 블록 수행 과정에서의 대기 지연시간을 제외하고는 연속적으로 결과 값이 출력된다. 따라서 파이프라인이 안정화된 상태에서는 매 클럭 당 1개의 픽셀이 입력되고 동시에 1개의 양자화 값이 출력되도록 하였다. 그림에서 보면 알 수 있듯이 DC 값에 대한 하다마드 변환 및 양자화는 일반 DCT 변환 및 양자화용 하드웨어를 사용하는 것이 아니라 병렬로 수행되는 하드웨어를 추가하여 구현되었으며, DCT 양자화 수행보다 하다마드 양자화 수행이 일찍 완료되기 때문에 후단의 VLC 변환이 즉각적으로 수행될 수 있는 구조이다. 즉, 하다마드 변환의 유무와 상관없이 DCT 변환은 파이프라인 방식으로 33클럭의 대기 지연시간 이후에는 지속적으로 결과 값을 출력할 수 있다.

본 논문에서 설계한 DCT, 하다마드 변환 및 양자화는 Verilog HDL로 코딩하고, Xilinx 7.1i ISE툴을 사용하여 합성 및 검증하였다. 합성 결과 SPARTAN3S-1000 디바이스에서 106MHz에서 동작하는 것을 확인하였다. 1595개의 슬라이스와 1941개의 슬라이스 플립플롭 리소스

를 사용하여 구현되었다. 353x288 CIF 영상 프레임은 최대 697프레임을 처리할 수 있으며, 1920X1088 HD 영상 프레임의 경우 최대 33프레임을 처리할 수 있다. 따라서 본 논문에서 설계된 IP는 30프레임 이상의 1920X1080 HD 영상 압축에도 응용이 가능하다.

VI. 결론

본 논문에서는 H.264/AVC에 중요한 역할을 하는 DCT 변환, 하다마드 변환 및 양자화에 대한 하드웨어적 구현에 대해 연구하였다. 특히 파이프라인 기법을 적용하여 33클럭의 대기 지연시간 이후에는 매 클럭 당 1개의 양자화된 결과를 출력할 수 있는 아키텍처를 제안하였다. 하다마드 변환 및 양자화는 일반적인 매크로 블록내에서 DCT 변환과 양자화 과정과 병행하여 매크로 블록의 수행 사이클 이내에서 수행되기 때문에 추가적인 시간을 필요로 하지 않는다. H.264/AVC 코덱은 지상파 및 위성 DMB분야, 감사용 카메라, 자동차용 블랙박스 등 다양한 분야에 응용되고 있으며, 본 연구에서 제안한 하드웨어는 1920x1080 HD 급의 H.264/AVC용 DCT, 하다마드 변환 및 양자화를 위한 IP로 응용할 수 있을 것으로 사료된다.

향후 연구는 본 연구에서 확보한 IP와 역 DCT, 역하다마드 변환 및 역양자화 등을 포함시키고, 움직임 예측과 보상을 합체한 H.264/AVC 인코더 구현에 활용될 수 있을 것으로 기대된다.

참고문헌

- [1] ITU-T Rec. H.264, "Advanced Video Coding for Generic Audio Visual Services", 2005
- [2] Logashanmugam, Ramachandran, "An Efficient Hardware Architecture for H.264 Transform and Quantization Algorithms", IICSNS, Vol.8, No.6, pp167-173, June 2008
- [3] Hassen, Imen, Nouri, Ahmed and Patrice, "FPGA Design of an Intra 16x16 Module for H.264/AVC Video Encoder", Circuits and Systems, pp18-29, Jan. 2010

- [4] Chen, Huang and Lai, "Analysis and Architecture Design of Multi-Transform Architecture for H.264/AVC Intra Fram Coder", IEEE ICME, pp145-148, 2008
- [5] 정제창 역, *H.264/AVC 비디오 압축 표준*, 홍릉과학출판사, 2004.
- [6] Wang, Huang, Fang and Chen, "Parallel 4x4 2D Transform and Inverse Transform Architecture for MPEG-4 AVC/H.264", Proceedings of the 2003 IEEE International Symposium on Circuits and Systems, pp800-803, 2003
- [7] Iain Richardson, "Video Compression Design, Analysis, Consulting and Research", VCodex, pp1-13, Apr. 2009
- [8] 호요성, 김승환, *H.264/AVC 표준의 소스 코드 분석*, 두양사, 2006

저자소개



손승일(Seung-II Sonh)

1989년 연세대학교
전자공학과(학사)
1991년 연세대학교 대학원
전자공학과(석사)

1998년 연세대학교 대학원 전자공학과(박사)
1998~2002년 호남대학교 컴퓨터공학과 조교수
2008~2009년 미국 미시간공과대학 방문교수
2002년~현재 한신대학교 정보통신학과 교수
※관심분야: ATM 통신 및 보안, ASIC 설계