

고성능 H.264/AVC 복호기를 위한 병렬 역양자화 및 역변환 구조 설계

정홍균, 류광기
한밭대학교 정보통신전문대학원
e-mail:{hkjung, kkryoo}@hanbat.ac.kr

Design of Parallel Inverse Quantization and Inverse Transform Architecture for High Performance H.264/AVC Decoder

Hongkyun Jung and Kwangki Ryoo
Graduate School of Information and Communication
Hanbat National University

요 약

본 논문에서는 H.264/AVC 복호기의 성능을 향상시키기 위해 병렬 역양자화 구조와 역변환 구조를 제안한다. 제안하는 역양자화 구조는 공통 연산기를 사용하여 계산 복잡도를 감소시키고, 4개의 공통 연산기를 사용하여 역양자화 수행 사이클 수를 1 사이클로 감소시킨다. 제안하는 역변환 구조는 4개의 변환 연산기를 사용하여 역변환 연산을 수행하는데 2 사이클이 소요된다. 또한 제안하는 구조는 역양자화 연산과 수평 역변환 연산을 동시에 수행하는 병렬 구조를 채택하여 역양자화 및 역변환 수행 사이클 수를 2 사이클로 감소시킨다. 제안하는 구조를 Magnachip 0.18um CMOS 공정 라이브러리를 이용하여 합성한 결과 1.5MHz의 동작 주파수에서 게이트 수는 14,173이고, 표준 참조 소프트웨어 JM 9.4에서 추출한 데이터를 이용하여 성능을 측정한 결과 제안하는 구조의 수행 사이클 수가 기존 구조 대비 38.74% 향상되었다.

1. 서 론

H.264/AVC는 ITU-T의 비디오 코딩 전문가 그룹(VCEG)과 ISO/IEC의 동영상 전문가 그룹(MPEG)이 함께 구성한 JVT(Joint Video Team)에 의해 제안된 동영상 압축 기술로 기존 MPEG-4 Part 2 표준보다 50%이상의 높은 압축률을 제공한다. H.264/AVC는 비디오 압축 성능을 높이기 위해 기존의 비디오 부호화 표준에 비해 더 작은 크기인 4x4 블록 단위의 움직임 보상과 1/4 화소 단위의 움직임 예측, 향상된 엔트로피 부호화 방식, 그리고 새로운 비용함수를 이용한 모드결정 방법 등을 사용한다[1-2].

H.264/AVC의 정수 기반 변환은 정수단위 연산을 사용하기 때문에 부호화기와 복호화기 사이의 변환 계수 불일치 문제를 해결하였고, 변환의 곱셈연산이 양자화에 통합되어 덧셈과 쉬프트 연산만으로 구현이 가능하다. [3]의 역양자화 및 역변환 구조는 DC 계수와 AC 계수에 따라 수행 순서가 다르기 때문에

제어 구조가 복잡하고, 서로 다른 역양자화 수식으로 인한 역양자화기의 계산 복잡도가 높다. 또한 역변환을 수행하기 위해 하나의 변환 연산기를 사용하기 때문에 역변환을 수행하는데 8 사이클이 소요된다. [4]의 역양자화 구조는 공통 연산기를 사용하여 역양자화기의 계산 복잡도를 감소시켰지만 1개의 수평 변환 연산기를 사용하기 때문에 수평 역변환 연산을 수행하는데 4 사이클이 소요된다.

따라서 본 논문에서는 4개의 공통 연산기를 사용하는 역양자화 구조를 제안하여 역양자화기의 계산 복잡도를 감소시키고 역양자화 수행 사이클 수를 1 사이클로 감소시킨다. 또한 역변환 구조는 4개의 변환 연산기를 사용하여 역변환 수행 사이클 수를 2 사이클로 감소시킨다. 제안하는 역변환 및 역양자화 구조의 수행순서는 DC 계수와 AC 계수에 상관없이 동일한 수행 순서를 갖고 역양자화 과정과 수평 역변환 과정을 동시에 수행함으로써 4x4 블록을 처리하는데 1 사이클이 소요된다.

본 논문의 구성은 다음과 같다. 2장에서는 역변환

및 역양자화 알고리즘에 대해 기술하며, 3장에서는 제안하는 역변환과 역양자화 구조에 대해 기술하고, 4장에서는 제안하는 구조와 기존 구조의 사이클 수를 비교하고 검증결과를 기술한다. 마지막으로 5장에서는 본 연구의 결론을 도출한다.

2. 역변환 및 역양자화 알고리즘

H.264/AVC는 변환과 양자화의 계산 복잡도를 감소하기 위해 정수 변환과 양자화가 통합된 알고리즘을 사용하며 이에 따른 정수 역변환은 식(1)과 같다. 식(1)에서 X'는 정수 역변환된 블록 행렬, Y'는 역양자화와 정수 변환된 블록 행렬, Ci는 역방향 정수 변환 행렬을 나타내며 CiT는 Ci 행렬의 전치행렬을 의미한다[4-5].

$$X' = C_i^T Y' C_i = \begin{bmatrix} 1 & 1 & 1 & \frac{1}{2} \\ 1 & \frac{1}{2} & -1 & -1 \\ 1 - \frac{1}{2} & -1 & 1 & \\ 1 & -1 & 1 & -\frac{1}{2} \end{bmatrix} [Y'] \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & \frac{1}{2} & -\frac{1}{2} & -1 \\ 1 & -1 & -1 & 1 \\ \frac{1}{2} & -1 & 1 & \frac{1}{2} \end{bmatrix} \quad (1)$$

정수 변환 행렬들의 정수 근사화를 위해 각각 행렬 위치의 곱셈 인자와 크기조정 인자는 역양자화에 통합된다. 역양자화 구현시 연산량을 감소시키고자 변경된 역양자화식은 식(2)과 같고 $V_{(i,j)}$ 는 스케일링 계수를 의미한다.

$$W_{D(i,j)} = Z_{QD(i,j)} \cdot V_{(i,j)} \cdot 2^{\text{floor}(QP/6)} \quad (2)$$

Intra 16x16 모드로 코딩된 4x4 휘도 DC 계수의 역양자화는 식(3)와 (4)에 의해 수행된다. QP (Quantization Parameter)값이 12보다 크거나 같을 때는 식(3)와 같이 역양자화를 수행하며, QP가 12보다 작은 경우에는 식(4)와 같이 역양자화를 수행한다.

$$W_{D(i,j)} = Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)-2} \quad (3)$$

$$W_{D(i,j)} = Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{1-\text{floor}(QP/6)} \gg 2 - \text{floor}(QP/6) \quad (4)$$

2x2 색차 DC 계수의 역양자화는 식(5)와 (6)에 의해 수행된다. QP값이 6보다 크거나 같을 때는 식(5)과 같이 역양자화를 수행하며, QP값이 6보다 작은 경우에는 식(6)과 같이 역양자화를 수행한다.

$$W_{D(i,j)} = Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)-1} \quad (5)$$

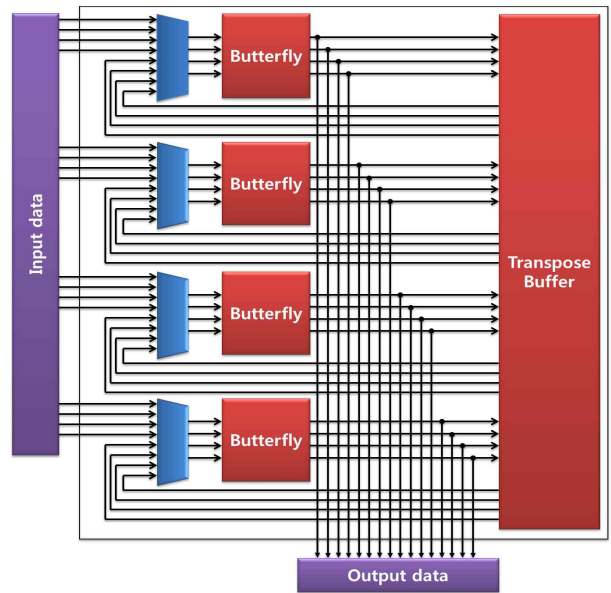
$$W_{D(i,j)} = Z_{QD(i,j)} \cdot V_{(0,0)} \gg 1 \quad (6)$$

3. 제안하는 역양자화 및 역변환 구조

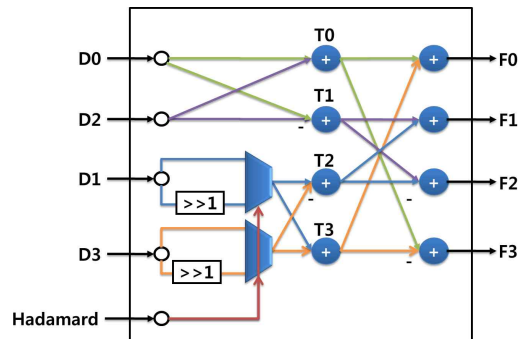
제안하는 역변환 및 역양자화 구조는 4개의 공통 연산기를 갖는 병렬 역양자화 구조, 4개의 변환 연산기를 갖는 역변환 구조를 채택하였고 역양자화 과정과 수평 역변환 과정을 동시에 수행한다.

3.1 역변환 구조

역변환 구조는 그림 1과 같이 4개의 변환 연산기(butterfly)와 트랜스포즈 버퍼(Transpose Buffer)로 구성된다. 역변환 구조는 4개의 변환 연산기를 사용하기 때문에 수평 역변환 연산과 수직 역변환 연산을 수행하는데 각각 1사이클이 소요된다. 변환 연산기는 그림 2와 같이 하나의 연산기를 사용하여 정수 변환 연산 또는 하다마드 변환 연산을 수행한다. 정수 변환 연산 또는 하다마드 변환 연산 수행은 Hadamard 신호에 의해 결정된다.



[그림 1] 제안하는 역변환 구조



[그림 2] 변환 연산기 구조

3.2 역양자화 구조

제안하는 역양자화 구조는 계산 복잡도를 감소시키기 위해 공통 연산기를 사용하고, 역양자화 구조의 수행 사이클 수를 감소시키기 위해 4개의 공통 연산기를 사용한 병렬 구조를 채택하였다. 공통 연산기를 사용하기 위해 서로 다른 5개의 양자화 수식인 (2)~(6)을 곱셈과 고정길이의 오른쪽 쉬프트 연산을 갖는 연산 수식인 (7)~(11)로 변경하였다. 식(7)은 4x4 AC 계수의 역양자화식이고, 식(8)과 (9)는 4x4 휘도 DC 계수의 역양자화 식이며, 식(10)과 (11)은 2x2 색차 DC 계수의 역양자화식이다[3].

$$\begin{aligned}
 W_{D(i,j)} &= Z_{QD(i,j)} \cdot V_{(i,j)} \cdot 2^{\text{floor}(QP/6)} \\
 &= Z_{QD(i,j)} \cdot V_{(i,j)} \cdot 2^{\text{floor}(QP/6)} \cdot 2^2 \gg 2 \\
 &= Z_{QD(i,j)} \cdot V_{(i,j)} \cdot 2^{\text{floor}(QP/6)+2} \gg 2
 \end{aligned}
 \tag{7}$$

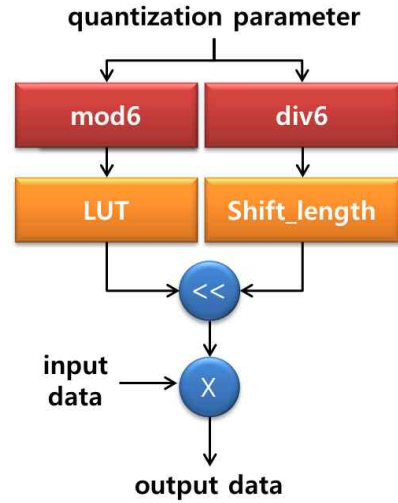
$$\begin{aligned}
 W_{D(i,j)} &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)-2} \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)-2} \cdot 2^2 \gg 2 \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)} \gg 2
 \end{aligned}
 \tag{8}$$

$$\begin{aligned}
 W_{D(i,j)} &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{1-\text{floor}(QP/6)} \gg 2 - \text{floor}(QP/6) \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{1-\text{floor}(QP/6)} \cdot 2^{\text{floor}(QP/6)} \gg 2 \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2 \gg 2 \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2 \gg 2
 \end{aligned}
 \tag{9}$$

$$\begin{aligned}
 W_{D(i,j)} &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)-1} \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)-1} \cdot 2^2 \gg 2 \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2^{\text{floor}(QP/6)+1} \gg 2
 \end{aligned}
 \tag{10}$$

$$\begin{aligned}
 W_{D(i,j)} &= Z_{QD(i,j)} \cdot V_{(0,0)} \gg 1 \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2 \gg (1+1) \\
 &= Z_{QD(i,j)} \cdot V_{(0,0)} \cdot 2 \gg 2
 \end{aligned}
 \tag{11}$$

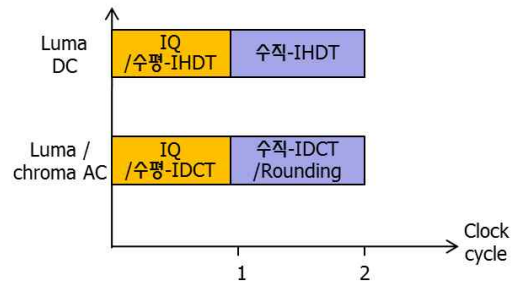
변경된 수식은 기존 수식의 가변길이의 오른쪽 쉬프트 연산을 고정 길이의 오른쪽 쉬프트 연산으로 변경되었고, 나눗셈 연산과 곱셈 연산이 분리되었다. 또한 고정 길이의 나눗셈 연산을 라운딩 과정에 통합하여 역양자화기는 곱셈 연산기만을 사용하여 역양자화 과정을 수행할 수 있다. 그림 3은 역양자화기 구조를 나타낸다. 역양자화기는 QP(Quantization parameter)를 6으로 나눈 나머지와 몫을 계산하는 mod6/div6 블록, 스케일링 계수에 대한 LUT 블록, 몫에 대한 shift 크기를 결정하는 shift length 블록과 공통 연산기로 구성된다. mod6/div6 블록은 나눗셈 연산을 사용하지 않고, 52개 QP에 대한 나머지와 몫을 LUT로 저장하고, 공통 연산기는 DC/AC 계수와 쉬프트 연산된 스케일링 계수를 곱한다.



[그림 3] 역양자화기 구조

3.3 역변환 및 역양자화 수행 순서

제안하는 역변환 및 역양자화 구조는 그림 4에서와 같이 DC 계수와 AC 계수에 상관없이 동일한 수행 순서를 갖고 수행 단계는 역양자화/역변환 단계와 역변환/라운딩 단계로 구성된다. 역양자화/역변환 단계는 역양자화 과정과 수평 역변환 과정을 동시에 수행하고, 역변환/라운딩 단계는 수직 역변환 과정과 라운딩 과정을 동시에 수행하기 때문에 4x4 블록을 처리하는데 2사이클이 소요된다. 따라서 제안하는 구조는 4x4 블록을 처리하는 수행 단계와 수행 순서에 대한 제어 로직이 단순화된다.



[그림 4] 제안하는 구조의 수행 순서

4. 실험 및 고찰

제안하는 역변환 및 역양자화 구조를 Magnachip 0.18um 표준 셀 라이브러리를 사용하여 합성한 결과 역변환 구조, 역양자화 구조, 제어로직의 게이트 수는 각각 3,547, 6,457, 4,169이고 총 게이트 수는 14,173이다. 표 1은 제안하는 구조와 기존 구조들의 성능을 비교한 결과이다. 제안하는 구조의 제어로직

의 게이트 수는 [3]과 [4] 대비 각각 58.53%, 49.74% 감소하였다. 제안하는 구조의 총 게이트 수는 [3] 대비 6.24% 증가하였고, [4] 대비 0.39% 감소하였다. 제안하는 구조는 역양자화 연산기와 역변환 연산기를 각각 4개를 사용하고, [3]은 각각 1개를 사용하기 때문에 제안하는 구조의 총 게이트 수가 증가하였다.

[표 1] 제안하는 역변환 및 역양자화 회로의 비교

		[3]	[4]	Proposed
공정(um)		0.18	0.18	0.18
동작 주파수(MHz)		1.5	1.5	1.5
연산기 수	역변환	1	5	4
	역양자화	1	1	4
게이트 수	역변환	883	4,483	3,547
	역양자화	2404	1,451	6,457
	제어로직	10,523	8,295	4,169
	총합	13,340	14,229	14,173

제안하는 구조는 H.264/AVC 참조 소프트웨어 JM 9.4[6]에서 추출한 데이터를 이용하여 동작 검증 및 성능을 측정하였다. 표 2는 기존의 구조와 제안하는 구조의 수행 사이클 수를 비교한 결과이다. 제안하는 구조는 하나의 16x6 매크로블록을 처리하는데 평균 77.5 사이클이 소요되고, [3]과 [4]는 각각 248.5 사이클, 126.5 사이클이 소요된다. 따라서 16x16 매크로블록을 처리하는 평균 수행 사이클 수를 기준으로 제안한 구조는 [3]과 [4] 대비 성능이 각각 68.81%, 38.74% 향상되었다.

[표 2] 16x16 매크로블록당 수행 사이클 수

예측 모드	수행 사이클 수		
	[3]	[4]	Proposed
Intra 16x16 모드	253	129	83
그 밖의 모드	244	124	72
평균	248.5	126.5	77.5

5. 결론

본 논문에서는 H.264/AVC 복호기의 성능을 향상시키기 위해 병렬 역양자화 및 역변환 구조를 제안한다. 병렬 역양자화 구조는 공통 연산기를 사용하여 계산 복잡도를 감소시켰고, 4개의 공통 연산기를 사용한 병렬 구조를 채택하여 역양자화 수행 사이클 수를 1사이클로 감소시켰다. 역변환 구조는 4개의 변환 연산기를 사용하여 역변환 수행 사이클 수를 2사이클로 감소시켰다. 또한 역양자화 과정과 수행

역변환 과정을 동시에 수행하기 때문에 4x4 블록을 처리하는데 2 사이클이 소요된다. 제안하는 구조를 Magnachip 0.18um CMOS 공정 라이브러리를 이용하여 합성한 결과 게이트 수는 14,173이고 JM 9.4에서 추출한 데이터를 이용하여 수행 사이클 수를 측정한 결과 제안하는 구조는 기존 구조 대비 38.74% 향상되었다.

감사의 글

본 연구는 한밭대학교의 2011년 교내학술연구비 지원사업과 IDEC의 CAD Tool 지원사업의 연구결과임.

참고문헌

- [1] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthar, "Overview of the H.264/AVC Video Coding Standard," IEEE Trans. Circuit and Systems for Video Technology, Vol. 13, pp. 560-576, July, 2003.
- [2] I. E. Richardson, The H.264 Advanced Video Compression Standard : Second Edition, Wiley, 2010.
- [3] K. Xu and C. S. Choy, "Low-Power Bitstream-Residual Decoder for H.264/AVC Baseline Profile Decoding", EURASIP Journal on Embedded Systems, vol. 2009, pp.1-17, July, 2009.
- [4] 정홍균, 차기중, 박승용, 김진영, 류광기, "H.264/AVC 복호기의 병렬 역변환 구조 및 저면적 역양자화 구조 설계", 한국해양정보통신학회 2011년 추계학술대회, 제15권, 제2호, pp.444-447, 10월, 2011.
- [5] 임성창, 허재호, 한기훈, 이영렬, "1-D 정수 변환과 양자화를 이용한 새로운 수직/수평 화면내 예측 방법", 19회 신호처리합동학술대회 논문집, 제16권, 제1호, pp.1-4, 9월, 2006.
- [6] Joint Video Team(JVT) Reference Software JM 9.4