

논문 2012-49-12-24

실시간 영상 부호화를 위한 H.264/AVC의 비트율 제어 하드웨어 설계

(Hardware Design of Rate Control for H.264/AVC Real-Time Video
Encoding)

김 창 호*, 류 광 기**

(Changho Kim and Kwangki Ryoo)

요 약

본 논문에서는 실시간 동영상 부호화를 위한 효과적인 비트율 제어 방법을 제안하고 하드웨어로 구현한다. 제안하는 비트율 제어는 각 기본유닛마다 R-D 파라미터 갱신에 의해 많은 연산 처리를 필요로 하는 이차원 R-D 모델을 사용하지 않고, 프레임의 평균 복잡도 가중치를 이용한 Qstep 결정 모델을 사용함으로써 연산량을 감소시킨다. 또한 적은 연산량과 빠른 MAD 예측을 위해 부호화된 기본유닛을 기반으로 영상의 공간적 및 시간적 상관관계를 이용하여 MAD를 예측한다. 제안하는 비트율 제어는 프레임 레벨 MAD 예측과 매크로블록 레벨 MAD 예측, GOP 비트 할당, 프레임 비트 할당, 가상버퍼, Qstep 결정 모델, 비트율 제어 파라미터 갱신, QP 결정 모듈을 포함하며 총 8개의 모듈로 구성된다. 비트율 제어 하드웨어는 Verilog-HDL을 이용하여 설계하였으며, Synopsys사의 Design Compiler를 이용하여 UMC 공정 0.18 μm 셀 라이브러리로 합성한 결과, 최대 동작 주파수는 108 MHz이고, 게이트 수는 19.1k이다. 실험 결과로부터 제안한 구조는 기존 구조 보다 게이트 수가 85% 감소하였고, 매크로블록 당 QP를 결정하는데 평균 220 사이클 수가 소요되어 기존 구조보다 64% 단축됨을 확인하였다.

Abstract

In this paper, the hardware design of rate control for real-time video encoded is proposed. In the proposed method, a quadratic rate distortion model with high-computational complexity is not used when quantization parameter values are being decided. Instead, for low-computational complexity, average complexity weight values of frames are used to calculate QP. For high speed and low computational prediction, the MAD is predicted based on the coded basic unit, using spacial and temporal correlation in sequences. The rate control is designed with the hardware for fast QP decision. In the proposed method, a quadratic rate distortion model with high-computational complexity is not used when quantization parameter values are being decided. Instead, for low-computational complexity, average complexity weight values of frames are used to calculate QP. In addition, the rate control is designed with the hardware for fast QP decision. The execution cycle and gate count of the proposed architecture were reduced about 65% and 85% respectively compared with those of previous architecture. The proposed RC was implemented using Verilog HDL and synthesized with UMC 0.18 μm standard cell library. The synthesis result shows that the gate count of the architecture is about 19.1k with 108MHz clock frequency.

Keywords : H.264/AVC, rate control, real time, low-computational, MAD

* 학생회원, ** 정회원, 한밭대학교 정보통신공학과

(Dept. of Information and Communication Engineering, Hanbat National University)

※ 본 연구는 교육과학기술부와 한국연구재단의 지역혁신인력양성사업 및 지식경제부 출연금으로 수행한 ETRI SW-SoC 융합 R&BD 센터와의 공동연구 결과임.

접수일자: 2012년1월18일, 수정완료일: 2012년11월26일

I. 서 론

H.264/AVC 기술은 ISO/IEC와 ITU-T가 공동으로 설립한 JVT(Joint Video Team)에 의해 공동 개발 및 발표한 표준으로 압축 성능이 우수한 비디오 부호화 표준이다. H.264/AVC 기술은 MPEG-1, MPEG-2, MPEG-4 비디오로 이어지는 기본 압축 흐름은 유사하게 사용하였지만 세부 알고리즘은 크게 변경 되었다^[1].

새롭게 채택된 기술 덕분에 다른 부호화 표준보다 동일 화질을 기준으로 높은 압축률을 이루었으나 다른 비디오 부호화 표준과 비교하여 부호화의 복잡도가 증가하는 단점이 있다. 따라서 H.264/AVC의 세부 알고리즘을 효율적으로 구현하는 기술이 매우 중요하게 되었다.

영상을 부호화 하여 실시간 전송할 경우 영상의 특성에 따라 부호화되는 비트량이 시간적으로 변동이 심할 수 있다. 이런 경우 제한된 채널 용량과 요구되는 비트량에 의해 복호기의 버퍼에 저장 되는 영상 데이터는 오버플로우(Overflow) 또는 언더플로우(Underflow)가 발생하여 화질의 열화 및 데이터 손실을 초래할 수 있다. 따라서 이러한 제한된 환경에서 영상의 품질과 부호화기의 성능 향상을 위해 비트율 제어는 필수적이라 할 수 있다.

비트율 제어는 선형 평균 절대 차이(Mean Absolute Difference)예측 방법을 사용하여 기본 유닛(BU:Basic Unit)의 왜곡 값을 구하며, 이차원 비트율 왜곡(Quadratic Rate Distortion) 모델을 기반으로 발생 비트와 양자화 왜곡의 비율을 효율적으로 조절하여 각 기본 유닛을 위한 QP(Quantization Parameter) 값을 결정한다. 하지만 이러한 방법을 소프트웨어로 구현했을 시, 영상을 부호화 하여 실시간 전송 할 경우 이차원 R-D 모델과 선형 MAD 예측 방법의 많은 연산량으로 인해 영상의 실시간 전송에 제약이 발생한다^[2].

또한, 모델 파라미터 기반으로 비트율을 제어할 경우 많은 연산량을 필요로 하기 때문에 하드웨어로 설계하기에 부적합하며^[3], Kuo가 제안한 하드웨어/소프트웨어 통합 설계 방식은 프레임 단위 Qstep 결정과 기본 유닛의 타겟비트를 생성 부분을 소프트웨어로 구현하여 사용하여 빠른 Qstep 결정이 어렵고 많은 연산이 필요하다^[4]. 따라서 본 논문은 비트율 제어의 성능을 향상시키기 위한 Qstep 결정 방법과 낮은 연산량을 갖는 비트율 제어의 하드웨어 구조를 제안한다. 본 논문에서는

기존 이차원 R-D 모델을 사용하지 않고, 프레임의 평균 복잡도 가중치 값을 이용하여 Qstep 값을 구함으로써, 많은 연산량을 감소시킨다. 또한 빠른 Qstep 결정을 위해 비트율 제어를 하드웨어로 설계한다.

본 논문의 구성은 다음과 같다. II장에서는 H.264/AVC의 비트율 제어 알고리즘에 대한 소개, III장에서는 제안한 비트율 제어의 구조에 대해 설명하고, IV장에서는 제안한 비트율 제어의 실험 결과를 제시한다. 마지막으로 V장에서는 본 논문의 결론을 맺는다.

II. 기존의 비트율 제어 방식

1. 비트율 제어 참고 소프트웨어

그림 1은 H.264/AVC 참조 모델의 비트율 제어 방법을 나타내며, 일반적인 처리 과정은 다음과 같다.

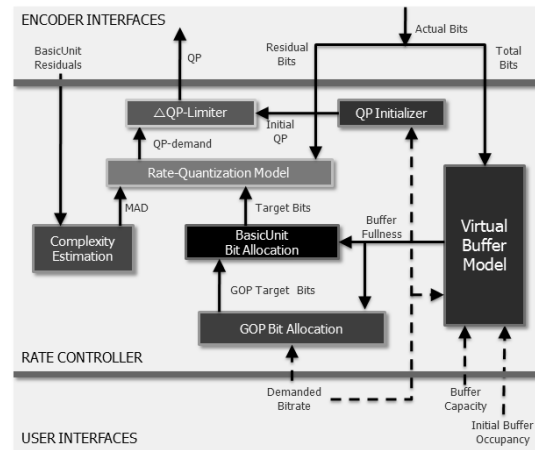


그림 1. H.264/AVC 참조 소프트웨어 블럭도
Fig. 1. Block diagram of H.264/AVC reference software.

- 가. 유동 트래픽 모델(가상 버퍼)을 이용하여 현재 화면의 목표 비트를 할당한다.
 - 나. 남아있는 비트를 부호화할 전체 부호화 단위에 균등하게 할당한다.
 - 다. 현재 부호화할 단위의 왜곡 값을 선형 예측 모델을 이용하여 결정한다.
 - 라. 위의 과정을 통해 정해진 값들을 기반으로 이차원 비트율 왜곡 모델을 통해 양자화 스텝을 구한다^[5].
 - 마. 왜곡 예측을 위한 선형 모델
- H.264/AVC 부호화기는 QP 값 결정을 위해 이차원

$$a1 = \begin{cases} \sum_{k=1}^w \frac{MAD_{k,i}}{w \times MAD_{k,i-1}}, & \text{if } MAD_{k,i-1} = MAD_{k,i} \\ \frac{\left(\sum_{k=1}^w w MAD_{k,i} \times MAD_{k,i-1} \right) - \left(\sum_{k=1}^w MAD_{k,i} \right) \left(\sum_{k=1}^w MAD_{k,i-1} \right)}{\left(\sum_{k=1}^w MAD_{k,i-1}^2 \right) - \left(\sum_{k=1}^w MAD_{k,i} \right)^2}, & \text{else.} \end{cases} \quad (2)$$

$$a2 = \begin{cases} 0, & \text{if } MAD_{k,i-1} = MAD_{k,i} \\ \frac{\left(\sum_{k=1}^w MAD_{k,i} \right) \left(\sum_{k=1}^w MAD_{k,i-1}^2 \right) - \left(\sum_{k=1}^w MAD_{k,i} \times MAD_{k,i-1} \right) \left(\sum_{k=1}^w MAD_{k,i-1} \right)}{\left(\sum_{k=1}^w MAD_{k,i-1}^2 \right) - \left(\sum_{k=1}^w MAD_{k,i-1} \right)^2}, & \text{else.} \end{cases}$$

비트율 왜곡(Quadratic Rate-Distortion) 모델을 사용한 다. 이차원 비트율 왜곡 모델은 결정된 모드의 MAD 정보를 바탕으로 QP 값을 산출하며, 필요한 MAD 정보는 선형모델을 통해 제공 받는다. 선형 MAD 예측 모델은 현재 매크로블록의 MAD를 예측하기 위해 현재 프레임 매크로블록의 MAD_{cb}와 동일한 위치의 이전 프레임 매크로블록 MAD_{pb}를 이용하여 계산된다. 현재 매크로블록의 MAD 계산 방법은 식 (1)과 같다.

$$MAD_{cb} = a_1 \times MAD_{pb} + a_2 \quad (1)$$

여기서, a1, a2는 선형 모델의 모델 파라미터로써, 각 매크로블록의 부호화를 수행하여 갱신된다. 모델 파라미터를 구하는 방법은 식 (2)와 같다.

여기서 MAD_{k,i}는 i 번째 프레임 내 k 번째 매크로블록의 실제 MAD 값이다. 위의 모델 파라미터는 각 매크로블록 마다 갱신되며, 갱신과정은 다음과 같다.

(1) 슬라이딩 윈도우 데이터 포인트 선택

이 방법은 선형 모델 갱신 할 때 사용되어 영상 전환 시 부드러운 효과를 낸다. 선형 모델 계산 시 참조되는 기본 유닛 수 w는 식 (3)과 같이 결정된다.

$$w = \begin{cases} \text{ceil} \left(\frac{MAD_{j,k}}{MAD_{j,k-1}} \times W_{\max} \right), & \text{if } (MAD_{j,k-1} > MAD_k) \\ \text{ceil} \left(\frac{MAD_{j,k-1}}{MAD_{j,k}} \times W_{\max} \right), & \text{Otherwise} \end{cases} \quad (3)$$

여기서 k는 기본 유닛의 번호, WMAX는 슬라이딩 윈도우의 최대 값을 나타내며, 기본 값은 20이다. w 값은 장면 전환 발생 후 증가 한다

(2) 데이터 통계적 제거

현재 매크로블록의 MAD 예측에 참조되는 양는 기본유닛을 제거하기 위해 식 (4)를 사용한다.

$$\begin{aligned} std &= \sqrt{\frac{1}{w} \sum_{k=0}^w (a_1 \times MAD_{k,i-1} + a_2 - MAD_{k,i})^2} \\ e(k) &= |a_1 \times MAD_{k,i-1} + a_2 - MAD_{k,i}| \end{aligned} \quad (4)$$

만약 e(k)가 std보다 클 경우, 기본유닛 k는 MAD 예측 모델 갱신에서 제거된다. 이와 같은 과정을 통해 MAD 예측을 하는 선형 모델은 복잡하고 많은 연산 처리를 필요로 한다.

가. 이차원 비트율 왜곡 모델

이차원 비트율 왜곡 모델(Quadratic R-D model)^[6]은 H.264 부호화기에서 QP 값을 설정하기 위해 사용하는 모델로써, 레벨 비트 할당에서 계산된 목표 비트량과 MAD 값, 두 개의 모델 파라미터를 기반으로 현재 기본 유닛에 대한 Qstep 값을 식 (5)와 같이 계산한다.

$$T(n_{i,j}) = \frac{X1 \times MAD(n_{i,j})}{Q_{step}(n_{i,j})} + \frac{X2 \times MAD(n_{i,j})}{Q_{step}^2(n_{i,j})} \quad (5)$$

이차원 모델 파라미터는 현재 프레임의 부호화 결과를 기반으로 계속 갱신된다. 갱신 과정은 다음과 같다.

(1) 슬라이딩 윈도우 데이터 포인트 선택

이 방법은 R-D 모델 갱신 할 때 사용되어 영상 전환 시 부드러운 효과를 내며 식 (3)과 같이 계산한다.

(2) 데이터 통계적 제거

이차원 모델 파라미터 X1과 X2는 최소 제곱 추정을

사용함으로써 계산된다. 데이터 포인트는 식 (6)과 같이 통계적 측정을 통해 참조되지 않을 기본 유닛을 제거한다.

$$std = \sqrt{\frac{1}{w} \sum_{k=0}^w \left(\frac{X1 \times MAD_{j,k}}{Q_{step_{j,k}}} + \frac{X2 \times MAD_{j,k}}{Q_{step_{j,k}}^2} - R_{j,k} \right)^2}$$

$$e(j,k) = \left| \frac{X1 \times MAD_{j,k}}{Q_{step_{j,k}}} + \frac{X2 \times MAD_{j,k}}{Q_{step_{j,k}}^2} - R_{j,k} \right| \quad (6)$$

여기서 R은 k 번째 기본 유닛의 실제 부호화된 비트를 나타낸다. 만약 e(j,k)가 std보다 클 경우, k 번째 기본 유닛은 모델 파라미터 계산 시 참조 유닛에서 제외된다.

(3) 모델 파라미터 갱신

이차원 모델 파라미터 X1, X2를 갱신을 위해 식 (7)을 사용한다. 여기서 n은 선택된 이전 기본유닛의 수를 나타낸다. 이와 같은 과정을 통해 Qstep을 결정하는 R-D 모델은 복잡하고 많은 연산 처리를 필요로 하며, 모델 파라미터에 따라 영상의 특성이 나타난다.

$$X2 = \frac{n \sum_{k=1}^n R_k - \left(\sum_{k=1}^n Q_k^{-1} \right) \left(\sum_{k=1}^n Q_k R_k \right)}{n \sum_{k=1}^n Q_k^{-2} - \left(\sum_{k=1}^n Q_k^{-1} \right)^2}$$

$$X1 = \frac{\sum_{k=1}^n Q_k R_k - X2 \times \sum_{k=1}^n Q_k^{-1}}{n} \quad (7)$$

III. 제안하는 비트율 제어

1. 하드웨어 구조

본 논문에서 제안하는 하드웨어 구조는 그림 2와 같이 8개의 모듈로 구성되며, 두 개의 비트할당 모듈과 가상버퍼 모듈은 H.264 참조 모델의 알고리즘^[7~8]을 사용하며, 양자화 파라미터 갱신 및 양자화 파라미터 결정을 위한 모듈은 Kuo^[8]의 알고리즘을 사용한다. 많은 연산량을 필요로 하는 두 개의 MAD 예측 모듈과 Qstep을 결정하는 모듈을 제안하여 비트율 제어를 하드웨어로 설계한다.

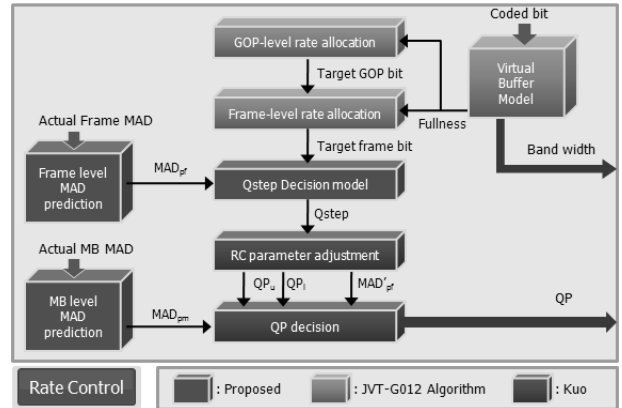


그림 2. 제안하는 비트율 제어 구조
Fig. 2. Architecture of the Proposed Rate Control.

가. 프레임 레벨 MAD 예측

프레임 레벨 MAD 예측은 많은 연산 처리를 필요로 하는 선형 예측 모델과 달리 그림 3과 같이 부호화가 완료된 프레임의 MAD와 QP를 기반으로 비교적 간단하게 결정된다. 결정 방법은 식 (8)과 같다. MAD_{p,f}는 현재 프레임에 대한 예측된 MAD를 나타내며, MAD_{a,f}와 QP는 이전에 부호화된 프레임의 실제 MAD와 QP 값이다. \overline{QP} 는 현재 프레임 전까지 부호화된 프레임의 평균 QP 값을 나타낸다. 현재 프레임에 대한 MAD는 프레임당 한번만 수행하며, 예측된 값은 매크로블록의 QP 값을 결정하는데 사용된다.

$$MAD_{p,f[i]} = MAD_{a,f[i-1]} \times (1 + ((QP[i-1] - \overline{QP}[i-1]) / 64)) \quad (8)$$

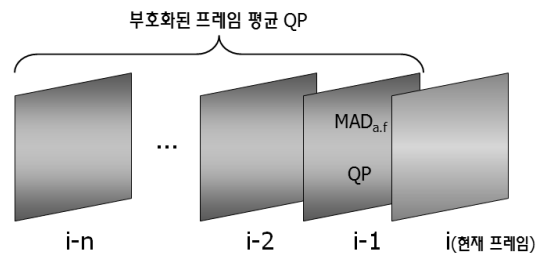


그림 3. 프레임 단위 MAD 예측
Fig. 3. Frame-level MAD Prediction.

나. 매크로블록 레벨 MAD 예측

매크로블록 MAD 예측은 기존의 복잡하고 많은 연산 대신 주변 매크로블록을 사용하여 현재 매크로블록의 MAD를 예측하는 방식이다. 참조되는 주변 매크로

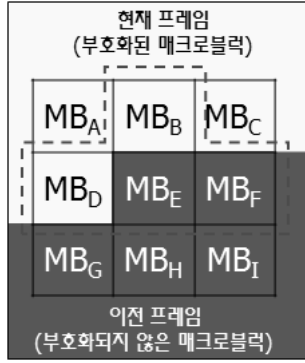


그림 4. MB_E MAD 예측을 위한 주변 매크로블록
Fig. 4. Surrounding MBs to predict the MAD of MB_E.

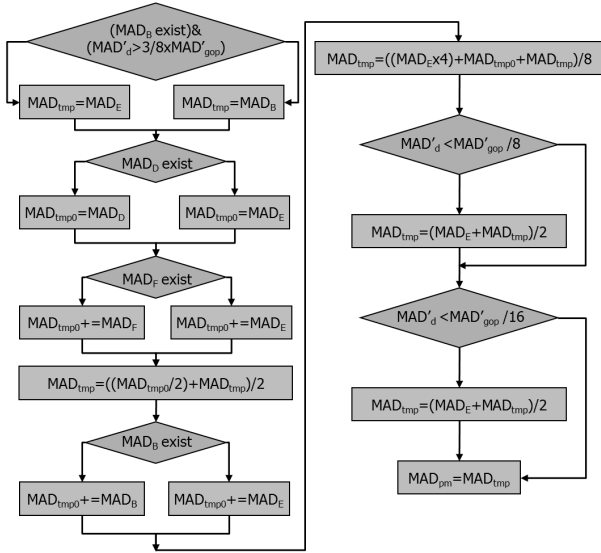


그림 5. 제안하는 MB 단위 MAD 예측 흐름도
Fig. 5. Flowchart of the MB-level MAD prediction.

블록은 그림 4와 같다. 16x16 매크로블록은 래스터 방식으로 부호화하므로 좌에서 우로, 위에서 아래로 부호화한다. 현재 부호화 매크로블록이 MB_E 일때 MB_A, MB_B, MB_C, MB_D는 부호화된 매크로블록이며, MB_E, MB_F, MB_G, MB_H, MB_I는 부호화되지 않는다. MB_E, MB_F, MB_G, MB_H, MB_I는 부호화되지 않는 매크로블록이므로 참조 블록으로 사용하기 위해 이전 부호화된 프레임에 동일한 위치의 매크로블록을 사용한다.

현재 예측할 매크로블록 MB_E를 기준으로 참조 매크로블록으로 사용되는 블록은 MB_B, MB_D, MB_E, MB_F이며, 4개의 매크로블록 MAD를 바탕으로 현재 매크로블록 MAD를 예측하는 방법은 그림 5와 같다. 최종 결정되는 MAD_{pm}은 참조되는 매크로블록의 존재 여부와 MAD'_d, MAD'_{gop}에 따라 결정된다. 여기서 MAD'_d와 MAD'_{gop}는 영상 전환 또는 영상 내 많은 움직임 찾기 위해 사용되며, 두 변수의 값에 따라 가중치를 적용한다. MAD'_d와 MAD'_{gop}를 결정하는 방법은 식 (9)와 같다. 여기서 i는 프레임, n은 매크로블록 번호를 나타내며, MAD_{am}와 MAD_{af}는 각각 매크로블록과 프레임의 실제 MAD이다.

다. 프레임 레벨 Qstep 결정

Qstep 결정은 프레임의 목표비트를 결정하기 위해 사용되는 프레임의 복잡도 값을 활용하여, 기존 이차원 비트율 왜곡 모델보다 연산량을 확연히 감소시켰다. Qstep 결정 방법은 식 (10)과 같다.

$$Qstep = \frac{\alpha \times MAD(n_{i,j})}{f(n_{i,j})} \quad (10)$$

여기서 MAD는 프레임 레벨에서 예측된 MAD, f는 프레임의 목표비트를 나타낸다. α는 식 (11)에서 추출된 프레임 복잡도 가중치를 식 (12)를 통해 결정한다.

$$\begin{aligned} \widetilde{W}_p(n_{i,j}) &= \frac{W_p(n_{i,j})}{8} + \frac{7 \times \widetilde{W}_p(n_{i,j-1})}{8} \\ \widetilde{W}_b(n_{i,j}) &= \frac{W_b(n_{i,j})}{8} + \frac{7 \times \widetilde{W}_b(n_{i,j-1})}{8} \\ W_p(n_{i,j}) &= S_p(n_{i,j}) Q_p(n_{i,j}) \\ W_b(n_{i,j}) &= \frac{S_b(n_{i,j}) Q_b(n_{i,j})}{1.3333} \end{aligned} \quad (11)$$

위의 수식은 각각 P 프레임과 B 프레임의 현재 프레임과 이전 프레임의 가중치를 두어 복잡도를 계산한다. 이러한 복잡도 계산에 필요한 나누기 연산은 쉬프트만을 사용하여 구현하였다.

$$\begin{aligned} MAD'_d &= \begin{cases} |MAD_{am}[i,n-1] - MAD_{am}[i-1,n-1]|, & \text{if } i=1 \text{ and } n \geq 1 \\ (MAD'_d[i,n-1] + |MAD_{am}[i,n-1] - MAD_{am}[i-1,n-1]|)/2, & \text{else} \end{cases} \\ MAD'_{gop} &= \begin{cases} |MAD_{af}[i-1], & \text{if } i=1 \text{ and } n \geq 1 \\ (MAD'_{gop}[i-1] \times \alpha + MAD_{af}[i-1] \times \beta), & \text{else} \end{cases} \end{aligned} \quad (9)$$

$$\alpha = \widetilde{W}_p(n_{i,j}) + \widetilde{W}_b(n_{i,j}) \gg 2 \quad (12)$$

위의 식을 통해 결정된 α 는 최종적으로 Qstep를 결정하는 모델에 사용된다.

라. QP 결정

프레임 레벨에서 결정된 Qstep 값은 식(13)을 통해 QP 값을 결정하기 위한 중간 결과 값으로 변환된다.

$$QP_d = 4 + 6 \times \log_2(Q_{step}) \quad (13)$$

변환된 중간 QP 값에는 연산에 의한 오차를 포함하여, 식(14)에서 결정된 QP_u 와 QP_l 을 바탕으로 오차값을 보정하기 위해 식 (15)와 같이 MAD_{pf} 값을 수정한다.

$$\begin{aligned} QP_u &= \text{ceil}(QP_d) \\ QP_l &= QP_u - 1 \end{aligned} \quad (14)$$

$$MAD'_{pf} = \begin{cases} MAD_{pf} \times 0.75, & \text{if } \frac{3QP_u + QP_l}{4} < QP_d < QP_u \\ MAD_{pf} \times 1.25, & \text{if } QP_l < QP_d < \frac{3QP_l + QP_u}{4} \\ MAD_{pf}, & \text{else.} \end{cases} \quad (15)$$

수정된 MAD'_{pf} 값은 매크로블록 MAD 예측으로부터 출력된 MAD_{pm} 을 이용하여 식 (16)의 연산으로 프레임과 매크로블록 단위의 QP 값을 결정한다.

$$QP[i,n] = \begin{cases} QP_u, & \text{if } MAD_{pm}[i,n] > MAD'_{pf}[i,n] \\ QP_l, & \text{else.} \end{cases} \quad (16)$$

IV. 실험 및 고찰

본 논문에서는 QCIF영상(174×144, 300프레임, 4:2:0)을 이용하여 H.264/AVC 참조 소프트웨어 JM13.0^[9]에서 비트스트림을 추출하고 테스트한 결과, 제안한 구조가 정상적으로 동작함을 확인하였다.

제안하는 비트율 제어는 Verilog HDL로 설계하고 IDE에서 지원하는 Synopsys사의 Design Compiler를 이용하여 UMC 0.18 μ m CMOS 공정으로 합성하였으며, 기존 비트율 제어 방법과 비교한 결과는 표 1과 같다.

제안한 비트율 제어는 Kuo가 제안한 구조와 비교하

표 1. 하드웨어 비교

Table 1. Hardware cost comparison.

Method	Hardware			Software	
	Implementation / Frequency	Gate Count	Cycles	Cycles	
Kuo ^[8]	Frame level	UMC 0.18 μ m / 100 MHz	N/A	N/A	9108
	MB level		5k	6	2204
Proposed	Frame level	UMC 0.18 μ m / 108 MHz	254	1	N/A
	MB level		3.9k	6	N/A
Gu ^[10]	N/A	144k	608/MB	N/A	N/A
Proposed	UMC 0.18 μ m / 106 MHz	19.1k	220/MB	N/A	N/A

표 2. MAD 예측 내 연산기 수 비교

Table 2. Number of operations in MAD predictions.

Operation	Kuo		Proposed	
	MB level	Frame level	MB level	Frame level
Shift	13	2	13	1
Compare	3	0	3	1
Add-sub	24	5	19	5
Multiplication	0	1	0	1
Division	0	1	0	0

여 공정은 0.18 μ m와 같으며 동작주파수는 8% 향상되었다. 또한 Gu가 제안한 구조는 144k의 게이트를 사용하였지만, 본 논문에서 제안한 구조는 19.1k의 게이트를 사용함으로써 게이트 수를 85% 감소시켰다. 표 2는 MAD 예측 레벨의 연산의 개수를 기준으로 기존 구조와 제안한 구조를 비교하였다.

제안한 구조는 Kuo보다 MB level에서 덧셈기 및 뺄셈 연산기가 5개가 감소하였고, Frame level에서는 로직의 큰 면적과 많은 게이트 수를 필요로 하는 나눗셈기 대신 쉬프트를 사용하여 비트율 제어의 성능을 향상시켰다. 표 3은 QP를 결정하는 기존 이차원 R-D 모델과 제안한 Qstep Decision model에 대해 PSNR을 비교하였다. 테스트는 4 종류의 영상은 QCIF, 300 프레임, 4:2:0 포맷이며 각각 48, 64, 128kb/s로 설정하여 테스트하였으며, 2 종류의 영상은 각각 HD, FullHD 사이즈와 300 프레임, 4:2:0 포맷을 갖는다.

테스트 결과, 제안한 구조의 PSNR은 기존 구조에 비해 평균 0.2dB가 낮게 측정되었다. 하지만 Kuo의 MB level 하드웨어 구조보다 게이트 수가 22% 감소하였고, Gu의 게이트 수는 85%가 감소하였다. 또한 제안한 구조의 평균 수행 사이클 수는 Gu에 비해 약 64% 가량 감소하였다.

표 3. Qstep 결정 모델의 PSNR 비교
Table 3. PSNR comparison of Qstep Decision model.

Sequence	Bit-Rate (kb/s)	Quadratic R-D model ^[9]		Qstep Decision model	
		Coded bit	PSNR	Coded bit	PSNR
foreman (QCIF)	48	480,840	30.65	480,792	30.41
	64	637,920	32.25	642,272	31.95
	128	1,277,776	36.19	1,270,584	35.85
akiyo (QCIF)	48	479,104	39.33	478,896	39.37
	64	638,736	40.85	638,040	40.80
	128	1,276,280	45.76	1,275,336	45.76
carphone (QCIF)	48	613,112	31.58	608,792	31.54
	64	816,776	32.83	811,088	32.75
	128	1,619,584	35.88	1,619,584	35.88
coastguard (QCIF)	48	479,912	29.39	476,600	29.08
	64	640,128	30.44	635,144	30.14
	128	1,279,464	33.16	1,270,992	32.81
stockholm (HD)	256	2,589,145	32.56	2,588,169	32.14
tractor (FHD)	256	2,556,144	33.14	2,654,369	32.87

V. 결 론

본 논문은 비트율 제어의 성능을 향상시키기 위한 QP 결정 방법과 낮은 연산량을 갖는 비트율 제어의 하드웨어 구조를 제안하였다. 제안하는 비트율 제어는 기존 연산 방식인 이차원 R-D 모델을 사용하지 않고, 프레임의 평균 복잡도 가중치 값을 이용하여 QP를 결정하는 Qstep Decision model을 사용하며, 기존 구조가 사용하는 MAD 예측 방법을 최적화하여, 많은 연산량을 감소시킨다. 또한 빠른 QP 결정을 위해 비트율 제어 전체를 하드웨어로 설계한다.

제안하는 비트율 제어는 프레임 레벨 MAD 예측, 매크로블록 레벨 MAD 예측, GOP 비트 할당, 프레임 비트 할당, 가상버퍼, Qstep 결정 모델, 비트율 제어 파라미터 갱신, QP 결정 모듈을 포함하며 총 8개의 모듈로 구성되며, MAD 예측 연산 시 결과 값의 정확도를 위해 정수부 7비트, 소수부 9비트를 갖는 고정소수점 방식을 사용하여 설계하였다. 검증용 테스트 영상은 176x144 QCIF, 4:2:0 포맷을 갖는 Akiyo, Foreman, Carphone, Coastguard와 HD, Full-HD 4:2:0 포맷을 갖는 stockholm과 tractor를 사용하였다.

비트율 제어는 Verilog-HDL을 이용하여 설계하였으며, Synopsys사의 Design Compiler를 이용하여 UMC 공정 0.18 μm 셀 라이브러리로 합성한 결과, 최대 동작

주파수는 108 MHz이고, 게이트 수는 19.1k이다. 따라서 제안한 구조는 기존 구조 보다 게이트 수가 85% 만큼 감소하였고, 매크로블록 당 QP를 결정하는데 평균 220 사이클 수가 소요되어 기존 구조보다 64% 단축됨을 확인하였다.

참 고 문 헌

- [1] 박기현, *이론과 실무의 조화 코덱의 세계로의 초대*, 홍릉과학출판사, 2007년.
- [2] 손남례, *장면전환에 효율적인 H.264/AVC 비트율 제어 기법*, 대한전자공학회, 제44권, 제1호, pp. 26-39, Jan. 2007.
- [3] 오호성, *H.264/AVC 알고리즘 이해와 프로그램 분석*, 두양사, 2009년.
- [4] T. Chiang, and Y. Q. Zhang, *A new rate control scheme using quadratic rate distortion model*, IEEE Trans. Circuits Syst. Video Tech., vol. 7, pp. 246-250, Feb. 1997.
- [5] 정진우, *실시간 부호화를 위한 모델 파라미터 기반 일정 화질 비트율 제어 기법*, 대한전자공학회, 제 45권, 제3호, pp. 93-102, May. 2008.
- [6] Adaptive Basic unit Layer Rate Control for JVT, document JVT-G012.doc, Joint Video Team(JVT) of ISO/IEC MPEG and ITU-T VCEG, Pattaya, Thailand, Mar. 2003.
- [7] Text Description of Joint Model Reference Encoding Methods and Decoding Concealment Methods, JVT-GO079.doc, Joint Video Team(JVT) of ISO/IECMPEG and ITU-TVCEG Busan, Korea, Apr. 2005.
- [8] C. H. Kuo, *Hardware/Software Codesign of a Low-Cost Rate Control Scheme for H.264/AVC*, IEEE Trans. Circuits Syst. Video Technol., vol. 20, pp. 250-261, Feb. 2010.
- [9] Joint Video Team Reference Software JM 13.0.
- [10] J. W. Gu *A high quality hardwired rate controller for H.264/AVC real-time video encoding*, MS thesis, Nat. Tsing Hua Univ., Hsinchu, Taiwan, Jul. 2007.

저 자 소 개



김 창 호(학생회원)
 2010년 한밭대학교 공과대학
 정보통신공학과 공학사
 2012년 한밭대학교
 정보통신전문대학원
 정보통신공학과 공학석사
 2012년 현재 (주)실리콘웍스
 연구소 연구원

<주관심분야 : 영상처리, SoC 플랫폼 설계 및 검증>



류 광 기(정회원)-교신저자
 1986년 한양대학교 전자공학과
 공학사
 1988년 한양대학교 대학원
 전자공학과 공학석사
 2000년 한양대학교 대학원
 전자공학과 공학박사
 1991년~1994년 육군사관학교 교수부 전자공학과
 전임강사
 2000년~2002년 한국전자통신연구원 시스템 IC
 설계팀 선임연구원
 2003년~현재 한밭대학교 정보통신공학과 교수
 <주관심분야 : SoC 플랫폼 설계 및 검증, 하드웨
 어/소프트웨어 통합설계 및 검증, 멀티미디어 코
 덱 설계>