

H.264 Encoder Hardware Chip설계

김 종 철** 서 기 범**

*우송대학교

A design of Encoder Hardware Chip For H.264

jong chul Kim** Ki-bum Suh**

*Electronic Dept. Graduate School, Woosung University

E-mail : jckim@wsu.ac.kr

요 약

본 논문에서는 AMBA 기반으로 사용될 수 있는 H.264용 Encoder Hardware 모듈(Intra Prediction, Deblocking Filter, Context-Based Adaptive Variable Length Coding, Motion Estimation)을 Integration하여 설계 하였다. 설계된 모듈은 한 매크로 블록당 최대 440 cycle내에 동작한다. 제안된 Encoder 구조를 검증하기 위하여 JM 9.4부터 reference C를 개발하였으며, reference C로부터 test vector를 추출하여 설계 된 회로를 검증하였다. 제안된 회로는 최대 166MHz clock에서 동작하며, 합성결과 Chartered 0.18 um 공정에 램 포함 약 180만 gate 크기 이다. MPW제작시 chip size 6x6mm의 크기와 208 pin의 Package 형태로 제작 하였다.

ABSTRACT

In this paper, we propose H.264 Encoder integrating Intra Prediction, Deblocking Filter, Context-Based Adaptive Variable Length Coding, and Motion Estimation encoder module.

This designed module can be operated in 440 cycle for one-macroblock. To verify the Encoder architecture, we developed the reference C from JM 9.4 and verified the our developed hardware using test vector generated by reference C. The designed circuit can be operated in 166MHz clock system, and has 1800K gate counts using Chartered 0.18 um process including SRAM memory. Manufactured chip has the size of 6x6 mm and 208 pins package.

키워드

H.264, Encoder

1. 서 론

H.264는 매우 높은 데이터 압축률을 가지는 디지털 비디오 코덱 표준으로 MPEG-4 part 10 또는 AVC(Advanced Video Coding)라 부르기도 한다. 이 표준은 ITU-T의 비디오 코딩 전문가 그룹(Video Coding Experts Group, VCEG)과 ISO/IEC의 동화상 전문가 그룹(Moving Picture Experts Group, MPEG)이 공동으로 JVT (Joint Video Team)을 구성하고 표준화를 진행한 결과물로 나온 것이다.

H.264/AVC는 기존의 표준(MPEG-2, H.263, MPEG-4 Part 2)과 비교했을 때, 낮은 bit-rate에서 비슷하거나 더 좋은 화질을 얻을 수 있도록

개발되었다. 그리고 구현할 때에 설계상의 유연성도 고려되었다. 또 다른 목표로서 다양한 시스템에 적용될 수 있도록 하였고, 다양한 종류의 네트워크와 시스템에서 작동하도록 하였다. [1]

본 논문에서는 설계된 베이스 프로파일 레벨 3 규격의 H.264 Encoder의 구조는 크게 Intra Prediction 모듈, Deblocking Filter 모듈, Context-Based Adaptive Variable Length Coding 모듈, Motion Estimation 모듈의 4개로 구성되어 있다. 원활한 데이터 입력과 처리를 위하여 Sram Buffer와 5단계의 pipeline을 통한 처리를 사용하여 설계 하였다. 그림 1은 설계된 H.264 Encoder의 전체 구조를 보여준 것이다.

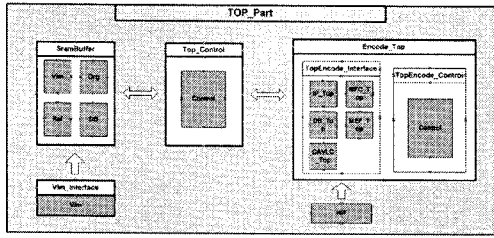


그림 1. 설계된 H.264 Encoder의 전체구조

II. 본 론

설계된 H.264 Encoder의 구조를 설명 하기 앞서 H.264 Encoder의 각각 모듈에 대한 설명을 하고자 한다. 그림 2는 H.264의 인코딩 과정을 나타낸다. 입력 프레임은 매크로블록 단위로 처리되며 각 매크로블록은 화면 내(Intra) 모드 또는 화면 외(Inter) 모드로 인코딩된다. 영상을 압축할 경우 모드를 선택하게 되고 모드에 따라 생성된 오차블록의 영상데이터를 정수 변환 및 양자화 과정을 거쳐 엔트로피 코딩을 한다. 기준영상을 생성하기 위해 양자화된 영상의 데이터를 역양자화 및 역정수 변환 단계를 거쳐서 복원하고 루프 필터를 거친 후 기준 영상으로 사용한다. 움직임 추정기는 인터 예측에서 사용되며 복원된 영상과 다음에 입력되는 영상의 중복을 비교한후 움직임 벡터를 출력 한다. 압축효율의 향상을 위하여 주변 영상데이터를 이용하여 현재블록의 영상데이터를 예측한다. 이를 화면내 예측이라고 한다.

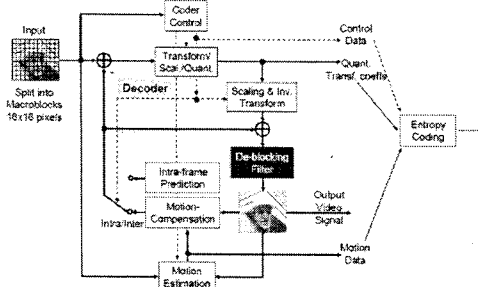


그림 2. H.264 Encoder 구조

위에서 말한 처리 과정에 필요한 5개(Intra Prediction 모듈, Deblocking Filter 모듈, Context-Based Adaptive Variable Length Coding 모듈, Motion Estimation 모듈(MEF, MEC))의 모듈에 관하여 상세 설명을 하겠다.

Intra Prediction 모듈은 움직임을 예측하여 입력 값 (original data)과의 차이를 최소화하는 가장 좋은 mode값을 출력하여 DCT & IDCT값과 더하여 복원된 값을 출력한다. 크게 입력 부분과 prediction 과정을 수행하는 부분, Inter prediction 값을 입력하는 부분, 출력된 결과물

intra prediction ram에 저장하고 loop filter의 입력으로 넘겨주는 부분, VLC 모듈로 coefficient 데이터를 넘겨주는 부분으로 나눌 수 있다.

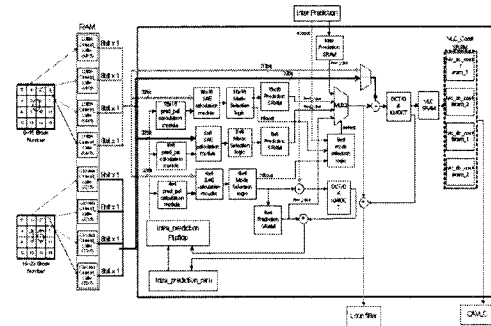


그림 2-1. Intra prediction 모듈의 하드웨어 구조

입력 부분은 MEF/MC에서 사용하는 ORG Data를 사용하기 위해 ORGY와 ORGCbCr RAM 두 개씩 쌍으로 이루어지며, Data의 처리는 자동적으로 이루어 진다. 그림 2-1은 Intra prediction 모듈의 기능을 하드웨어로 구현한 구조도 이다.

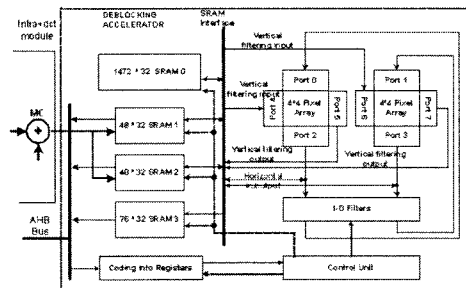


그림 2-2. Deblocking Filter 모듈의 하드웨어구조

Deblocking Filter 모듈은 엔코더 모드에서 외부 메모리와 데이터 입출력 기능, REC 결과 영상 저장 및 출력 기능, 디코더 모드에서 REC 결과 영상 저장 및 출력 기능, REC 결과 영상에 대한 디블러킹 처리 및 출력 기능을 수행하며 블럭도는 그림 2-2에서 볼수 있다.

기본 모듈과 달라진 점은 SRAM1, SRAM2, SRAM3를 듀얼 버퍼링하므로, REC 데이터 받는 시간을 단축시키고, DMA를 이용하여 SDRAM에 저장하는 작업을 DB동작과 동시에 함으로써 SDRAM에 저장하는 시간을 단축 하였다.

Context-Based Adaptive Variable Length Coding 모듈은 encoder mode에서 Macroblock 단위의 정보를 부호화하는데 사용한다. NAL unit 과 Annex B.1 의 stream 생성을 지원한다. Macroblock보다 상위의 syntax에 대한 bitstream의 generation은 firmware에서 수행하며, 이 비트 스트림의 생성시 AMBA interface를 통하여 32bit (비트수 지정가능)단위로 공급한다. Coeff_token

을 생성하기 위한 Neighboring information 은 내부에서 발생한다. 그림 2-3은 CAVLC 모듈의 하드웨어 구조를 나타낸 것이다.

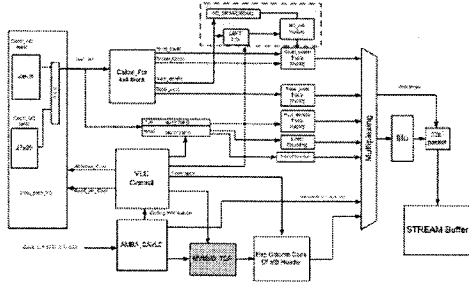


그림 2-3. CAVLC 모듈 하드웨어 구조

Motion Estimation 모듈은 다시 두 개의 모듈로 구분 되었는데 MEC 와 MEF/MC 모듈로 되어 있다.

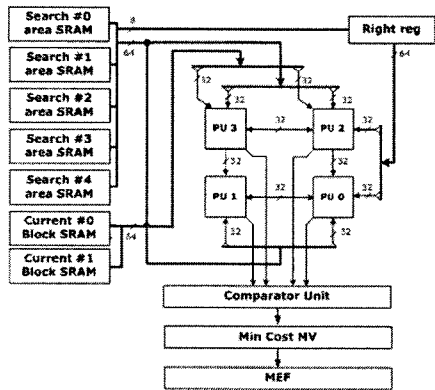


그림 2-4. MEC 모듈의 하드웨어 구조

MEC 기능 모듈은 움직임 벡터를 계산하는 기능을 가진 구조이다[4]. 움직임 벡터를 구하고자 하는 현재 영상내의 기준블록 데이터와 재생된 이전영상내의 대응되는 탐색영역데이터가 기준블록 및 탐색영역 데이터 메모리에 저장된다. 메모리에 저장되어 있는 기준 블록 및 탐색영역 데이터를 이용하여 2화소 단위의 움직임 탐색이 수행되며, 2화소 움직임 벡터가 구해진다. 이때 수직 및 수평 방향으로 2:1 로 샘플링된 기준블록 및 탐색영역이 사용 되어지며, 탐색범위는 -6 ~ +6 이 된다. 전체 구조에서는 Current 영역을 저장하는 메모리 2개와 Reference Data를 저장하는 메모리 5개로 구성 되어 지며, SAD 값을 계산하기 위한 PE(Processing Elements) 가 64개로 이루어져 1 Cycle 안에 8X8 블록의 SAD값을 계산할 수 있도록 이루어져 있다. 이렇게 계산 되어진 SAD값의 최소값을 구하기 위한 비교기(Comparator)로 구성되어져 있다. 여기서 구해진 Min cost 좌표는 MEF로 입력 되어 다시 계산되

어진다. 이 모듈이 1MB가 계산 되어지는데 315 cycle 소요된다.

MEF/MC은 MEC에서 탐색된 결과를 토대로 1-화소 단위의 미세 움직임 탐색을 수행한다, 수행 방법은 현재 매크로 블록과 개략적인 움직임 탐색으로 찾은 가장 근사한 위치의 이전 매크로 블록 및 주변 8개 후보 매크로 블록 사이에 9개의 후보 SAD 값을 구한 다음 중 최소인 SAD 값과 그에 해당하는 상대적 위치 좌표(1-화소 단위의 움직임 벡터)를 산출한다, 단 SAD 비교시 (0,0) 위치의 후보 SAD 값은 누산된 SAD에 96를 뺀 값을 적용한다. 만약 SAD 최소 값이 2개 이상이면 동일 행 또는 열에서 -1,0 그리고 +1 의 순으로 우선 선택되고, 행 보다 열이 우선한다. 후보 매크로블록으로 구성된 탐색 영역의 각 화소치는 재구성된 이전 영상 데이터 이다. 적용 알고리즘은 완전 탐색 블록 정합이며 탐색 범위는 수직 및 수평방향으로 -1 ~ +1 이다.

이후 1-화소 단위의 1차 미세 움직임 탐색 결과를 토대로 반화소 단위의 2차 미세 움직임 탐색을 수행한다. 수행 방법은 현재 매크로 블록과 1차 미세 움직임 탐색으로 찾은 가장 근사한 위치의 이전 매트릭 블록 주변 즉, 좌, 우, 상, 그리고 하측으로 각각 -0.5 화소 및 +0.5 화소 위치의 8개 후보 매크로 블록 사이에 8개의 후보 SAD 값을 각각 구한 다음 그 값과 1차 탐색에서 구한 최소 SAD 값을 포함하여 그 중 최소인 SAD 값과 그에 해당하는 상대적 위치 좌표(반화소 단위의 움직임 벡터)를 산출한다. 단, SAD 비교 시, 1차 움직임 탐색에서의와 같이, (0,0) 위치의 후보 SAD 값은 누산된 SAD에 96를 뺀 값을 적용한다. 후보 매크로 블록의 각 화소치는 재구성된 이전 영상의 실제 픽셀 값으로부터 보간한 것이다.

적용 알고리즘은 완전 탐색 블록 정합이면, 탐색 범위는 -0.5 ~ +0.5이다, 또한 MPEG-4 부호화 시 무제한 모드를 지원하며, 패딩 방법은 1 화소 단위의 움직임 탐색에서와 동일하다.

III. Encoder 동작과 pipeline 구조

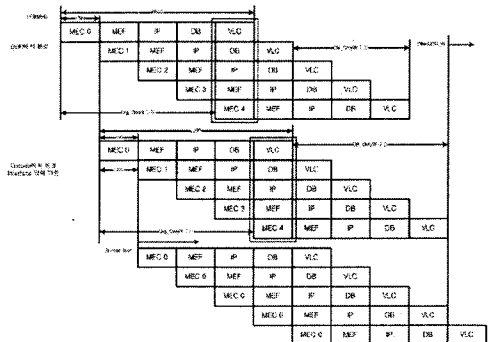


그림 3. 5단 pipeline

앞장에서 언급한 바와 같이 Encoder 구조는 5개의 큰 모듈로 구분 되어 있다. 입력 프레임은 Macro Block단위로 처리 되어 지는데 이때 1MB이 처리 되어지는 cycle은 440cycle이며, 5단의 pipeline을 통과 하여 1MB 처리되어 진다. 처음 1블럭의 처리 소요시간 이후 그림 3에서 보이는 것과 같이 다음 블록부터는 1단의 pipeline의 처리만으로도 1MB가 처리 되는 것을 볼 수 있다.

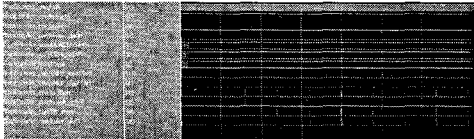


그림 3-1 I-frame control 신호

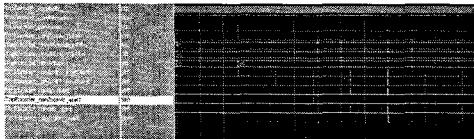


그림 3-2 P-frame control 신호

I-frame 처리시 MEC와 MEF의 처리가 불필요하지만 컨트롤러의 설계를 간단히 하기 위하여 그림 3-1처럼 앞에 두 MB처리 시간을 비워두고 시작 하게 만들었다. P-frame 시작시 MEC, MEF의 컨트롤 신호 또한 그림 3-2에서 볼 수 있다. DMA에서 데이터 입력을 원활하게 해주지 못할 경우 각각의 모듈에게 wait상태로 넘어 가게 컨트롤 하여 데이터의 정확한 입력을 요구하게 하였다.

IV. 결 론

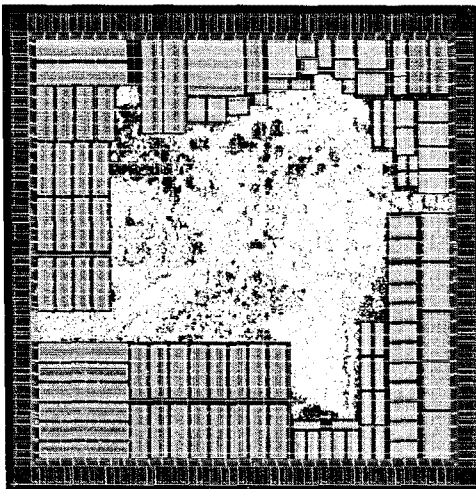


그림 4. 설계된 Encoder 모습

본 논문에서는 H.264 Encoder의 5개 Sub 모듈 (Intra Prediction, Deblocking Filter, Context-Based Adaptive Variable Length Coding, Motion Estimation(MEC, MEF))을 Integration 하여 Synopsys Design Compiler로 합성하고, Astro tool로 lay-out 하였다. 구현된 하드웨어의 동작주파수는 Encoder Top 모듈 100MHz, VIM 24MHz, Sram Buffer 166MHz 이다. 전체모듈 합성시 Chartered 0.18um 공정에 Ram 포함 약 1830k gate (Combinational area : 386k gate, Noncombinational area : 1451k gate)크기이며, chip size는 6x6mm의 208pin 으로 설계 하였다.

그림 4는 Astro Tool을 사용한 lay-out 그림이다.

참고문헌

- [1] ITU-T Rec. H.264/ISO/IEC 11496-10, "Advanced Video Coding", Final Committee Draft, Document JVT-F100, October 2004
- [2] Iain E G Richardson, "H.264 and MPEG-4 Video Compression", John Wiley & Sons, to be published late 2003
- [3] Iain E.G. Richardson "H.264 and MPEG-4 VIDEO COMPRESSION" John Wiley & Sons, Ltd. , 2003
- [4] "An Efficient Hardware Architecture of Variable Block Size Motion Estimation for Minimum Memory Bandwidth" ETRI