
H.264/SVC 복호기 C-Model 시뮬레이터 개발

Development of C-Model Simulator for H.264/SVC Decoder

정차근

호서대학교 공과대학 시스템제어공학과

Cha-Keon Cheong(cheong@hoseo.edu)

요약

본 논문에서는 최근 국제표준화가 이루어진 H.264/SVC 복호기 SoC 칩 개발을 위한 새로운 하드웨어 구조를 제안하고, 최적인 회로개발을 지원하기 위한 C-모델 시뮬레이터를 개발한다. 제안된 SVC 복호기는 표준규격의 기능들을 최적으로 처리하기 위한 하드웨어 엔진과 핵심 프로세서를 이용한 소프트웨어 등으로 구성되어 있어 기존의 임베디드 시스템으로 간단히 구현할 수 있다. 본 논문에서 제안한 복호기의 C-모델 시뮬레이터는 SVC의 스케일러블 베이스라인 프로파일을 기반으로 복잡도 감소를 위하여 B-픽처 구조를 사용하지 않는 IPPP 구조에 의한 스케일러블 만을 고려함으로써 칩 설계의 실용성을 증가시켰다. 하드웨어 구조와 C-모델 시뮬레이터의 유효성을 검증하기 위해 제안한 H.264/SVC 복호기 시스템에 대한 결과를 제시한다.

■ 중심어 : | H.264/SVC 복호기 | C-모델 시뮬레이터 | 스케일러블 베이스라인 프로파일 | 스케일러빌리티 | 하드웨어 구조 |

Abstract

In this paper, we propose a novel hardware architecture to facilitate the applicable SoC chip design of H.264/SVC which has a great deal of advancement in the international standardization in recent. Moreover, a new C-model simulator based on the proposed hardware system will be presented to support optimal SoC circuit development. Since the proposed SVC decoder is consist of some hardware engine for processing of major decoding tools and core processor for software processing, the system is simply implemented with the conventional embedded system. To improve the feasibility and applicability, and reduce the decoder complexity, the hardware decoder architecture is constructed with only the consideration of IPPP structure scalability without using the full B-picture. Finally, we present results of decoder hardware implementation and decoded picture to show the effectiveness of the proposed hardware architecture and C-model simulator.

■ keyword : | H.264/SVC Decoder | C-model Simulator | Scalable Baseline Profile | Scalability | Hardware Structure |

I. 서론

다양한 채널 환경에서 방대한 영상정보의 자유로운 이용을 촉진하기 위해, ISO/IEC MPEG과 ITU-T

VCEG의 JVT를 중심으로 DMB 방송의 표준안으로 채택되어 상용화 된 영상압축에 관한 국제표준화 규격인 H.264/AVC를 개선한 H.264/SVC의 새로운 표준규격의 연구가 활발히 진행되고 있다[1].

접수번호 : #090121-007

접수일자 : 2009년 01월 21일

심사완료일 : 2009년 03월 03일

교신저자 : 정차근, e-mail : cheong@hoseo.edu

Scalable Video Coding(SVC)은 제한된 네트워크나 채널 대역폭에 최적으로 적응한 영상품질의 전송과 채널 에러에 강인한 차등 에러 보호(Unequal Error Protection:UEP) 기능 구현 등 기존의 스케일러블 하지 않은 비디오 부호화 방법과는 차별화된 영상압축과 전송을 가능케 한다[2][3].

효율적인 SVC 알고리즘을 기반으로 혼성 클라이언트들에 영상전송이 가능하게 됨에 따라 디스플레이의 해상도, 처리능력 및 배터리 등에 따라 차별화된 서비스가 가능하다. 또한 중요성이 다른 비트스트림의 구성이 가능해져 채널에러에 강인한 효율적인 채널코덱의 구현이 용이하다. 따라서 응용에 적합한 감시 시스템이나 다양한 품질과 기능을 갖는 영상회의 시스템, 3G 시스템에서의 비디오 스트리밍 구성, 채널특성에 적응적 대응이 가능한 개선된 지상파 DMB 방송 등의 분야에 응용이 가능하다[4][5].

본 논문에서는 H.264/SVC 표준화 규격을 기반으로 복호기의 SoC 칩 개발을 위한 새로운 하드웨어 구조를 제안하고, 그 구조에 최적인 회로개발을 지원하기 위한 C-모델 시뮬레이터를 개발한다. 제안된 SVC 복호기는 표준규격의 기능을 최적으로 처리하기 위한 하드웨어 엔진과 핵심 프로세서를 이용한 소프트웨어 등으로 구성되어 기존의 임베디드 시스템으로 간단히 구현할 수 있다. 본 논문에서 제안한 하드웨어의 구조는 SVC의 스케일러블 비디오 부호화를 기반으로 복잡도 감소를 위하여 B-픽처를 사용하지 않는 IPPP 구조에 의한 스케일러블만을 고려하였다. C-모델 시뮬레이터를 통하여 제안된 하드웨어 구조의 각 엔진을 동시에 독립적으로 개발함으로써 응용 시스템에 최적인 SoC 개발 시간의 단축과 회로를 실시간으로 설계하고 검증케 함으로서 신뢰성의 확보와 다양한 미디어 시스템 개발을 촉진할 수 있다.

2절에서는 H.264/SVC 구성과 개요 등을 간단히 요약하고, 단일 루프 복호(single loop decoding)나 드리프트 문제 등과 같은 H.264/SVC 표준 알고리즘의 주요 부호화 기법을 기술한다. SoC화를 위한 하드웨어 구조와 특징, C-모델 시뮬레이터 및 데이터 포맷과 시뮬레이터에 의한 영상복호의 결과 등을 3절에서 제시한다.

II. H.264/SVC 구성과 개요

MPEGx, H.264/AVC 등과 같은 스케일러블 하지 않은 비디오 부호화 알고리즘은 다음의 기본적인 전제조건을 가정한 것이다.

- 부호화기 측 즉 송신측에서 전송대역폭 및 채널특성과 같은 채널 환경의 완전한 사전 인지.
- 모든 종류의 복호기는 수신된 비트스트림으로부터 채널환경에 충분히 대응할 수 있는 정도의 실시간 복호가 가능함.

그러나 실제의 통신시스템에서 유무선 채널 용량이나 대역폭은 시간이나 정보전송량에 따라 가변되고, 디스플레이 기술의 진전에 힘입어 SXGA(1280 x 1024)급에서 QCIF(176 x 144)급 이하까지 다양한 공간 해상도와 데이터 처리능력을 갖는다. 또한 인터넷 비디오 스트리밍과 같이 통신환경 및 기기에 따른 전송 대역폭의 급격한 변동으로 기존의 스케일러블 하지 않은 비디오 부호화 알고리즘의 전제조건은 본질적으로 충족될 수 없다[6-8].

스케일러블 하지 않은 비디오 부호화 비트스트림은 전송채널에 따라 디지털 컷오프 즉 “graceful degradation”의 품질을 갖는 수신기 구현이 어려운 문제가 있다. [그림 1]은 전송채널의 비트율에 따른 영상 품질의 예를 나타낸 것으로 rate-distortion 곡선은 임의의 비트율에서 부호화에 의해 달성할 수 있는 이론적인 상한이다. 스케일러블 하지 않은 비디오 부호화기는 정해진 비트율에서 rate-distortion 곡선에 가장 근접한 최적의 부호화 효율을 얻는 것으로 채널 용량이 발생 비트율과 동일하면 최적의 결과를 얻을 수 있지만,

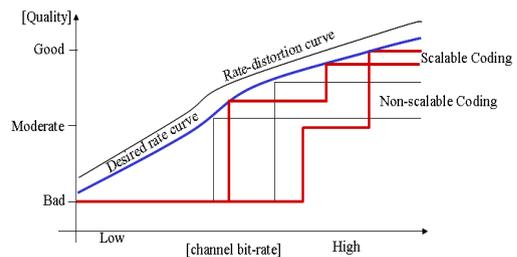


그림 1. 재구성한 비디오 화질에 대한 채널 비트율

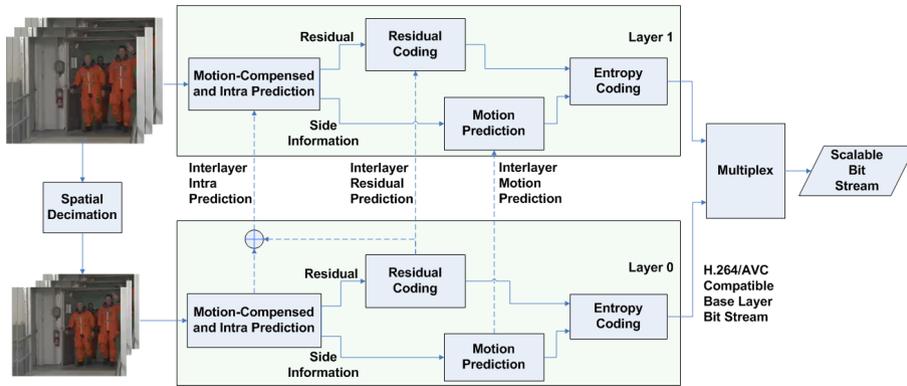
채널용량이 발생 비트율보다 적으면 정상적인 복호가 불가능하게 되어 회복 불가능한 결과를 초래하게 된다. 이에 반해 스케일러블 비디오 부호화기는 채널 환경 및 응용에 따라 적합한 비트스트림을 발생시켜 스케일러블 하지 않은 비디오 부호화기에서와 같은 파국적 결과를 피할 수 있다[6].

스케일러블 비디오에서 기본계층(Base Layer : BL)은 가장 낮은 화질과 해상도를 갖는 계층으로 H.264/AVC 베이스라인 프로파일 규격에 따라 부호화되고, 단일계층 복호기를 사용해서 복호하는 계층이고, 향상계층 (Enhancement Layer : EL)은 BL의 결과와 동일계층의 정보를 동시에 이용해서 보다 높은 화질과

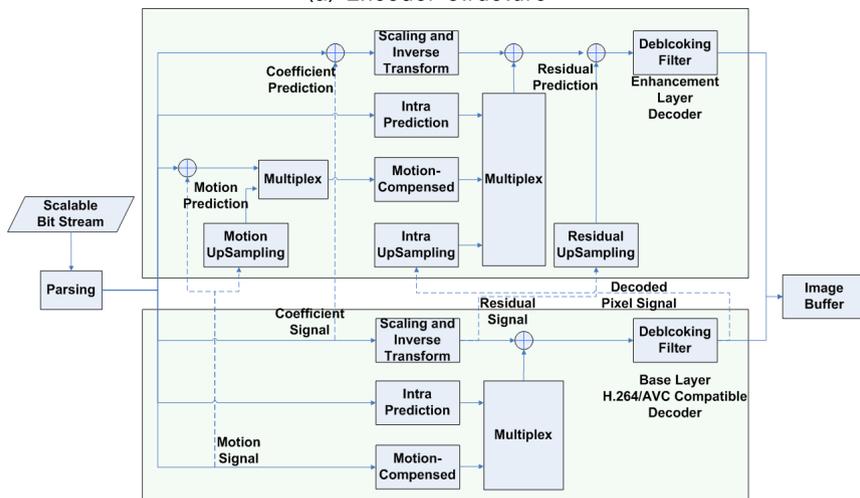
해상도를 갖는 영상이 복호되도록 하는 계층이다. H.264/SVC 규격에서는 하나의 BL과 복수개의 EL 영상을 수신할 수 있도록 시간, 공간, 화질 그리고 혼합 스케일러블을 지원한다. [그림 2](a)와 (b)는 2계층 구조의 H.264/SVC의 부호화기와 복호기 구조를 나타낸 것이다[9].

1. 주요 SVC 부호화 알고리즘[9]

H.264/SVC의 핵심기술은 복호 과정의 복잡도 증가 없이 부호화효율을 저하시키지 않으면서 스케일러블 기능을 지원하는 부호화 알고리즘이다. 영상정보의 중



(a) Encoder structure



(b) Decoder structure

그림 2. The structure of 2 layer H.264/SVC encoder and decoder structure

복성 제거를 통해 발생 정보 비트량을 줄이는 일반적인 방법으로 매크로블록 단위의 화면내 예측(intra-picture prediction)과 움직임보상에 의한 화면간 예측(motion compensated inter-picture prediction), 예측오차에 해당하는 잔류 신호(residual signal)의 DCT 변환과 양자화 등이 사용되고 있다.

H.264/SVC에서는 이상과 같은 기본적인 알고리즘 외에 계층간 정보의 중복성을 이용한 계층간 예측(inter-layer prediction) 방법을 도입해서 단일 루프 복호 구조에 의한 복호 과정의 복잡도의 증가문제를 해결하고 있다. 또한 기존의 스케일러블 부호화 알고리즘에서 부호화기와 복호기간 움직임 예측 루프의 동기가 일치하지 않아 발생하는 에러로써 시간에 따라 누적되고 영상품질에 심각한 결함을 야기하는 드리프트 문제를 "키 픽처" 개념을 도입해 해결한 것은 실용화를 촉진시키는 역할을 할 것이다. 계층간 예측은 복호과정으로 재구성된 보다 낮은 계층의 영상정보를 참조계층(reference layer)으로 사용해서 EL 계층의 데이터를 예측하는 방법이다.

H.264/SVC의 알고리즘의 핵심인 공간 스케일러블, 시간 스케일러블, 충실도 스케일러블 등에 관한 주요 핵심기술을 간단히 기술한다.

1.1 공간 스케일러블

다양한 공간해상도를 제공하기 위해 다중해상도(multiresolution) 기반의 계층화 부호화 기법을 기반으로 연속되는 계층간 예측 부호화 방법을 사용한다. 이때 각 계층의 비트스트림은 목표로 하는 영상의 공간해상도를 제공할 수 있다.

부호화 효율의 개선과 동시에 단일 루프 복호를 지원하기 위해 화면간 또는 동일 화면내 예측뿐만 아니라, 계층간 예측 알고리즘을 기본으로 채용하고 있다. 계층간 예측 방법의 기본은 EL의 rate-distortion 개선을 위해 하위계층 정보의 활용을 극대화 한 것이다. 이와 같은 계층간 예측은 계층간 움직임 정보 예측(inter-layer motion prediction), 계층간 잔류정보 예측(inter-layer residual prediction), 계층간 화면내예측(inter-layer intra prediction) 등 3종류가 있다.

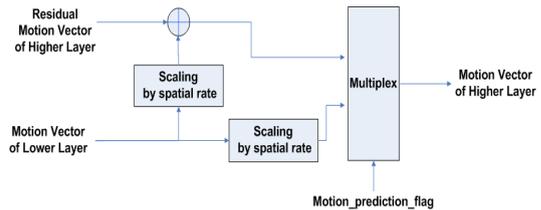


그림 3. Inter-layer motion prediction reconstruction procedure

■ Inter-layer motion prediction

계층간 움직임 정보 예측을 위해 H.264/SVC 규격에서는 H.264/AVC에 규정된 매크로블록 타입 외에 신택스 "base_mode_flag"로 제어되는 새로운 매크로블록 타입을 도입했다. 이 매크로블록 모드에서는 매크로블록 예측 및 영상 재구성에 필요한 매크로블록 타입과 모션 파라미터 등의 부가정보는 부호화하지 않고 잔류 신호만을 부호화해서 전송한다.

[그림 3]은 계층간 움직임 정보 예측 부호화에 대한 움직임 벡터의 재구성과정을 나타낸 것이다. 신택스 "motion_prediction_flag"에 따라 상위계층의 움직임 벡터 예측이 결정되고, 매크로블록 타입과 분할정보, 참조 영상 번호 등에 의해 움직임 보상 예측 과정이 수행된다.

■ Inter-layer residual prediction

하위계층의 동일 위치 블록이 화면간 예측으로 부호화 되었을 경우 계층간 잔류신호의 중복성을 제거해서 부호화 효율을 개선하기 위한 것으로 매크로블록의 예측 타입과 상관없이 모든 화면간 부호화된 매크로블록에 적용해서 잔류신호를 예측하는 과정이다. 이를 위해 신택스 "residual_prediction_flag"를 사용하고, 이 플래그의 값이 "1"이면 참조 계층내 대응하는 8x8 블록의 잔류신호를 DCT 변환 블록 단위를 중심으로 바이리니어 필터를 사용해서 업-샘플링 한다. [그림 4]는 계층간 잔여신호 예측으로 부호화된 상위계층의 잔류신호를 복호하는 과정을 나타낸 것으로, 하위계층의 잔류 신호 계수의 스케일링 및 IDCT 과정 후, 바이리니어 필터에 의한 업-샘플링과 상위계층의 차분 잔류신호와 합하여 상위계층의 잔류신호를 예측한다.

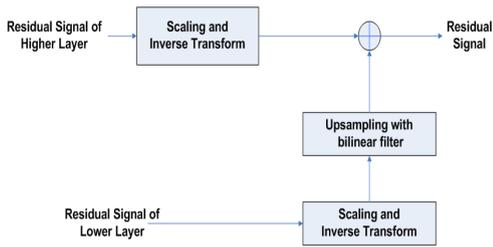


그림 4. Decoding procedure of inter-layer residual prediction

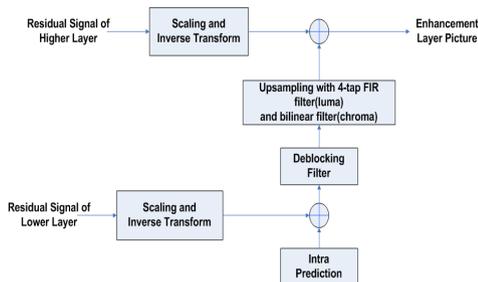


그림 5. Decoding procedure of inter-layer intra prediction

■ Inter-layer intra prediction

선택스 "base_mode_flag"가 "1"이고, 참조계층의 대응하는 8x8 블록이 화면내 예측으로 부호화된 경우, 하위계층의 복호된 영상정보를 그대로 활용하는 것으로 특히 "INTRA_BL" 모드라 한다. INTRA_BL 모드는 복잡도의 감소를 위하여 I-픽처에 대해서만 매크로 블록 단위로 수행된다. 하위계층의 영상은 디-블록킹 필터를 거친 후 업-샘플링 하며 예측 정보로 사용한다. 업-샘플링 과정에서 휘도성분은 4-탭 FIR 필터를, 색차성분은 바이리니어 필터를 사용한다. [그림 5]는 이상과 같은 계층간 화면내 예측으로 부호화된 비트스트림으로부터 영상을 복호하는 과정을 나타낸 것이다.

1.2 시간 스케일러블 과 SNR 스케일러블

시간 스케일러블은 H.264/AVC에서 정의된 양방향 예측 구조를 갖는 GOP단위의 B-픽처를 기반으로 프레임 레이트를 제어할 수 있게 한 것이다. 이전에 부호화된 모든 영상이 현재 부호화하려는 영상보다 화면 표시 순서상 앞에 위치하여 한 GOP 내의 다른 영상들의

부호화에 기본이 되는 영상을 "키 픽처(시간 스케일러블 레벨 T0)"라 부른다. "키 픽처"는 영상의 랜덤 액세스를 보장할 수 있도록 I-픽처로 부호화 하거나 순방향 예측을 사용하여 P-픽처로 부호화해서 GOP내의 다른 영상들보다 우선적으로 전송한다. GOP 내의 다른 영상은 참조영상으로 부터 계층적인 재구성을 수행함으로써 다양한 프레임 레이트를 가질 수 있다.

그러나 계층적인 B-픽처 구조는 양방향 예측 픽처의 사용으로 복호기의 항상계층 영상의 복호를 위해서는 GOP 내부의 모든 참조영상을 저장해야 한다. 따라서 GOP 사이즈가 커지게 되면 복호기의 복잡도 및 메모리 사용량이 방대하게 된다. 또한 영상 디스플레이 순서와 복호순서가 일치하지 않아 최대 GOP 단위시간의 디스플레이 지연이 발생한다. 따라서 B-픽처를 제외한 I-픽처와 P-픽처만을 사용해서 디스플레이 지연이 발생하지 않는 계층구조의 시간 스케일러블의 구성도 가능하다.[1]

H.264/SVC의 SNR 스케일러블은 CGS(Coarse Grain Scalability) 또는 MGS(Medium Grain Scalability)를 통하여 지원된다. CGS는 부호화 하려는 영상의 크기가 일정한 공간 스케일러블의 방식에 양자화 계수만을 변경하는 것을 말하며 MGS는 부호화하려는 영상과 하위계층의 양자화 계수를 제어해서 부호화하는 것을 말한다.

III. SoC화를 위한 복호기 구조와 C-모델 시뮬레이터

1. SoC화를 위한 H.264/SVC 복호기 구조

[그림 2](b)의 복호기 구조는 부호화 과정의 역순으로 복호하는 데이터 흐름 순서에 따라 복호 도구를 배치한 것으로 공개된 소프트웨어 코덱에서의 복호과정과 일치한다. 이와 같은 복호기는 기본적으로 프로세서의 계산능력, 내부 메모리나 버퍼, HDD 등 저장 공간의 한계에 대한 제한을 두지 않는다. 또한 메모리 액세스를 위한 제어나 주변 시스템과의 인터페이스가 완비된 것을 가정한다. 따라서 PC 환경을 기반으로 한 소프트

웨어에 의한 복호는 문제가 없으나, 다양한 응용에의 실시간처리나 계산능력에 한계가 있는 전용 프로세서로서 소프트웨어 시스템을 구현하는 경우에는 [그림 2](b)의 복호기 구조의 개선이 필요하다. 뿐만 아니라 진보된 DMB나 새로운 감시 시스템과 같은 응용 시스템의 통합 SoC를 개발하기 위해서는 하드웨어 구현에 적합한 새로운 복호기 구조가 요구된다.

H.264/SVC는 부호화 효율을 개선하기 위해 복잡한 구조의 움직임 벡터 추정과 움직임 보상 예측 방법을 적용하고 있다. 즉 다양한 움직임 보상 화면간 또는 계층간 예측구조, 1/4pel 단위의 움직임 벡터, 움직임 보상 예측을 위한 6-탭 움직임 보상 보간 필터, 계층간 움

직임 보상을 위한 바이리니어 필터, 복수개의 참조 픽처의 사용 등으로 인해 복호기의 계산량이 방대하다.

뿐만 아니라 프레임 내 예측 구조도 MPEGx 및 H.263 등과 달리 예측 모드 및 예측방법을 다양하게 해서 부호화 효율을 개선하고 있으나, 복호기에서의 메모리 사용 및 예측에 많은 연산량을 필요로 한다. 16x16, 8x8, 4x4 블록크기의 매크로블록의 분할과 9종류의 예측 방법을 처리해야 하고, 특히 계층간 프레임 내 예측에서는 4-탭 FIR 필터를 사용해서 업-샘플링의 보간을 수행한다.

이상과 같은 특성으로 인해 기존의 MPEGx 및 H.263 등에 비해 H.264/SVC의 복호기 복잡도는 매우 증가되

표 1. Decoding tools and I/O for each decoding engine

Name of Engine	Decoding Tools	I/O
Variable Length Decoding Engine (VLD)	<ul style="list-style-type: none"> o Syntax parsing o VLC decoding o Extensive error detection o Input from SRAM 	<ul style="list-style-type: none"> o Input from SRAM <ul style="list-style-type: none"> ▪ Encoded bit stream o Output <ul style="list-style-type: none"> ▪ Command for CP ▪ Transformed coefficient for CP&TDE
Coefficient Prediction and Transform Decoding Engine(CP&TDE)	<ul style="list-style-type: none"> o Coefficient Prediction(MGS) <ul style="list-style-type: none"> ▪ Residual difference coefficient prediction ▪ Storing scaled coefficient o Transformed coefficient scaling o Inverse quantization o Inverse transform(HDT and DCT) 	<ul style="list-style-type: none"> o Input from VLD <ul style="list-style-type: none"> ▪ Transformed coefficient ▪ Quantization parameter o Output <ul style="list-style-type: none"> ▪ residual sample for IP&PR ▪ residual sample of lower layer for IRP
Inter-Layer Residual Prediction (LRP)	<ul style="list-style-type: none"> o Up-sampling with bilinear filter o Accumulation residual sample 	<ul style="list-style-type: none"> o Input from CP&TDE <ul style="list-style-type: none"> ▪ Residual sample value of 2-layer o Output for IP&PR <ul style="list-style-type: none"> ▪ residual sample
Inter-Layer Motion Prediction and Motion Compensation (MP&MC)	<ul style="list-style-type: none"> o Inter-picture prediction <ul style="list-style-type: none"> ▪ 1/4 pel motion compensation o Inter-layer motion prediction <ul style="list-style-type: none"> ▪ Scaling by spatial rate ▪ Accumulation motion vector 	<ul style="list-style-type: none"> o Input from CP <ul style="list-style-type: none"> ▪ Motion vector of 2-layer ▪ Spatial rate ▪ Motion prediction mode ▪ Reference index ▪ Macroblock partition MbPartIdx o Output <ul style="list-style-type: none"> ▪ Inter prediction sample
Inter-Layer Intra/Intra Picture Prediction & Picture Reconstruct (IP&PR)	<ul style="list-style-type: none"> o Intra-picture prediction <ul style="list-style-type: none"> ▪ Intra-prediction mode direction ▪ Intra sample prediction o Inter-layer intra prediction <ul style="list-style-type: none"> ▪ Up-sampling by spatial rate ▪ 4-tap FIR filter for luma sample ▪ Bilinear filter for chroma sample o Picture reconstruction 	<ul style="list-style-type: none"> o Input <ul style="list-style-type: none"> ▪ Intra_NxN prediction mode and Macroblock index from CP ▪ Picture sample from DBF ▪ Spatial rate from CP ▪ Residual sample from CP&TDE or IRP ▪ Prediction sample from MP&MC o Output <ul style="list-style-type: none"> ▪ Reconstructed picture sample to SRAM
Core Processor (CP)	<ul style="list-style-type: none"> o Memory control o H/W engine interface control o Quantization parameter processing o Macroblock partitioning processing o Motion prediction mode processing o Intra-prediction modes processing o Spatial rate processing 	<ul style="list-style-type: none"> o Input <ul style="list-style-type: none"> ▪ Command from VLD o Output <ul style="list-style-type: none"> ▪ Control signal for all Engine

어 있다. 따라서 실시간성 및 응용 시스템에 적합한 통합된 SoC 칩을 개발하기 위해서는 새로운 하드웨어 구조 개발이 이루어져야 한다. [그림 6]은 제안한 H.264/SVC 복호기 구조를 나타낸 것이다. 제안한 복호기 구조의 특징은 다음과 같다.

- H.264/SVC 단일 루프 복호 특성을 기반으로 복호기 시스템 구성을 간략화 함.
- 응용 시스템에 최적인 복호영상 품질과 해상도를 지원할 수 있는 단일 계층 하드웨어 구조
- 플랫폼 기반의 시스템 설계 가능
- H.264/SVC 스케일러블 베이스라인 프로파일 지원 : H.264/AVC 베이스라인 프로파일을 기반으로 하고, EB슬라이스를 제외한 I/P/EI/EP 슬라이스와 4:2:0 비디오 포맷 및 픽처 구조의 영상 지원
- SVC의 다양한 응용 시스템과의 통합 SoC
- 핵심 프로세서를 내장한 임베디드 시스템
- 시스템 집적화에 유리한 구조
- 공간 스케일러블, IPPP 구조에 의한 디스플레이 지연이 없는 delay zero temporal scalability
- MGS와 CGS의 화질 스케일러블 지원

시스템 내부 버스는 내부 복호엔진과 외부 메모리간의 데이터 전송이나 내부 복호엔진 간의 복호 명령 및 데이터 전송을 처리한다. 각 복호엔진에서 처리되는 데

이터 파일 포맷은 파라미터 헤더 (8비트)와 제어 명령 데이터(8비트)의 16비트로 구성해서 복호된 Non-VCL NAL 파라미터 세트와 복호된 VCL NAL 유닛을 처리하고, 내부 복호엔진 간의 데이터 전송은 기술한 바와 같이 내부 시스템 버스를 사용해서 이루어진다. 각 복호엔진에서 처리되는 복호 도구와와 입출력 데이터를 정리해서 나타낸 것이 [표 2]이다.

2. C-모델 시뮬레이터

C-모델 시뮬레이터는 내부 복호엔진 블록간 제어 명령, 데이터의 효율적인 송수신을 위한 인터페이스의 검증과 회로설계에서 개발 IP의 동작 검증과 체계화된 시스템 개발을 지원하는 것을 목적으로 한다.

JSVM(Joint Scalable Video Model)은 JVT에서 표준화 알고리즘을 개발하기 위해 적용한 참조 프로그램으로 그 소스 코드가 공개되어 있다. [그림 7]은 JSVM ver. 9.14에서의 복호과정을 흐름도로 정리한 것으로 비디오 비트스트림에서 VCL NAL 유닛의 복호과정을 나타낸 것이다. JSVM의 복호기는 슬라이스 단위별로 macroblock_layer와 BL의 업-샘플링 등과 같이 입력 비트스트림에 전적으로 좌우되어 순차적으로 복호과정을 처리한다. 이와 같은 순차적인 슬라이스 단위 기반의 복호과정은 데이터 처리에 필요한 계산량을 증가시

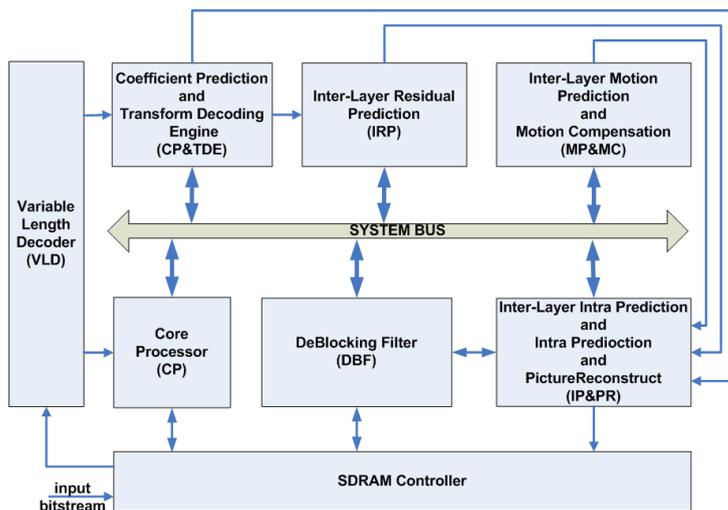


그림 6. The proposed architecture of H.264/SVC decoder

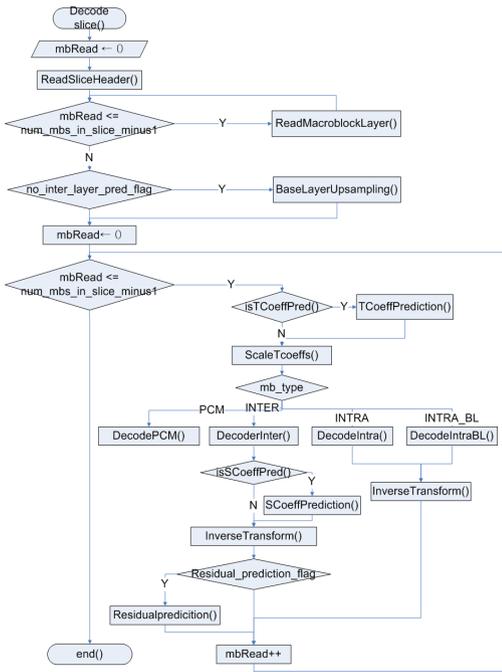


그림 7. Decoding flow of JSVM ver 9.14 open source program

켜 하드웨어에 의한 실시간 복호를 어렵게 한다. 또한 메모리를 무한대에 가깝게 사용할 수 있는 소프트웨어의 특성상 JSVM 복호기의 과도한 블록간 I/O 데이터 저장을 위한 메모리와 내부 레지스터의 사용은 하드웨어 구현 비용을 증가시키는 요인으로 작용한다. 또한 복호 엔진 각 블록설계에 필요한 제어명령 및 데이터를 발생시키지 않으므로 회로설계에 이용할 수 없는 문제가 있다.

C-모델에 의해 발생된 I/O 데이터를 사용하여 독립적으로 구현된 하드웨어의 세부 블록 성능을 검증함으로써 회로 구현의 유연성을 충분히 확보함과 동시에, 하드웨어의 개발을 촉진시키고 회로설계의 신뢰성 확보할 수 있다. 또한 각 복호엔진의 동시설계(parallel design)가 가능해 회로개발 시간을 단축시킬 수 있다.

[그림 8]은 [그림 6]의 복호기 구조를 기반으로 개발한 H.264/SVC C-모델 시뮬레이터의 흐름도를 나타낸 것이다. 본 연구에서 개발한 C-모델 시뮬레이터의 핵심 내용은 다음과 같다.

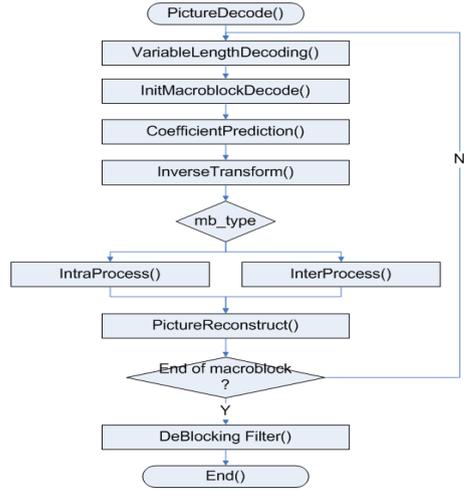


그림 8. Decoding flow for the proposed C-model simulator

- 슬라이스 단위의 복호 흐름을 매크로블록 단위의 복호 흐름으로 변경
- 각 하드웨어 엔진 블록의 복호 도구와 일치하는 형태의 함수 처리
- 하드웨어 엔진 블록간 I/O 관계와 일치하는 데이터 발생

[그림 9]의 흐름도에서 처리되는 주요 복호 알고리즘은 다음과 같다.

- VariableLengthDecoding() - VLD 블록과 일치하는 형태로 매크로블록 계층의 비트스트림을 복호해 CP, CP&TDE로 입력되는 데이터 발생
- InitMacroblockDecode() - CP 블록의 기능을 수행하여 VLD로부터 입력되는 제어 명령 데이터를 사용하여 각 엔진 구동에 필요한 제어 신호를 발생
- CoefficientPrediction() - CP&TDE 블록중 변환 계수의 예측과 스케일링의 기능을 수행하며 MGS를 위한 스케일된 계수를 발생
- InverseTransform() - CP&TDE 블록중 역변환의 기능을 수행하여 IP&PR, IRP로 입력되는 잔류신호 샘플 발생
- InterProcess() - IRP와 MP&MC 2개 블록의 기능과 일치하며 잔여신호 샘플의 업-샘플링과 계층간 움직임 예측으로 인한 움직임 보상을 수행하여

IP&PR 으로 입력되는 잔류신호 샘플과 화면간 예측 신호 샘플 발생

- IntraProcess() - IP&PR 블록 중 화면내 예측과 계층간 화면내 예측의 기능을 수행하며 화면내 예측 신호 샘플 발생
- PictureReconstruct() - IP&PR 블록중 픽처 재구성 기능을 수행하여 InverseTransform(), InterProcess(), IntraProcess()로부터의 잔류신호 샘플과 예측신호 샘플을 누산하여 재구성된 픽처 샘플

을 발생

3. SVC 복호기 구현 결과

H.264/SVC 복호기 C-모델 시뮬레이터의 성능을 검증하기 위해 JSVM 부호기를 사용하여 2계층의 SNR 스케일러블과 공간 스케일러블, 3종류의 시간 스케일러블 등의 계층을 갖는 혼합 스케일러블 비트스트림을 발생시켜 구현한 복호기의 메모리에 저장해서 사용했다. 시험표준 영상 CREW를 대상으로 최상위 공간 스케일

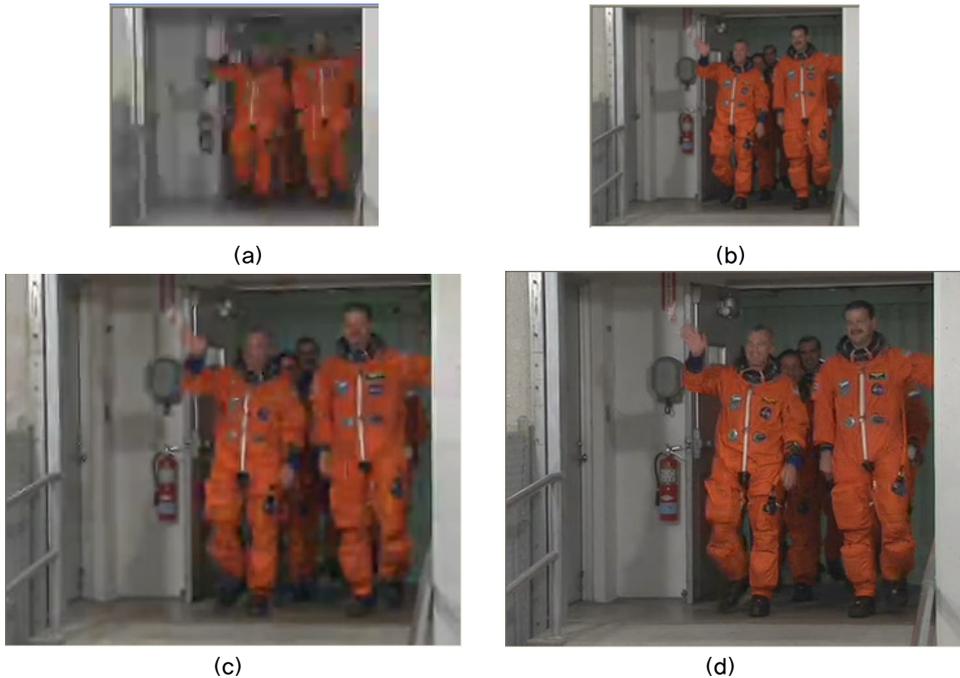


그림 9. Decoding results with the proposed decoding architecture and its implementation. (a) Spatial layer L0 and SNR layer Q0, (b) Spatial layer L0 and SNR layer Q1 (c) Spatial layer L1 and SNR layer Q0, (d) Spatial layer L1 and SNR layer Q1

표 2. Experimental results in terms of PSNR(db) and Bit rate(Kbit/s)

Scalability \ PSNR & Bitrates	PSNR_Y (db)	PSNR_U (db)	PSNR_V (db)	7.5fps (Kbps)	15fpS (kbps)	30fpS (kbps)
Spatial layer L0 and SNR layer Q0	28.63	36.11	33.49	11.1	15.5	19.8
Spatial layer L0 and SNR layer Q1	38.27	42.63	40.9	190.1	286.3	389.8
Spatial layer L1 and SNR layer Q0	30.93	39.94	37.03	208.5	315.9	433.8
Spatial layer L1 and SNR layer Q1	38.05	42.84	42.01	729.1	1121.3	1613.4

러블은 CIF(352x288)급, 최대 프레임 레이트는 30fps (maximum temporal scalability)를 처리할 수 있도록 했다.

SNR 스케일러블의 제어는 양자화계수(Qp) 값만을 조정해서 수행했다. SNR-계층 0 (Q0) 영상의 복호는 양자화계수 $Q_p=48$ 의 값을, SNR-계층 1(Q1)은 $Q_p=28$ 의 값을 고정해서 사용했다. 또한 공간 스케일러블은 기본계층(L0)으로 QCIF(176x144)의 공간해상도, 향상계층(L1)으로 CIF(352x288)의 공간해상도를 갖는 정수계수 공간비율(spatial rate)을 갖는 것으로 했다.

[그림 9]는 이들 각 계층의 결과를 나타내는 것으로 [그림 9](a)-(d)는 각각 Q0L0, Q1L0, Q0L1, Q1L1을 나타낸다. 이들 영상의 PSNR과 시간 스케일러블을 나타내는 프레임 레이트에 따른 비트율을 [표 2]에 제시한다. [표 2]는 시간, 공간, 화질계층에 따라 변화되는 PSNR과 전송 비트율을 나타낸 것으로, 특히 같은 시간 및 공간 계층에 대해 화질계층의 개선에 따라 발생 정보량이 2배 이상 증가하는 것을 알 수 있다[10].

이들 결과로부터 제안구조의 H.264/SVC 하드웨어 복호기는 공간, 시간 및 SNR 스케일러블의 혼합 스케일러블을 지원하고, 비교적 간단한 회로구성으로 정상적인 복호동작이 수행되는 것을 확인 했다. 또한 C-모델은 [그림 9]의 영상을 복호하는 과정에서 제안한 하드웨어 엔진 블록간 I/O 데이터와 일치하는 데이터를 발생하여 SoC 칩 설계에 효과적으로 사용될 수 있음을 확인 했다.

IV. 결론

본 논문에서는 최근 국제표준화가 완료된 H.264/SVC 복호기 구현을 위한 새로운 하드웨어 구조를 제안하고, 이를 기반으로 한 C-모델 시뮬레이터를 개발했다. 본 논문에서 취급한 H.264/SVC 규격은 스케일러블 베이스라인 프로파일과 IPPP 구조에 의한 비트스트림의 복호과정과 디스플레이 출력간에 시간지연을 없애고, 복호기의 복잡도를 크게 감소시킬 수 있다. 따라서 하드웨어 구조가 간단하므로 감시 시스템, 진보된

DMB 칩 등 다양한 분야에 응용할 수 있는 특징을 갖는다. 또한 핵심 복호 엔진과 핵심 프로세서로 처리가 가능해 기존의 임베디드 시스템으로 SVC 복호기를 구현할 수 있다.

하드웨어 구조 기반의 C-모델 시뮬레이터는 내부 복호 엔진 블록간 제어명령 및 데이터의 효율적인 송수신을 위한 인터페이스의 검증과 회로설계에서 개발된 IP의 동작 검증과 체계화된 시스템 설계를 지원함으로써 SoC 시스템 설계를 용이하게 한다. 이로 인해 다양한 응용 SoC 칩 설계 시간을 단축시킬 수 있다. 실제 동영상을 사용해서 제안된 하드웨어 구조와 C-모델 시뮬레이터의 동작을 검증했다. 지속적인 연구를 통해 H.264/SVC 복호기의 회로설계 및 향후 응용 시스템에 적합한 통합 SoC 칩 설계를 수행하는 것이 필요하다.

참고 문헌

- [1] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *IEEE Trans. on Circuits and Video Tech.*, Vol.17, No.9, pp.1103-1120, 2007.
- [2] Y. K. Wang, M. M. Hannuksela, "System and Transport Interface of SVC," *IEEE Trans. on Circuits and Video Tech.*, Vol.17, No.9, pp.1149-1163, 2007.
- [3] M. Wien, H. Schwarz, and T. Oelbaum, "Performance Analysis of SVC," *IEEE Trans. on Circuits and Video Tech.*, Vol.17, No.9, pp.1194-1203, 2007.
- [4] P. Amon, T. Rathgen, and D. Singer, "File Format for Scalable Video coding," *IEEE Trans. on Circuits and Video Tech.*, Vol.17, No.9, pp.1174-1185, 2007.
- [5] D. W. Song and C. W. Chen, "Scalable H.264/AVC Video Transmission Over MIMO Wireless System With Adaptive Channel

Selection Based on Partial Channel Information,"
IEEE Trans. on Circuits and Video Tech.,
Vol.17, No.9, pp.1218-1226, 2007.

- [6] L. Weiping, "Overview of Fine Granularity Scalability in MPEG-4 Video Standard," IEEE Trans. on Circuits and Video Tech., Vol.11, No.3, pp.301-317, 2001.
- [7] Miheala van der Schaar and H. Radha, "A Hybrid Temporal-SNR Fine-Granular Scalability for Internet Video," IEEE Trans. on Circuits and Video Tech., Vol.11, No.3, pp.318-331, 2001.
- [8] P. N. Josep, G. W. Cook, and E. J. Delp, "An Analysis of the Efficiency of Different SNR-Scalable Strategies for Video Coders," IEEE Transactions on Image Processing, Vol.15, No.4, pp.848-864, 2006.
- [9] Joint Draft ITU-T Rec. H.264/ISO/IEC-14496-10/Amd.3 Scalable Video Coding, JVT-X201.doc, 2007.
- [10] 정차근, 길대남, "C-모델 시뮬레이터 기반 H.264/SVC 복호기 시스템 구현", 한국콘텐츠학회 논문지, Vol.9, No.2, 2009(2).

저 자 소 개

정 차 근(Cha-Keon Cheong)

정회원



- 1984년 2월 : 서울대학교 전기공학(공학석사)
- 1993년 2월 : 일본 동경대학 전기공학과(공학박사)
- 1984년 1월 ~ 1997년 8월 : LG 종합기술원 책임연구원
- 1995년 4월 ~ 2006년 3월 : 일본 방송통신기구 초빙연구원
- 2002년 1월 ~ 2002년 4월 : 일본 동경대학 초빙교수
- 1997년 9월 ~ 현재 : 호서대학교 시스템 제어공학과 <관심분야> : 디지털 통신, 영상처리 및 부호화, 지능형차량제어, Machine Vision