

HEVC CABAC 복호화기의 역이진화기 설계

Hardware Implantation of De-Binarizer in HEVC CABAC Decoder

김 두 환*, 김 소 현*, 이 성 수*
Doohwan Kim*, Sohyun Kim*, Seongsoo Lee*

Abstract

HEVC CABAC encoder performs binary arithmetic encoding after syntax elements are converted into binary values. Therefore, in HEVC CABAC decoder, binarized syntax elements from binary arithmetic decoder should be de-binarized into original syntax elements in the de-binarizer. In this paper, a HEVC CABAC de-binarizer architecture was proposed and implemented. It consists of a controller that analyzes and merges binarized syntax elements and an engine that converts merged binarized syntax elements into original syntax elements. The designed de-binarizer was described in Verilog HDL and it was synthesized and verified in 0.18um process technology. Its gate count and maximum operating frequency are 3,114 gates and 220 MHz, respectively.

요 약

HEVC CABAC 부호화기에서는 이진 산술 부호화를 수행하기 전에 구문 요소를 이진 값으로 변환하는 과정이 선행된다. 따라서 HEVC CABAC 복호화기에서도 이진 산술 복호화기를 통해 이진 값으로 나타낸 구문 요소들을 원래의 값으로 역이진화 하는 역이진화기를 필요로 한다. 본 논문에서는 구문 요소의 종류를 파악하여 이진 값의 병합을 수행하는 제어기와, 제어기로부터 병합된 이진 값을 원래의 구문 요소로 변환시키는 엔진으로 구성된 역이진화기의 구조를 제안하고 이를 구현하였다. 설계된 역이진화기는 Verilog HDL로 기술하고 0.18um 공정에서 합성 및 검증하였으며, 하드웨어 크기는 3,114 게이트이고 최대 동작 속도는 220 MHz이다.

Key words: HEVC, CABAC, Decoder, De-Binarizer, Syntax Element

1. 서론

HEVC[1-6]는 CABAC(context-based adaptive binary arithmetic coding)이라는 문맥 기반의 적응적 이진 산술 부호화 방식을 통해 구문 요소(SE : syntax element)들을 비트열(bitstream)로

변환한다. CABAC 복호화기에서는 이 비트열을 읽어 원래의 구문 요소로 변환하는 역할을 수행하는데, 문맥 모델 선택기와 이진 산술 복호화기를 통해 이진 값으로 복호화된 구문 요소 값을 적절히 병합하여 원래의 값으로 변환시키는 것이 역이진화기(de-binarizer)의 역할이다. 각 이진 값

* School of Electronic Engineering, Soongsil University

★ Corresponding author (e-mail: sslee@ssu.ac.kr, tel: 02-820-0692)

※ Acknowledgment

“This research was supported by System IC Commercialization R&BD Program (10049498) funded by the Ministry of Trade, Industry & Energy.”

Manuscript received Sep. 19, 2016, revised Sep. 27, accepted Sep. 27, 2016

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

별로 어떤 종류의 구문 요소인지에 대한 정보가 포함될 수 있으나, 연속적으로 같은 종류의 구문 요소가 발생할 경우 이러한 구문 요소 값의 구분이 필요한 경우가 발생한다. 그렇기 때문에 이진 값의 병합은 구문 요소의 이진화 방식에 따라 나타날 수 있는 이진 값의 패턴을 고려하여 병합되어야 한다. 본 논문에서는 CABAC 복호화기에서 이러한 패턴을 고려하여 이진 값의 병합을 수행하는 제어기와, 제어기로부터 병합된 이진 값을 원래의 구문 요소로 변환시키는 엔진으로 구성된 역이진화기의 구조를 제안하고 이를 구현하였다.

II. 역이진화기 아키텍처

HEVC CABAC에서는 3가지 방식을 통해 이진화를 수행하는데 그 방식으로는 단일 이진화(single binarization), 결합 이진화(combined binarization), 맞춤 이진화(custom binarization)가 있다[2-6]. 단일 이진화는 FL(fixed length), TR(truncated Rice), EGk(k-th order exp-Golomb)의 3가지 알고리즘을 통해 부호화를 수행한다. 결합 이진화는 구문 요소를 접두부와 접미부로 나누어 위의 이진화 알고리즘 2가지를 결합하여 사용하는 방식이다. 맞춤 이진화는 각 구문 요소의 값에 따라 이미 정해져 있는 테이블의 값으로 변환을 수행한다. 역이진화기에서는 위의 방식으로 이진 변환된 값을 다시 구문 요소로 변환시키는 역할을 수행한다.

본 논문에서 제안하는 역이진화기는 그림 1과 같이 bin 병합 제어기(bin merge controller)와 역이진화 엔진(de-binarization engine)으로 나누어 설계하였다.

III. bin 병합 제어기 아키텍처

bin 병합 제어기에서는 각 구문 요소에 따라 이진화 방식을 분별한 후 각각의 이진화 방식에 맞춰 여러 개의 bin이 하나의 구문 요소가 될 때까지 병합한다. 기본적으로 이들 bin은 자신의 구문 요소가 어떤 종류인지와 역이진화를 위한 블록 정보(DB parameter)를 가지고 있어 연속적으로 등장하지 않는 구문 요소의 경우에는 종류의 구분만으로 간단히 병합이 가능하다. 하지만 같은

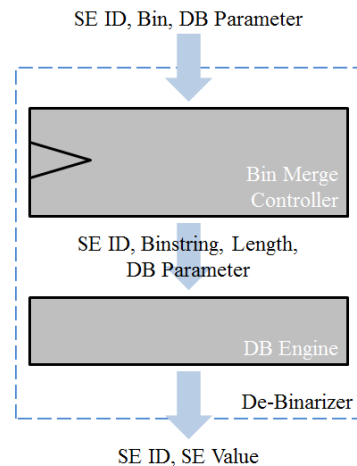


Fig. 1. Block diagram of HEVC CABAC Debinarizer

그림 1. HEVC CABAC 역이진화기의 블록도

Table 1. FL length

표 1. FL의 길이

cMax	Binstring Length
1	1
3	2
7	3
31	5

Table 2. EGk length

표 2. EGk의 길이

k	SE Value	Prefix Lenth	Suffix Length	Lenth
0	0	1	0	1
	1~2	2	1	3
	3~6	3	2	5
	7~14	4	3	7
1	0~1	1	1	2
	2~5	2	2	4
	6~13	3	3	6
	14~29	4	4	8
2	0~3	1	2	3
	4~11	2	3	5
	12~27	3	4	7

Table 3. EGk format

표 3. EGk의 포맷

k=0	k=1	k=2
0	0x	0xx
10x	10xx	10xxx
110x	110xxx	110xxxx
1110xxx	1110xxxx	1110xxxxx

구문 요소가 연속적으로 나타날 경우 구문 요소의 종류로 분류를 할 수 없기 때문 각 이진화 방식에 따른 패턴을 분석하여 bin열(binstring)의 시

작과 끝을 계산하여야 한다^[6].

1. 단일 이진화 방식

FL의 경우 *cMax* 값에 의해 표 1과 같이 bin열의 길이가 고정되기 때문에 그 길이만큼의 bin만 병합하면 된다.

TR의 경우 *cRiceParam* 값이 0일 경우 TU(truncated unary)로 동작하여 길이가 *cMax* 값과 동일할 경우 혹은 bin이 0일 경우 병합이 종료된다. *cRiceParam*이 1보다 클 경우 TU로 병합된 접두부에 접미부가 추가되며 접미부의 길이는 *cRiceParam*과 동일하다. 따라서 TR의 길이는 TU의 길이와 *cRiceParam*의 합으로 계산할 수 있다.

EGk의 경우 표 2와 같이 접두부의 길이와 접미부의 길이의 합으로 계산된다. 접미부의 길이는 접두부의 길이에 *k*값을 더한 후 1을 뺀 값과 동일하며 접두부는 bin 값이 0일 때까지의 길이로 계산될 수 있다. EGk의 포맷은 표 3과 같다.

2. 결합 이진화 방식

결합 이진화 방식은 먼저 접두부의 길이를 계산하고, 접미부의 존재 여부를 확인한 후 그 길이를 계산하여 그 합으로 전체 길이를 계산한다.

*cu_qp_delta_abs*의 경우 접두부는 TU로 접미부는 EGk로 계산하며, *coeff_abs_level_remaining*은 접두부는 TR로, 접미부는 EGk로 계산한다.

3. 맞춤 이진화 방식

맞춤 이진화 방식의 경우 DB Parameter에 따라 정해진 bin의 패턴이 있기에 그 패턴에 따라 길이가 계산된다.

IV. 역이진화 엔진 아키텍처

역이진화 엔진은 제어기를 통해 가공된 정보를 통해 구문 요소로 변환시킨다. 맞춤 이진화 방식의 경우 간단히 테이블 참조를 통해 이루어진다.

1. 단일 이진화 방식

FL의 경우 길이만큼의 bin열이 구문 요소 값이 된다. TU는 길이가 *cMax*값보다 작을 경우 길이에 1을 더한 값이 구문 요소 값이 되며 *cMax*와

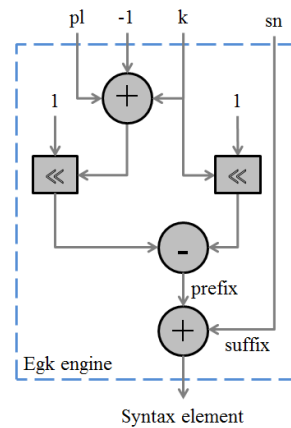


Fig. 2. Block diagram of EGk de-binarization engine
그림 2. EGk 역이진화 엔진의 블록도

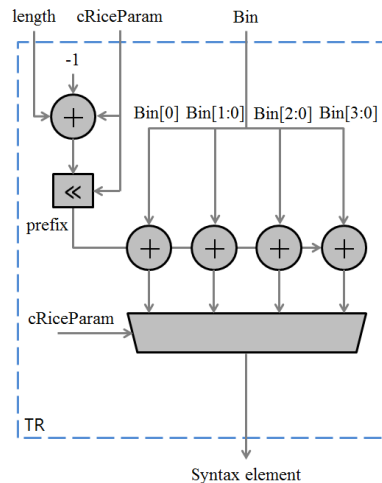


Fig. 3. Block diagram of TR de-binarization engine
그림 3. TR 역이진화 엔진의 블록도

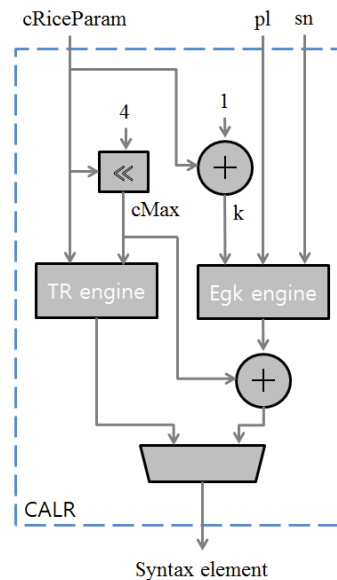


Fig. 4. Block diagram of CALR de-binarization engine
그림 4. CALR 역이진화 엔진의 블록도

Table 4. Test sequences

표 4. 테스트 영상

Sequence	Class A	Class B	Class C	Class D
Size	2560x 1600	1920x 1080	832x 480	416x 240
Frame rate	60	24	60	50
Frames	30			
Encoder configuration	low_delay, random access, all_intra			

Table 5. Synthesize result

표 5. 합성 결과

Technology	0.18um
Max Operating Frequency	220MHz
Max Throughput	220MBin/s
Gates	3114 gates

동일할 경우 길이가 구문 요소 값이 된다. EGk는 그림 2와 같이 접두부의 길이(pl), 접미부의 빈 열(sn)과 k값을 통해서 접두부와 접미부를 각각 계산 후 둘의 합으로 계산 가능하다.

2. 결합 이진화 방식

$cu_qp_delta_abs$ 는 값이 5보다 클 경우 접미부가 생성되기 때문에 접미부가 있을 때의 접두부 값은 항상 5로 고정된다. 따라서 EGk의 값에 5를 더하여 구문 요소의 값을 계산할 수 있다.

$coeff_abs_level_remaining$ (CALR)은 접두부가 $cMax$ 보다 클 경우 접미부가 생성된다. 따라서 이 경우는 EGk의 값과 $cMax$ 값의 합으로 나타낼 수 있다. $cMax$ 의 값보다 작을 경우 접두부만 존재하기 때문에 TR의 값이 구문 요소가 된다. 그림 3에 TR 역이진화 엔진의 구조를 나타내었으며 그림 4에 CALR 역이진화 엔진의 구조를 나타내었다.

V. 구현 결과 및 결론

본 논문에서는 제안된 역이진화기 아키텍처를 검증하기 위해 HEVC 테스트 모델인 HM11[7]로 시뮬레이션을 수행하였다. 표 4와 같이 4개의 클래스에 대해 각각 3개의 부호화 구성을 적용하여 총 12개의 테스트 영상을 사용하였으며, 이들 모두에 대해 정상적으로 동작함을 확인하였다.

또한 제안된 역이진화기 아키텍처를 Verilog HDL로 설계한 후, 0.18um 공정에서 합성하였다. 구현된 하드웨어는 3114개의 게이트로 이루어져 있으며, 최대 동작속도는 220MHz, 최대 처리율은 220Mbin/s이다.

본 논문에서 제안한 역이진화기 하드웨어는 HEVC CABAC 복호화기의 핵심 블록으로, 크기가 작고 동작 속도가 빨라서 다양한 HEVC 복호화기에 널리 응용될 수 있다.

References

- [1] G. Sullivan, J. Ohm, W. Han, T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol.* vol. 22, no. 15, pp. 1649-1668, Sep. 2012.
- [2] D. Pham, J. Moon, and S. Lee, "Hardware Implementation of HEVC CABAC Binarizer," *j.inst.Korean.electr.electron.eng.*, vol. 18, no. 3, pp. 356-361, Sep. 2014.
- [3] D. Pham, J. Moon, D. Kim, and S. Lee, "Hardware Implementation of HEVC CABAC Binary Arithmetic Encoder," *j.inst.Korean.electr.electron.eng.*, vol. 18, no. 4, pp. 630-635, Dec. 2014.
- [4] D. Kim, J. Moon, and S. Lee, "Hardware Implementation of HEVC CABAC Context Modeler," *j.inst.Korean.electr.electron.eng.*, vol. 19, no. 2, pp. 254-259, Jun. 2015.
- [5] D. Kim, J. Moon, and S. Lee, "Design of HEVC CABAC Encoder with Parallel Processing of Bypass Bins," *j.inst.Korean.electr.electron.eng.*, vol. 19, no. 4, pp. 583-589, Dec. 2015.
- [6] D. Kim, S. Kim, and S. Lee, "Hardware Implementation of HEVC CABAC Decoder De-Binarizer," in *Proc. of IKEEE Summer Conference*, 2016, pp. 1-4.
- [7] JVT of ISO/IEC MPEG and ITU-T VCEG, "HM Reference Software," https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware