

논문 2010-47SD-1-4

H.264/AVC를 위한 디블록킹 필터의 최적화된 하드웨어 설계

(Optimized Hardware Design of Deblocking Filter for H.264/AVC)

정 윤 진*, 류 광 기**

(Younjin Jung and Kwangki Ryoo)

요 약

본 논문에서는 고성능 H.264/AVC 복호기 설계를 위해 디블록킹 필터의 수행시간 단축과 저전력 설계를 위한 필터링 순서 및 효율적인 메모리 구조를 제안하고 5단 파이프라인으로 구성된 필터의 설계에 대해 기술한다. 디블록킹 필터는 블록 경계에서 발생하는 왜곡을 제거하여 영상의 화질을 개선시키지만 하나의 경계에 여러 번 필터링을 수행하여 많은 메모리 접근과 반복되는 연산과정이 수반된다. 따라서 본 논문에서는 메모리 접근과 필터 수행 사이클을 최소화하는 새로운 필터 순서를 제안하고 반복되는 연산의 효율적 관리를 위해 파이프라인 구조를 적용하였다. 제안하는 디블록킹 필터는 메모리 읽기, 임계값 계산, 전처리 연산, 필터 연산, 메모리 쓰기로 구성된 5단 파이프라인으로 구현되어 순차적인 필터 연산에 병렬적 처리가 가능하며 각 단계에 클럭 게이팅을 적용하여 하드웨어 자원에 불필요한 전력을 감소시켰다. 또한, 적은 내부 트랜스포지션 버퍼를 사용하면서 필터링 순서를 효율적으로 개선하여 필터 수행을 위한 메모리 접근과 수행 사이클을 감소시켰다.

제안하는 디블록킹 필터의 하드웨어는 Verilog HDL로 설계 하였으며 기존의 복호기에 통합하여 Modelsim 6.2g 시뮬레이터를 이용해 검증하였다. 입력으로는 표준 참조 소프트웨어 JM9.4 부호기를 통해 압축한 다양한 QCIF영상 샘플을 사용하였다. 기존 필터들과 수행 사이클을 비교한 결과, 제안하는 구조의 설계가 비교적 적은 트랜스포지션 버퍼를 사용했으며 최소 20%의 수행 사이클이 감소함을 확인하였다.

Abstract

This paper describes a design of 5-stage pipelined de-blocking filter with power reduction scheme and proposes a efficient memory architecture and filter order for high performance H.264/AVC Decoder. Generally the de-blocking filter removes block boundary artifacts and enhances image quality. Nevertheless filter has a few disadvantage that it requires a number of memory access and iterated operations because of filter operation for 4 time to one edge. So this paper proposes a optimized filter ordering and efficient hardware architecture for the reduction of memory access and total filter cycles. In proposed filter parallel processing is available because of structured 5-stage pipeline consisted of memory read, threshold decider, pre-calculation, filter operation and write back. Also it can reduce power consumption because it uses a clock gating scheme which disable unnecessary clock switching. Besides total number of filtering cycle is decreased by new filter order.

The proposed filter is designed with Verilog-HDL and functionally verified with the whole H.264/AVC decoder using the Modelsim 6.2g simulator. Input vectors are QCIF images generated by JM9.4 standard encoder software. As a result of experiment, it shows that the filter can make about 20% total filter cycles reduction and it requires small transposition buffer size.

Keywords : 디블록킹 필터, H.264/AVC, 파이프라인, 클럭 게이팅, 트랜스포지션 버퍼

* 학생회원, ** 정회원, 한밭대학교 정보통신공학과
(Department of Information and Communication
Engineering, Hanbat National University)

※ 본 연구는 IDEC의 CAD Tool지원, 중소기업청의 산학협력실 지원사업 및 ETRI 시스템 반도체 산업진흥센터의 IT SoC 핵심설계인력 양성 사업의 연구결과임.

접수일자: 2009년11월20일, 수정완료일: 2009년12월15일

I. 서 론

H.264/AVC는 비디오 코딩 전문가 그룹 ITU-T와 동영상 전문가 그룹 MPEG이 함께 구성한 JVT에 의해 제안된 동영상 압축 기술로 모션 JPEG 포맷과

MPEG-4 Part 2 표준보다 더 높은 압축률과 개선된 품질을 제공함으로써 세계적인 차세대 동영상 압축 기술로 평가받고 있으며 디지털 TV, 위성지상파 DMB, 차세대 DVD 등 다채널 고품질의 영상 압축이 필요한 다양한 멀티미디어 서비스에 활용되고 있다.

H.264/AVC는 비디오 압축 성능을 높이기 위해 기존의 비디오 부호화 표준에 비해 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. 또한, 영상의 화질을 높이기 위해 부호화 과정 중 디블록킹 필터를 추가하고 정수 기반의 DCT 사용으로 기존 부호화기와 복호기 사이의 DCT 계수 불일치 문제를 해결하였다^[1].

디블록킹 필터는 블록 경계에서 발생하는 왜곡을 제거하는 역할을 하며 기존 영상부호화 방식에서 발생하는 영상 화질의 열화 방지를 위해 블록 경계를 기준으로 좌우, 상하에 위치한 픽셀들의 유사성, 경계 강도 및 양자화 계수를 고려하여 왜곡의 발생정도에 따라 적응적으로 수행된다. 하지만, 하나의 경계에 대해 여러 번의 필터링이 수행되어 메모리 접근이 많고, 복호기 연산량의 30~50%를 차지하는 단점이 있다.

본 논문에서는 고성능 H.264 복호기 설계를 위해 디블록킹 필터의 수행시간 단축과 저전력 설계를 위한 필터링 순서 및 효율적인 메모리 구조를 제안하고 5단 파이프라인으로 구성된 필터의 설계에 대해 기술한다.

II. 디블록킹 필터 알고리즘

1. 디블록킹 필터의 동작

H.264/AVC 복호기 내부의 디블록킹 필터는 그림 1과 같이 하나의 매크로블록이 복원되어 디스플레이 또는 참조 메모리에 저장되기 전 단계에서 수행된다. 4x4 블록 단위의 블록 경계에 대해 휘도와 색차 성분을 분리하여 적용되며 프레임의 왼쪽 위부터 시작하여 매크

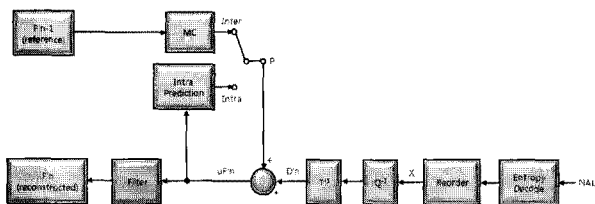


그림 1. H.264/AVC 복호기 구조
Fig. 1. Architecture of H.264/AVC decoder.

로블록의 번호가 증가하는 순서로 처리된다.

2. 경계강도 설정 및 필터링 순서

디블록킹 필터는 블록 경계의 세기인 경계 강도에 따라 경계강도가 4일 경우 강한 필터, 1에서 3일 경우 일반 필터를 선택 적용하며 0인 경우 필터 적용을 하지 않는다. 표 1은 블록의 경계강도 결정 조건 및 경계부에서 디블록킹 필터가 적용되는 화소들을 나타낸다. 필터링 할 경계를 기준으로 좌우, 상하 각각 4개의 블록을 참조하여 p, q라 하였을 때 양쪽 블록에 대해 경계강도를 적용한다.

디블록킹 필터는 매크로블록의 수직 경계면을 왼쪽에서 오른쪽으로 필터링한 후, 수평 경계면을 위에서 아래로 필터링 하는 순서로 진행된다. 그림 2는 표준에 제시된 필터링 순서를 나타낸다.

표 1. 블록 경계강도 정의
Table 1. Definition of block boundary strength.

경계강도	조건
4	p 또는 q블록 중 적어도 하나의 블록이 인트라 예측 부호화 매크로 블록에 속하고 매크로 블록 경계에 위치한다.
3	p 또는 q블록 중 적어도 한쪽이 인트라 예측 부호화 매크로 블록에 속하고 매크로 블록 경계에 위치하지 않는다.
2	p와 q의 블록 중 어느 한 쪽도 인트라 예측 부호화 매크로 블록에 속하지 않고 어느 한쪽이 직교변환계수를 가진다.
1	p와 q의 블록 중 어느 한쪽도 인트라 예측 부호화 매크로 블록에 속하지 않고 어느 쪽도 직교변환계수를 갖지 않으며, 참조프레임이 다르거나 참조프레임의 매수가 다르거나 움직임 벡터 값이 다르다.
0	위의 어느 경우에도 포함되지 않는다.

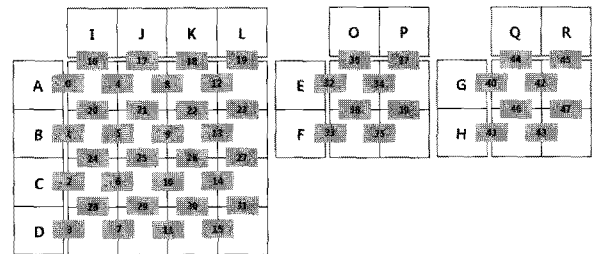


그림 2. 표준의 필터링 순서
Fig. 2. Filter orders of standard.

3. 디블록킹 필터 연산

경계강도가 0이 아닌 경우에 필터링 온/오프 결정을 위해 식(1)의 조건식을 이용하며 조건을 만족할 경우 디블록킹 필터가 적용된다.

$$Bs \neq 0 \ \&\& \ |p_0 - q_0| < a \ \&\& \ |p_1 - p_0| < \beta \ \&\& \ |q_1 - q_0| < \beta \tag{1}$$

위 조건을 만족하는 경우 중 경계강도가 1부터 3인 경우에 식(2)의 연산으로서 필터링이 이루어지며 델타는 화소 값을 조절하는 값으로서 식(3)으로 구한다.

$$p0' = \text{clip1}(p0 + \Delta), q0' = \text{clip1}(q0 + \Delta) \quad (2)$$

$$\Delta = \text{clip3}(-tc, tc, (((q0 - p0) \ll 2 + (p1 - q1) + 4) \gg 3)) \quad (3)$$

$$\alpha p < \beta \ \&\& \ |p0 - q0| < ((\alpha \gg 2) + 2) \quad (4)$$

$$p0' = (p2 + 2 * p1 + 2 * p0 + 2 * q0 + q1 + 4) \gg 3$$

$$p1' = (p2 + p1 + p0 + q0 + 2) \gg 2$$

$$p2' = (2 * p3 + 3 * p2 + p1 + p0 + q0 + 4) \gg 3 \quad (5)$$

$$p0' = (2 * p1 + p0 + q1 + 2) \gg 2, p1' = p1, p2' = p \quad (6)$$

강한 필터링이 필요한 경계강도가 4인 경우에는 4탭, 5탭 필터가 적용되며 식(4)의 조건을 만족할 경우에는 식(5), 그렇지 않으면 식(6)이 적용된다^[2~3].

III. 제안하는 더블록킹 필터

1. 하드웨어 구조

제안하는 더블록킹 필터의 하드웨어 구성은 그림 3과 같이 필터링에 사용될 픽셀 및 필터링 연산 중간결과가 저장될 메모리 블록, 필터링 연산을 담당하는 파이프라인으로 구성되며 필터링 완료된 픽셀은 외부 디스플레이 메모리 또는 참조 메모리로 출력된다.

제안하는 필터의 동작은 그림 4와 같이 메모리 읽기, 임계값 계산, 전처리 연산, 필터 연산, 메모리 쓰기로 구성된 5단 파이프라인으로 수행된다. 파이프라인의 각 단계마다 추가적인 컨트롤 신호를 사용하여 클록 게이팅을 적용함으로써 전력소모를 감소시켰다.

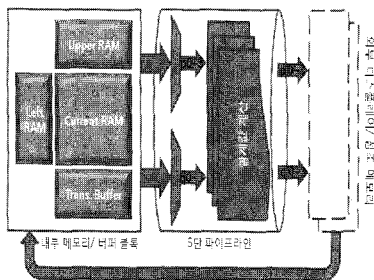


그림 3. 제안하는 하드웨어 구조
Fig. 3. Proposed hardware architecture.

가. 파이프라인 구조

연속적인 필터링 과정에서 두 개의 필터링 동작 사이

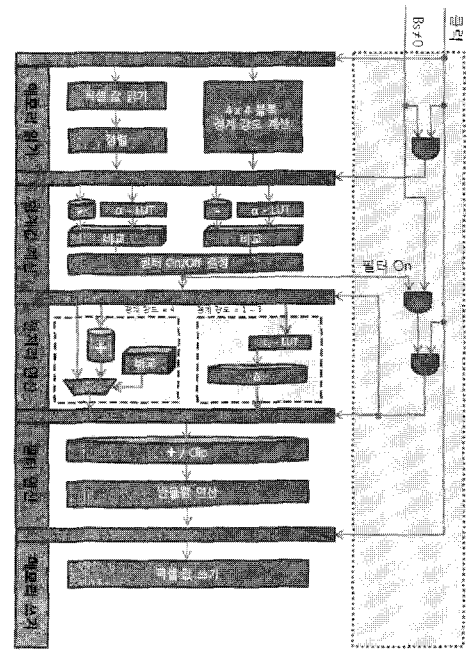


그림 4. 제안하는 파이프라인 구조
Fig. 4. Proposed pipeline architecture.

의 데이터 의존성이 없다면 파이프라인 구조가 적용될 수 있다. 따라서 본 논문에서는 H.264/AVC 더블록킹 필터의 처리를 향상 위해 필터 전체 연산 플로우와 테스크를 분석하고 최적화된 5단 파이프라인 구조를 제안한다. 파이프라인의 첫 번째 단계인 메모리 읽기 단계에서는 메모리로부터 필터링을 위한 픽셀데이터를 읽어 할당하는 것은 물론 현재 매크로블록의 경계 강도로부터 필터링할 4x4블록의 경계 강도 값을 얻는다. 만약 현재의 Bs 값이 0이라면 이어지는 파이프라인 단계는 모두 NOP (No-Operation) 모드로 채워진다.

임계값 계산 단계는 필터링 온오프에 필요한 식(1)을 검사하기 위해 룩업테이블을 참조하여 α, β 값을 결정한다. α, β 값은 초기 설정 값으로 양자화 파라미터 QP 값에 따라 정해지며 슬라이스 헤더에 포함된다. 이 두 개의 파라미터에 의해 사용자는 필터 강도를 선택적으로 조절할 수 있다. 전처리 연산 단계는 경계 강도가 4일 경우, 필터 연산이 길어 하나의 단계에서 수행될 수 없으므로 식(4)의 결과에 따라 필터링 수식의 일부를 미리 계산하며 경계 강도가 1에서 3일 경우, C1 룩업테이블을 참조하여 필터 연산에 필요한 Δ 값을 결정한다.

필터 연산 단계에서는 계산된 임계값과 경계 강도에 따라 입력된 픽셀들을 각기 다른 탭의 필터를 적용하여 연산하며 완료된 데이터는 메모리 쓰기 단계에서 다음 필터링에 사용하기 위해 내부 버퍼에 저장, 트랜스포트

본 논문에서는 효율적인 더블록킹 필터의 파이프라인 구조를 사용하고 해저드를 제거하기 위해 새로운 필터링 순서와 효율적인 메모리 구조를 제한한다. 제안하는 필터링 순서는 표준에서처럼 매크로블록 단위로 모든 수직경계에 대해 필터링을 수행한 후, 수평 경계에 대해 필터링 하는 것이 아니라 그림 6과 같이 하나의 매크로블록을 8x8블록으로 나누어 처리하는 방식을 사용하였다.

제안하는 순서는 4x4블록에 대해 표준과 같이 왼쪽 경계면을 처음, 아래쪽 경계면을 마지막에 처리함으로써 동일한 필터링 결과를 얻을 수 있으며 내부의 트랜스포지션 버퍼를 이용하여 연산 중간의 데이터를 저장함으로써 하나의 4x4블록이 가진 4개의 모든 경계면에 대해 가능한 처리를 완료한 뒤 메모리에 저장한다. 따라서 매크로블록의 가장 왼쪽 경계면과 위쪽 경계면의 필터링에 필요한 픽셀이 저장된 좌측, 상단 매크로블록과 현재 매크로블록에 대한 픽셀 값이 저장될 3개의 메모리, 수직 경계 필터링 이후 중간 데이터 값의 저장을 위한 7개의 내부 버퍼가 요구된다. 좌측 메모리는 하나의 매크로블록의 필터링 동안 총 8번의 접근이 수행되며 상단 메모리는 총 8번 이외에 추가적으로 5, 9, 13, 15번 경계의 필터링 후 중간 데이터를 저장할 경우에도 접근된다. 내부 버퍼는 트랜스포지션 버퍼로서 그

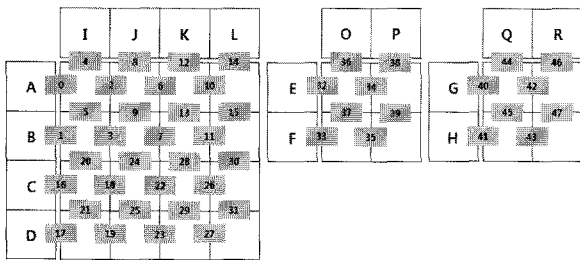


그림 6. 제안하는 필터링 순서
Fig. 6. Proposed filtering orders.

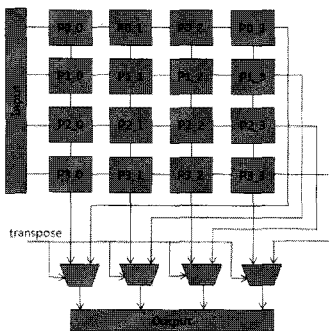


그림 7. 트랜스포지션 버퍼 구조
Fig. 7. Transposition buffer architecture.

표 2. 사용된 메모리 및 버퍼 크기
Table 2. Occupied memory and buffer size.

메모리 종류	크기		
	휘도	색차	전체
좌측 메모리	512	512	1024
상단 메모리	32N	32N	64N
현재 메모리	2048	1024	3072
트랜스포지션 버퍼	7 x (32 x 4) = 896		

림 7과 같이 단순히 데이터 값을 저장하는 기능뿐만 아니라 수직 경계에 대한 필터링 이후, 수평 경계 필터링에 사용될 데이터 형태로의 변환을 담당한다.

더블록킹 필터에서는 하나의 매크로블록의 필터링을 완료하였을 때, 트랜스포지션이 필요하며 이를 디스플레이 또는 참조 메모리로의 저장을 위해 추가로 클럭이 소요된다. 제안하는 구조에서는 이를 제거하기 위해 다음 매크로블록의 필터링 수행과 메모리 쓰기 동작을 병행함으로써 프레임 마지막 매크로블록을 제외한 나머지 블록에서 192 클럭이 소요된다. 사용된 메모리와 내부 버퍼 사이즈는 표 2와 같으며 여기서 N은 프레임 개수를 의미한다.

IV. 더블록킹 필터의 검증

1. 검증 환경

본 논문에서 제안하는 더블록킹 필터는 Verilog HDL로 설계되었으며 완전한 검증을 위해 이미 설계된 복호기에 통합하여 검증 환경을 구축하였으며 통합된 복호기는 그림 8과 같다.

제안하는 더블록킹 필터는 일반적인 H.264/AVC 복호기에서와 같이 복호기의 가장 마지막 단계로서, 예측 모듈의 출력과 변환/양자화 모듈의 출력 합으로 구해지는 화소 값을 재구성하여 입력으로 받는다. 또한 더블록킹 필터의 출력은 참조영상을 위한 메모리 또는 디스

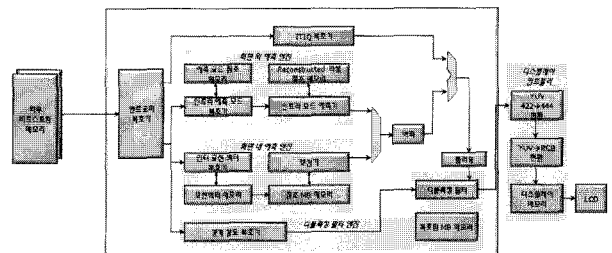


그림 8. 제안하는 더블록킹 필터를 통합한 복호기 구조
Fig. 8. Integrated decoder architecture with proposed deblocking filter.

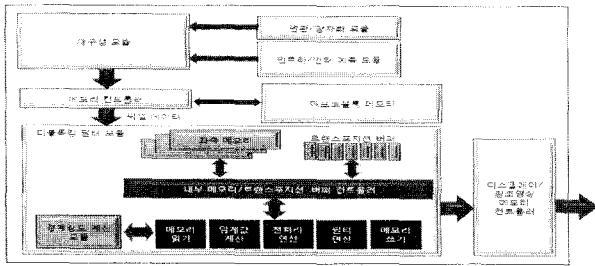


그림 9. 디블로킹 필터검증을 위한 주변 블록
Fig. 9. Neighbor block for verification of deblocking filter.

플레이 위한 메모리로 출력된다. 검증에 필요한 주변 메모리 블록과 필터의 입력이 되는 이전 단계의 블록 구성 및 흐름이 그림 9에 나타난다.

2. 검증 과정

통합한 복호기를 통해 디블로킹 필터의 동작을 검증하기 위해 표준 참조 소프트웨어 JM9.4^[6]를 이용하여 다양한 영상을 베이스라인 프로파일로 부호화하였다. 입력 영상은 모두 QCIF 해상도로 크기는 144x176이며 그림 10은 제안한 디블로킹 필터의 검증에 사용한 영상 샘플을 보여준다.

영상은 부호화를 거치면 이진파일로 출력되며, 출력 파일은 시뮬레이션 검증을 위한 입력 벡터로 사용하기 위해 소프트웨어 프로그램을 통해 hex파일로 변환하였다. 이를 테스트벤치에서 Verilog HDL 문법 중 하나인 \$readmemb 함수를 통해 복호기의 비트스트림 메모리에 초기화하여 복호기의 입력으로 사용한다. 복호 이후, 출력은 디스플레이용 메모리와 참조영상용 메모리에 저장하지만 입력 영상과 같은 형태로 가시화하기 위해 텍스트 포맷으로 저장하고 소프트웨어 프로그램을 통해 다시 이진파일 형태로 변환한다.

변환된 파일은 yuv 뷰어를 통해 영상을 검증하였으



그림 10. 검증에 사용된 영상 샘플
Fig. 10. Image samples for deblocking filter verification.

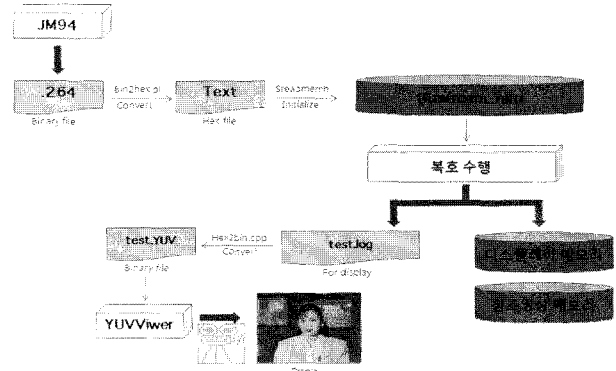


그림 11. 디블로킹 필터 검증 과정
Fig. 11. Verification flow of deblocking filter.

며 기존 필터와 제안한 필터의 결과를 비교하여 필터의 동작을 검증하였다. 이 과정은 그림 11에 나타난다.

3. 성능 비교

검증시 소요되는 디블로킹 필터의 수행 사이클을 기존의 필터들과 비교하여 표 3에 나타내었다. 제안한 구조는 4x4블록의 하나의 경계를 필터링하는데 오직 한 사이클이 소요되어 매크로블록당 192사이클이 소요된다. 제안한 구조와 동일하게 파이프라인 구조화되고 하이브리드 필터링 순서를 사용한 [10]과 비교할 때, 사이클 수는 동일하지만 트랜스포지션 버퍼수가 상대적으로 적은 장점을 가지고 있다. 제안한 디블로킹 필터를 Chartered 0.18um CMOS공정을 이용해 합성한 결과 최대 180MHz에서 동작 가능하며 13K개의 게이트 수를 갖는다. 0.18um 공정을 사용한 합성 결과 [9]에 비해 적은 게이트 수를 가지며 0.13um 공정의 [10]과는 절대비교가 어렵지만 셀 크기가 약2~3배 차이를 갖는다고 가정할 때, 더 적은 게이트 수를 갖을 것으로 예상된다.

그림 12는 파이프라인 구조, 트랜스포지션 버퍼, 하이브리드 필터링 순서 사용에 따른 수행 사이클 감소 및 제안하는 구조의 수행 사이클 비교 결과를 나타낸

표 3. 디블로킹 필터의 수행 성능 비교
Table 3. Comparison for filter processing cycles.

	[7]	[8]	[9]	[10]	제안
파이프라인	미적용	미적용	미적용	적용	적용
필터링순서	하이브리드	순차적	하이브리드	하이브리드	하이브리드
트랜스포지션버퍼	32x92	32x16	32x16	32x40	32x28
수행사이클	6144	614	243	192	192
공정	FPGA	0.25um	0.18um	0.13um	0.18um
게이트 수	NA	20.66K	21.2K	7.5K	13K
최대동작주파수	72MHz	100MHz	100MHz	200MHz	180MHz

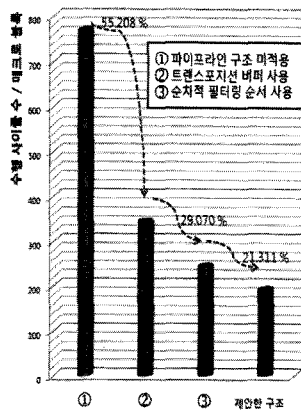


그림 12. 적용 기술에 따른 사이클 감소
Fig. 12. Cycle reduction according to proposed architecture.

다. 제안한 더블록킹 필터는 파이프라인 구조화를 통해 파이프라인을 사용하지 않은 필터에 비해 55%의 수행 사이클을 감소시켰으며, 트랜스포지션 버퍼 사용 및 순차적 필터링 순서를 적용하여 각각 29%, 21%의 사이클을 감소시켜 하나의 매크로블록마다 192사이클이 소요된다.

0.18um 공정을 사용한 [11]의 경우, 하나의 매크로블록을 처리하는데 2.39uW의 전력을 소모하는 반면, 제안하는 더블록킹 필터는 2.05uW의 전력을 소모하는 것으로 나타났다. 이는 클럭게이팅 적용 및 메모리 접근 횟수의 감소로 인해 전력소모가 7% 감소된 결과이다.

V. 결론

본 논문에서는 고성능 H.264/AVC 복호기 설계를 위해 더블록킹 필터의 수행시간 단축과 저전력 설계를 위한 필터링 순서 및 효율적인 메모리 구조를 제안하고 클럭게이팅을 이용한 5단 파이프라인으로 구성된 필터의 설계에 대해 기술하였다. 제안하는 더블록킹 필터는 Verilog HDL로 설계하였으며 기존의 복호기에 통합하여 Modelsim 6.2g 시뮬레이터를 이용해 검증하였다. 기존 필터들과 수행 사이클을 비교한 결과, 제안하는 구조의 설계가 비교적 적은 트랜스포지션 버퍼를 사용했으며 하나의 매크로 블록마다 192사이클이 소요되어 최소 20%의 감소를 확인하였다. 또한, 클럭 게이팅 적용과 메모리 접근 감소로 인해 저전력 구현이 가능하다.

더블록킹 필터는 복호기에서 30~50%의 연산량을 차

지하고 있으므로 제안하는 더블록킹 필터를 적용하여 전체 복호기의 시스템 향상 및 전력 감소 효과를 가져올 것으로 기대된다.

참고 문헌

- [1] S. Wenger, M. Hannuksela, and T. Stockhammer, "Identified H.26L Applications," ITU-T SG 16 Doc. VCEG-L34, Eibsee, 2001.
- [2] Joint Video Team, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification. ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, March 2005.
- [3] 이성만, 박태근, "H.264/AVC를 위한 더블록킹필터의 효율적인 VLSI 구조", 대한전자공학회 논문지, 제 45권, SD편, 제 7호, 2008년 7월.
- [4] D. Garrett, M. Stan, and A. Dean, "Challenges in clockgating for a low power ASIC methodology," Proc. Int. Symp. Low Power Electron. Design, pp. 176-181, 1999.
- [5] Souman Mandal, "Pipeline Processing", M-Tech I, IITB, September 2009.
- [6] Joint Video Team Reference Software JM 9.4.
- [7] M. Parlak, I. Hamzaoglu, "An efficient hardware architecture for H.264 adaptive deblocking filter algorithm," Conference on Adaptive Hardware and Systems, pp. 381-385, 2006.
- [8] Y. W. Huang, T. W. Chen, B. Y. Hsieh, T. C. Wang, T. H. Chang, and L. G. Chen, "Architecture design for deblocking filter in H.264/JVT/AVC," Proc. IEEE Int. Conf. Multimedia Expo., vol. 1, pp. 693-696. July 2003.
- [9] T. M. Liu, W. P. Lee, and C. Y. Lee, "An in/post-loop deblocking filter with hybrid filtering schedule," IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 7, pp. 937-943, July 2007.
- [10] G. Khurana and A. A.Kassim, "A Pipelined Hardware Implementation of In-loop Deblocking Filter in H.264/AVC," IEEE Transactions On Consumer Electronics, Vol. 52, No. 2, pp. 536-540, May 2006.
- [11] T. A. Lin, T. M. Liu, and C. Y. Lee, "A low-power H.264/AVC decoder," IEEE Int. Symp. VLSI Design Autom. Test, pp. 283-288, Apr. 2005.

— 저 자 소 개 —



정 윤 진(학생회원)
 2008년 한밭대학교
 정보통신공학과 공학사
 2008년~현재 한밭대학교 정보
 통신공학과 석사과정.
 <주관심분야 : SoC 플랫폼 설계
 및 검증, 영상신호처리>



류 광 기(정회원)
 1986년 한양대학교 공과대학
 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 부교수
 <주관심분야 : SoC 플랫폼 설계 및 검증, 하드웨
 어/소프트웨어 통합설계 및 통합검증, 멀티미디어
 코덱 설계>