

새로운 Ternary CAM을 이용한 고속 허프만 디코더 설계

正會員 이 광 진*, 김 상 훈**, 이 주 석***, 박 노 경****, 차 균 현*****

A High Speed Huffman Decoder Using New Ternary CAM

Kwang-Jin Lee*, Sang-Hoon Kim**, Joo-Sock Lee***, Noh-Kyung Park****,
Kyun-Hyon Tchah***** *Regular Members*

요 약

본 논문에서는 새로운 Ternary Content Addressable Memory을 설계한 다음, 그것을 MPEG의 기초가 되며 대중매체, HDTV등에 직접 이용되는 JPEG 표준안의 코덱 중에서 가변 길이 복호화기(VLD) 모듈에 적용하였다. 설계한 256 word \times 16 bit 크기의 비트-워드 모두 병렬 방식 Ternary CAM은 CAM셀, 데이터/마스크 레지스터 블럭, 읽기/쓰기 제어 회로, 순차적 어드레스 인코더, 감지 증폭기 블럭등으로 구성되며, 탐색 데이터의 마스크와 저장 데이터의 마스크 기능은 허프만 디코더외에 다른 분야에 CAM이 사용될 수 있도록 해준다. 설계한 허프만 디코더에서는 허프만 코드 테이블의 저장을 위해서 Ternary CAM을 사용하였으며, 본 연구에서 성능이 개선된 Ternary CAM은 shifting 방식을 이용한 초고속의 아키텍처로 블럭화되어 사용되었다. 이와 같은 블럭화 기법은 워드-비트 모두 병렬 방식 CAM의 단점인 많은 전력 소비와 과다한 부하로 인한 신호 지연 문제를 근본적으로 개선함으로써 허프만 디코더의 디코딩 속도의 향상을 가져왔다. Cadence Verilog-XL을 이용해서 설계한 허프만 디코더의 논리 동작 검증은 하였으며, HSPICE를 이용해서 회로 동작 특성을 구하였다. 0.8 μ m 공정파라미터를 이용한 SPICE 시뮬레이션 결과, 허프만 디코더의 critical path의 지연 시간이 17.5ns가 나왔다. 이와같은 결과는 설계된 허프만 디코더가 동작 주파수 50MHz 정도인 고속 동작 코덱의 가변 길이 복호화기의 SPEC.을 증가하는 모듈로 사용 가능함을 보여 준다.

ABSTRACT

In this paper, the huffman decoder which is a part of the decoder in JPEG standard format is designed by using

*삼성전자 반도체 근무
**경인여자전문대 전산과 교수
***고려대학교 전자공학과 박사과정
****호서대학교 정보통신공학과 부교수
*****고려대학교 전자공학과 교수
論文番號:96025-0124
接受日字:1996年1月24日

a new Ternary CAM. First, the 256 word \times 16 bit-size new bit-word all parallel Ternary CAM system is designed and verified using SPICE and CADENCE Verilog-XL, and then the verified novel Ternary CAM is applied to the new huffman decoder architecture of JPEG. So the performance of the designed CAM cell and it's block is verified. The new Ternary CAM has various applications because it has search data mask and storing data mask function, which enable bit-wise search and don't care state storing. When the CAM is used for huffman look-up table in huffman decoder, the CAM is partitioned according to the decoding symbol frequency. The scheme of partitioning CAM for huffman table overcomes the drawbacks of all-parallel CAM with much power and load. So operation speed and power consumption are improved.

I. 서 론

영상 신호는 대부분의 경우 상관 관계가 크기 때문에 많은 중복성을 가지고 있어서, 이러한 중복성을 제거함으로써 데이터 압축 효과를 얻을 수 있다. 많은 발생 빈도를 갖는 데이터에는 짧은 부호 길이를 할당하고, 적은 발생 빈도를 갖는 데이터에는 긴 부호 길이를 할당하여 데이터의 통계적인 중복을 줄이는 가변 길이 부호화(Variable Length Coding)는 런LENGTH 부호화(Run-Length Coding)와 함께 무손실 데이터 압축에 널리 사용되는데, 허프만 코딩은 가변 길이 부호화의 대표적인 기법으로 영상 압축/복원에 관한 국제 표준인 영상의 압축과 복원에 핵심적으로 쓰인다^{[1][2]}. 영상 코덱에서, 가변길이 부호화기/부호화기는 실질적으로 데이터를 압축하고 복원해 주는 부분이므로 꾸준히 연구되고 있다. 그래서 본 논문에서는, MPEG의 기초가 되며 대중매체, HDTV 등에 직접 이용되는 중요한 기법인 JPEG의 디코더중에서 가변 길이 부호화기를 새로운 Ternary Content Addressable Memory과 독창적인 구성방식을 가지고 설계하여, 본 연구에서의 디코더의 실용 가능성을 제시하였다. 특히 JPEG은 MPEG, H.261등과는 달리 절대적인 pre-defined 코드 테이블을 명시하지 않고 있다는 점에서, 응용 분야에 따라서 데이터의 통계적 분포에 맞추어 코드 테이블이 재구성될 필요가 있기 때문에 재구성이 가능한 CAM을 사용하는 것은 바람직하다. 룩업 테이블 방식 중에서, PLA-based 또는 ROM-based 구조를 사용하면 고속의 codec을 구현할 수 있지만, 적용 분야가 한정된다. 다시 말하면, 허프만 테이블의 재구성이 어렵다. 그래서 허프만 테이블의 재구성이 가능하기 위해서는 RAM을 쓰는 것이 바람직하다.

그러나, RAM을 쓰는 경우는 속도의 저하를 가져온다^[3]. 입력 비트열 데이터와 RAM에 저장된 코드 워드 데이터가 일치할 때까지 비교 동작을 수행해야 하기 때문이다. 그러므로 병렬 비교 기능으로 고속의 인코딩과 디코딩이 가능한 CAM을 사용하는 것이 필요하다. 본 논문에서는, 기존의 Ternary CAM 셀 보다 트랜지스터의 수가 적은 새로운 Ternary CAM 셀^[4]을 제시하여 회로 면적의 감소 가능성을 보였다. 또한 허프만 테이블을 블럭화해서 분할함으로써 디코딩 속도를 한층 증가시켰다. Cadence Verilog-XL을 이용해서 설계한 허프만 디코더의 논리 동작 검증을 하였으며, 0.8 μ m CMOS n-well 공정 파라미터를 가지고 HSPICE시뮬레이션을 수행하여 회로 동작 특성을 구하였다.

본 논문 II장에서는 기존의 허프만 디코더의 구조를, III장에서는 새로운 허프만 디코더의 구조를 설명하며, IV장에서는 FDCT의 과정을 거치고 양자화된 8×8 영상을 허프만 부호화하는 과정과 결과 비트열을 제시하며, V장에서는 이 비트열을 수신 받아서 디코딩해주는 설계된 VLD 시뮬레이션 결과를 보여주고 VI장에서 결론을 맺었다.

II. 기존의 허프만 디코더 구조

허프만 코딩은 대표적인 가변 길이 코딩 방법 중 하나이며, 실제로 여러 분야에서 핵심적인 압축 기법으로 사용되고 있다. 영상 압축 코덱에서, 가변 길이 부호화기/복호화기 부분은 실제로 데이터를 무손실 압축하고 복원해 주는 부분으로써 여러 방식으로 그것에 대한 구현이 시도되고 있으며, 여기서 말하는 허프만 디코더는 가변 길이 디코더의 일종이라 말할

수 있다²⁾기.

2.1 허프만 디코더 종류

① tree-based 구조

이 구조는 비트 직렬(serial) 방식으로 디코딩을 하며, 디코딩할 때에 코드 워드 길이 만큼의 디코딩 사이클이 필요하다. 그러므로 디코딩 시간이 길어지는 단점이 있지만, 하드웨어 구조가 간단하고, 회로의 크기가 작다는 장점이 있다²⁾기.

② look-up table based 구조

이 구조는 비트 병렬 방식으로 디코딩을 하며 룩업 테이블로 PLA를 많이 쓴다. 가변적인 코드 워드들을 PLA에 룩업 테이블 형태로 저장한다. 이 구조는 코드 워드에 길이에 상관없이 한 사이클에 디코딩 동작을 수행한다. 그러나 코드 워드들이 많아짐에 따라 하드웨어 구조가 복잡해지고 회로의 크기가 아주 커진다는 단점이 있다. 또 PLA를 사용하는 경우에는 재구성이 불가능하므로, 재구성이 가능한 SRAM, DRAM등과 같은 기존의 메모리나 연상 메모리(CAM)를 사용해야 한다. 그림 2.1은 디코딩의 결과를 일정한 비율로 출력하는 PLA-based 가변 길이 디코더를 보여주고 있다. 이 구조는 코드 워드의 최대 길이가 '16'이라는 가정인 것이다. 그리고 인코딩시에 코드 워드에 추가 비트가 부가되지 않고, 바로 그 다음 코드 워드가 인코딩 되었다는 가정 하에서 구성된 구조이다. JPEG의 경우는 코드 워드 다음에 부가 비트가 붙기 때문에, 이 회로는 수정되어야 한다. 여기서 포

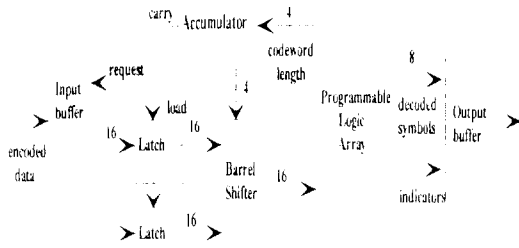


그림 2.1 고정 출력 비율을 갖는 PLA이용 구조
Fig 2.1 Constant output rate PLA-based VLD

인터는 다음 코드 워드를 디코딩하기 위해서 사용되며, 다음 코드 워드 시작 위치를 나타내게 된다²⁾기.

표 2.1은 코드 테이블을 저장하는 매체를 비교하고 있다. 코드의 변경이 필요한 응용 분야의 경우에는 CAM이나 RAM을 써야 하지만, 가변 길이 코드의 경우에 CAM을 쓰는 경우가 훨씬 효과적이다.

표 1. 허프만 코덱에서 코드 테이블 저장 매체 비교
Table 1. The comparison of code table storage medium for Huffman CODEC

	ROM	RAM	PLA	CAM
Waste space	크다	크다	작다	크다
Layout size	작다	크다	작다	크다
Code programmable (변경시)	불가	가능	불가	가능
RandomAccess (Variable length)	느리다	느리다	빠르다	빠르다

표 2.2는 룩업 테이블 방식을 이용한 여러 구조의 허프만 디코더를 비교하고 있다. 만약 pre-defined table을 사용하지 않으며, 여러 응용 분야에 쓸 수 있도록 하기 위해서는 CAM과 ROM 구조를 같이 써야 한다는 것을 보여주고 있다. CAM에는 코드 워드를 저장하고, ROM에는 decoded symbol을 저장한다.

표 2.2 여러 방식의 VLD 비교
Table 2.2 The comparison of various variable length decoder

pre-defined table	응용 분야	재구성	적합한 구성 방식	비 고
사용	한정	불가능	PLA	PLA에 codeword code length, decode symbol 등을 저장
사용	다양	가능	PLA + RAM	RAM에 decode symbol 저장
사용 혹은 사용 안함	다양	가능	CAM + ROM	CAM에 codeword 저장

그림 2.2는 Ternary CAM을 사용하지 않고 기본적인 CAM셀을 사용한 기존의 가변 길이 복호화기이다. 입력 데이터를 마스크 시키면서, 오직 하나의 매치가 나올 때까지 하나의 코드 워드를 위한 디코딩 동작을 수행한다. 코드 워드의 길이 정보를 따로 저장하지 않으므로써 회로의 크기를 작게 하였지만, 그로 인해 한 코드 워드를 디코딩하기 위한 디코딩 시간이 길어지며, 코드 워드의 길이를 알아내기 위한 부가 회로가 필요하다는 점이 단점이다. CAM의 크기를 162 워드로 한 것은 JPEG용 가변 길이 복호화

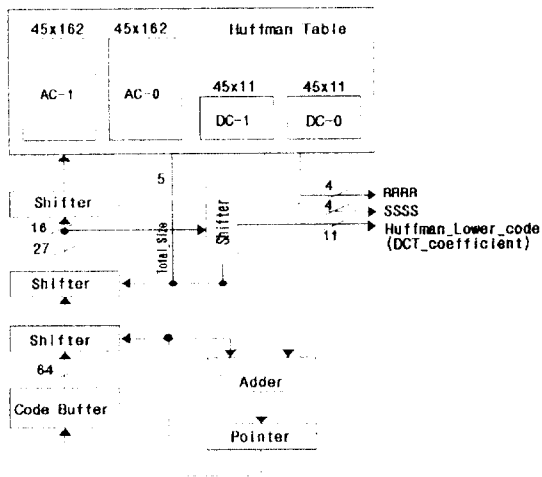


그림 2.2 CAM을 사용하지 않는 기존의 가변 길이 복호화기
Fig. 2.2 Conventional VLD using CAM

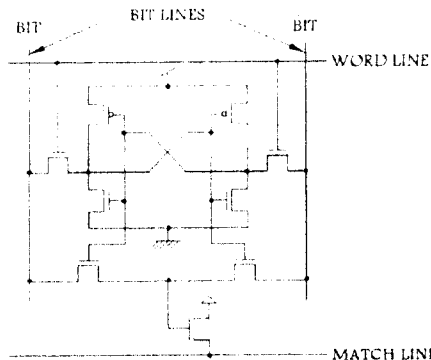


그림 2.3 기본적인 CAM 셀
Fig. 2.3 Basic CAM cell

기를 만들기 위한 것이지만, 이 회로가 JPEG에 응용되기 위해서는 회로의 수정 및 추가적인 부가 회로들이 많이 필요하게 된다.

그림 2.3은 트랜지스터 9개로 구성된 기본적인 CAM 셀을 보여준다.

III. 새로운 허프만 디코더 구조

그림 3.1은 새로운 Ternary CAM을 사용한 JPEG 가변길이 복호화기(VLD)를 나타낸다. 프로그램이 가능하기 때문에 테이블 크기만 조절된다면, MPEG, H.261 등에 적용 가능하다. 그림 3.1에서, 누적이 캐리가 발생하면 입력 데이터가 입력 버퍼에서 26비트씩 래치로 전송되는 것은 허프만 코드 워드의 최대 길이(Lmax = 16)와 최대의 AC계수 그룹번호(Smax = 10)의 합이 26이기 때문이다. 실제적으로, 압축 데이터를 비트열로 보낼때, 코드워드에 부가적으로 계수 그룹번호 만큼의 계수 위치 정보를 코드워드에 붙여서 전송하게 된다. 그러므로 CAM 블록에는 허프만 코드 워드 테이블뿐만 아니라 코드 워드 길이와 계수 그룹번호를 더한 길이(최대 길이=26)의 정보가 저장되어 있다.

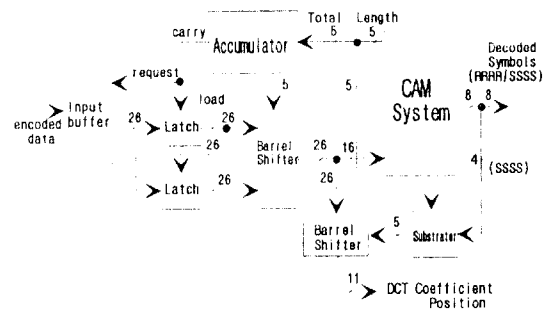


그림 3.1 새로운 Ternary CAM을 이용한 JPEG 허프만 디코더
Fig. 3.1 JPEG Huffman decoder using new Ternary CAM

디코딩 동작을 살펴보면 다음과 같다.

1. 입력데이터는 D 래치로 전송된 다음 52x26 배럴 쉬프터에 의해 CAM에 저장된 허프만 코드들과 비교를 수행하게 된다.

- 나. 데이터의 일치가 일어날 경우는 디코드 심벌인 RLC코드를 출력하고 Match Flag를 'High'로 만들며, 총길이(Total Length)정보를 누적기와 5비트 감산기에 보낸다. 여기서, 누적기는 다음 코드 워드의 시작 비트를 지시해주는 포인터의 역할을 하고, 감산기는 26×11 배열 슈프터가 현재 디코딩된 코드워드만큼을 이동해서 양자화 계수의 위치 정보값을 출력시키 주는 역할을 한다.
- 다. 만약 현재의 디코딩 사이클에서 데이터의 일치가 일어나지 않으면, 빈도수가 낮은 CAM 블록을 다음 사이클에서 검색한다. 이때에도 매칭이 일어나지 않으면 매치 플래그는 'Low'로 되어 외부에 입력 데이터의 왜곡이 있음을 알린다.
- 르. 누적기는 값이 '26'이 되면 캐리 신호를 발생시키고 '0'을 된다. 이 캐리 신호는 입력 비퍼에서 26 비트의 새로운 입력 데이터를 받아들여 디코딩 동작을 수행하도록 한다.

이 동작들은 한 동작 사이클내에서 이루어지기 때문에 매우 고속으로 동작하며, 병렬로 디코드 값을 출력하므로 매우 높은 throughput을 낸다.

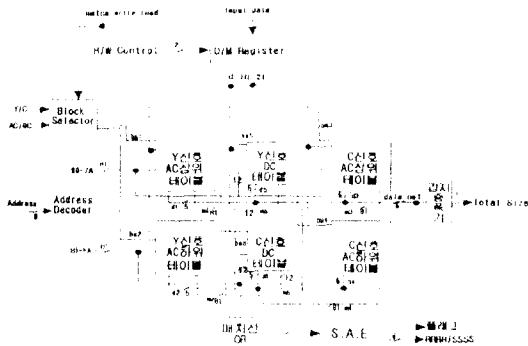


그림 3.2 허프만 디코더에 쓰인 CAM 시스템의 구조
Fig. 3.2 CAM system structure for Huffman decoder

그림 3.2는 CAM 시스템의 구조를 보여주고 있다. 여기서 각각의 코드 테이블은 CAM 셀과 RAM 셀로 구성되며, CAM 셀은 허프만 코드워드를 저장하기 위해서, RAM 셀은 총 길이 정보를 저장하기 위해서

쓰인다. CAM은 word-bit all parallel 방식으로 동작하므로, 비교 모드(match mode)에서는 많은 전력의 소비와 과도한 부하로 인한 신호의 지연이 있게 된다. 이 단점을 보완하기 위해서, CAM 셀 어레이를 블럭화해서 사용한다. 다시 말하자면, 짧은 코드워드들이 많이 존재하는 run/level 부분을 상위 블럭에 저장하고 그 외의 run/level 부분은 하위 블럭에 저장해서 사용한다. 디코딩할때는 항상 먼저 상위 블럭만 검색하고, 매칭이 일어나지 않을 경우에는 그 다음 사이클에서 하위 블럭을 접근하는 방식을 취한다. 이와같은 동작식 방식은 CAM 셀의 빠른 탐색기능과 함께 전력소비를 신호지연의 감소를 가져오게 함으로써 성능향상에 기여하였다. CAM 셀 어레이 내부 블럭들은 블럭 선택기에 의해 제어된다.

허프만 테이블

R/S	Size	R/S
0/0	신호	0/0
	AC 상위	DC
	cam셀 ram셀 81-16 81-5	cam셀 ram셀 12-16 12-5
7/A	11	7/A
8/1	신호	8/1
	AC 상위	신호
	cam셀 ram셀 81-16 81-5	DC
F/A	11	F/A
		cam셀 ram셀 31 16 81-5

그림 3.3 허프만 테이블을 저장하는 CAM의 내부구조
Fig. 3.3 Inner configuration of Huffman table

그림 3.4은 블럭 선택기를 보여주고 있는데, 각 블럭은 휘도/색신호(Y/C), DC/AC 등의 신호 상태에 따라서 블럭 선택기에 의해 선택된다. 그림 3.5는 허프만 테이블을 저장한 CAM셀과 RAM을 보여주고 있다. 만약 CAM에서 매치가 발생하면, RAM셀의 해당 워드선을 'high'로 만들어서 RAM에 저장된 총 길이 정보를 출력할 수 있도록 함을 알 수 있다. 이러한 구조를 한 word에 저장하기 위해서는 don't care 상태의 저장이 가능한 ternary CAM을 사용해야 한다. 또한 CAM을 허프만 테이블로 쓰기 위해서는 don't

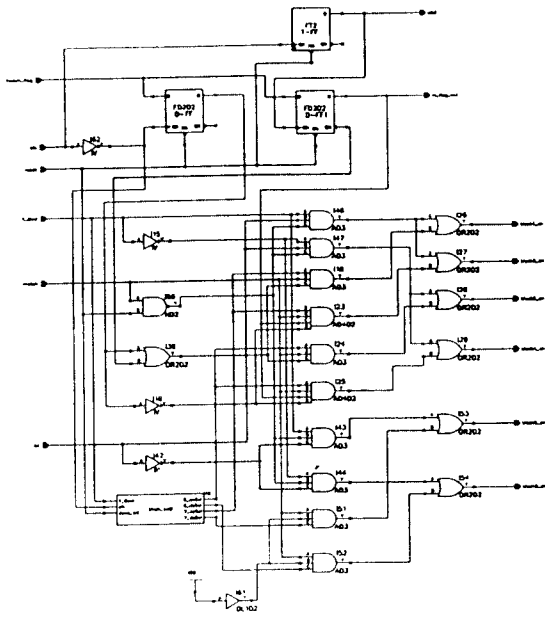


그림 3.4 블럭 선택기
Fig 3.4 Block selector

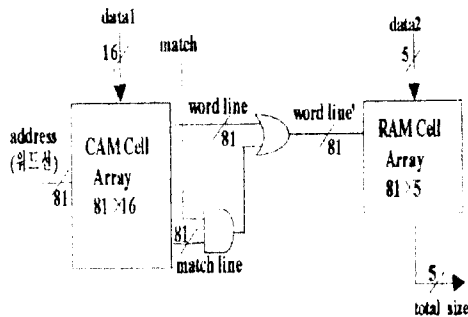
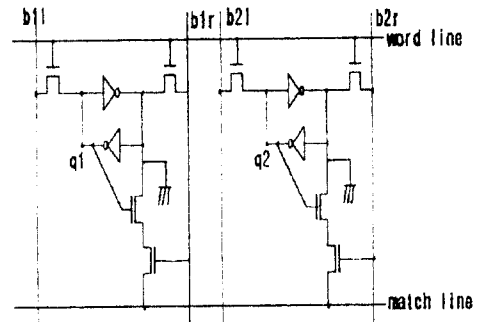
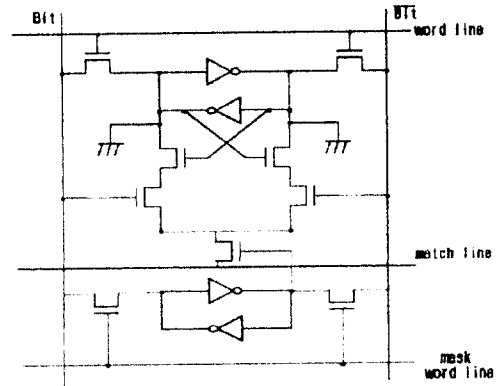


그림 3.5 CAM셀과 RAM셀의 연결 관계
Fig 3.5 The Interconnection between CAM and RAM

care 상태가 저장 가능한 ternary CAM을 써야 한다. 그림 3.6에는 기존의 ternary CAM이 제시되어 있다⁶⁾. 그림 3.6(a) 회로는 데이터를 쓰거나 읽을 때 '01', '10', '00' 등에 대한 코딩과 디코딩 작업이 필요하며, 이 값들은 '1', '0' 혹은 don't care 값과 매핑된다. '00' 혹은 '11'이 저장된 경우는 매치선의 방진 경로를 끊어져서 매치선이 계속 'high'로 남게 되므로 이 값



(a) 기존의 Ternary CAM 셀(tr = 16)



(b) 기존의 Ternary CAM 셀(tr = 17)

그림 3.6 기존의 Ternary CAM 셀들
Fig 3.6 Conventional Ternary CAM cells

- (a) CAM cell with transistor = 16
- (b) CAM cell with transistor = 17

을 don't care로 한다. 이 회로는 two binary static CAM 이므로 'N'비트 워드를 저장하기 위해서는 2. N 비트가 필요하게 되고, CAM 셀 어레이의 width가 증가하게 된다. 그림 3.6(b)의 회로는 data를 읽고, 저장하기 위해 데이터 워드선과 마스크 워드선을 각각 구동해야 하며, 트랜지스터의 수가 17개나 되고 CAM 셀 어레이의 height가 증가하는 단점이 있다.

본 논문에서는, 위 두 회로들의 단점을 극복한 새로운 ternary CAM을 그림 3.7과 같이 설계하였다. 그림 3.7의 회로는 기존의 ternary CAM 셀 회로들보다

트랜지스터의 수가 적고(그림 3.6(a)(b) 참조), 데이터를 저장하고 읽을 때 부가적으로 데이터 인코딩이나 디코딩이 작업이 불필요하며(그림 3.6(a) 참조), 한 번의 쓰기 동작으로 셀에 값을 저장할 수 있다(그림 3.6(b) 참조).

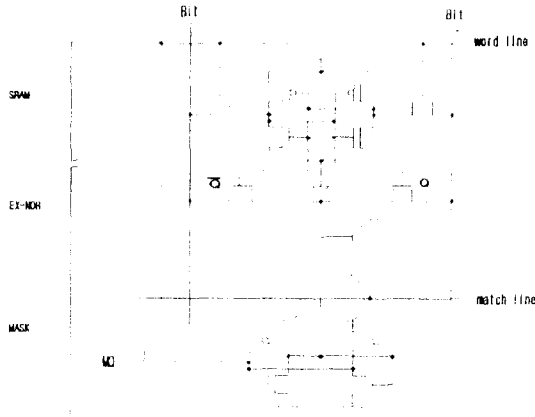


그림 3.7 새로운 Ternary CAM 회로
Fig 3.7 New Ternary CAM circuit

IV. 허프만 코드 부호화 과정

FDCT의 과정을 거치고 양자화된 한 8×8 영상을 허프만 부호화 하는 과정과 결과 비트열을 제시하고자 한다^[6]. 만약 FDCT를 거친 8×8 영상이 다음과 같이 주어진다면,

538	-12	-2	-15	0	0	0	0
448	-6	-30	0	0	0	0	0
-3	449	0	1	0	0	0	0
0	0	-7	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

그림 4.1 FDCT 과정 거친 후의 8×8입력 영상의 예
Fig 4.1 An example of 8×8 Input Image

이 영상 데이터는 런랭스 코딩 과정을 거치면 (이진 블록의 DC계수값 = 10이라 가정)

DC: (A, 528)

AC: (0, -12) (0, 448) (0, -3) (0, -6) (0, -2) (0, -15) (0, -30) (0, 449) (8, 1) (0, -7) EOB와 같은 RLC 코드 형태로 만들어 지게 된다.

이것을 JPEG 표준안에서 제시된 전형적인 허프만 테이블을 써서 가변길이 부호화를 하고 26비트씩 잘라서 나타내면 다음과 같다.

```

첫번째 입력 비트열(26 비트):
11 1111 1010 0001 0000 1011 0011 (h' 3FA701B3)
(A) (0.4)

두번째 입력 비트열(26 비트):
11 1111 1110 0000 1011 1000 0000 (h' 3FF6B350)
(0.3) (0.2)

세번째 입력 비트열(26 비트):
10 0100 0010 1011 0110 0001 1010 (h' 242B31A)
(0.3) (0.2) (0.4) (0.5)

네번째 입력 비트열(26 비트):
00 0011 1111 1111 0000 0101 1100 (h' 03E705C)
(0.3)

다섯번째 입력 비트열(26 비트):
00 0111 1111 0001 1000 0010 1010 (h' 07F782A)
(8.1) (0.3) (EOB)
    
```

V. 설계한 디코더 성능 검증

위에서 제시한 8×8 영상 데이터를 실제로 디코딩 해봄으로써 설계한 디코더의 성능을 검증하였다. 0.8 um 셀 라이브러리를 가지고 설계한 전체 허프만 디코더 시스템의 동작 확인은 Cadence Verilog-XL을 가지고 수행하였으며, full-custom 방식으로 설계된 허프만 디코더의 각 블럭별 회로는 0.8 um CMOS 모델 파라미터를 가지고 HSPICE 시뮬레이션을 수행하여, 지어진 시간과 전력 소비를 구하였다. 시뮬레이션 환경은 다음과 같다.

Vdd = 5v, T = room temperature, CLK 주파수 = 41MHz

그림 5.1은 허프만 디코더의 전체 회로도를 보여주고

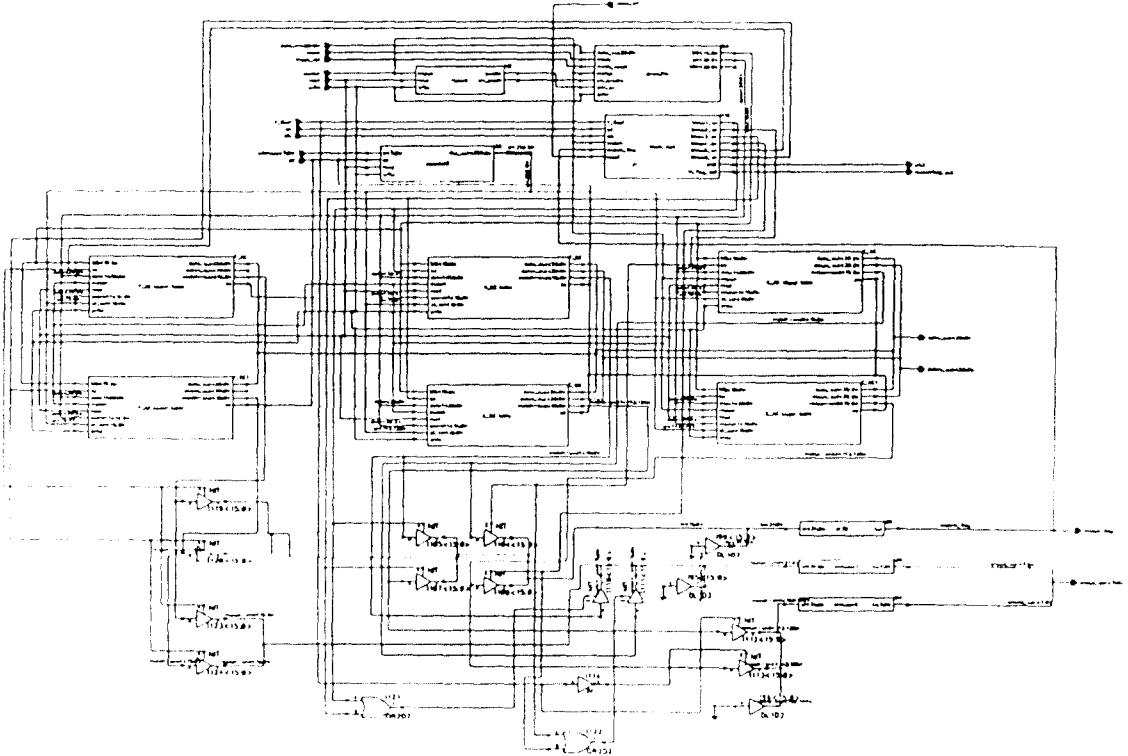


그림 5.1 허프만 디코더 전체 회로도
Fig 5.1 The Circuit of Huffman decoder

저장 데이터 마스크 된 경우

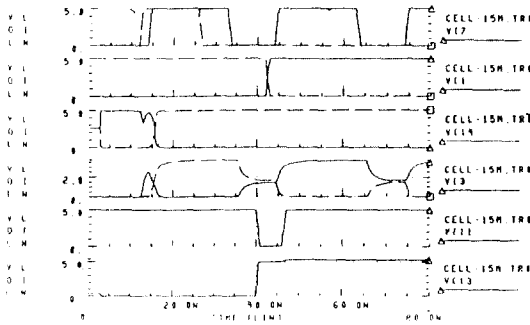


그림 5.2 새로운 Ternary CAM Cell의 쓰기·탐색 동작 파형
Fig 5.2 Write-match operation waveform of new Ternary CAM Cell

있으며, CAM 시스템은 앞에서 언급했던 것처럼 블럭화 되어 사용되었다.

위 로직 시뮬레이션 결과는 앞 4 장에서 제시한 임의의 한 8×8 영상을 허프만 부호화한 결과와 같음을 보여 준다. 만약 상단의 허프만 테이블에서 매치가 안 일어나면, 그 다음 클럭에서 하단의 허프만 테이블을 접근하여 디코딩 동작을 수행함을 알 수 있다. 설계한 허프만 디코더의 각 블럭별 회로의 전력 소비와 지연 시간을 알기 위해서 수행한 회로 시뮬레이션 결과가 표 5.1에 제시되어 있다. 표 5.1의 결과처럼 CAM을 블럭화해서 사용함으로써 전력 이득뿐만 아니라, 지연 시간이 개선됨을 알 수 있다.

허프만 디코더가 디코딩 동작을 수행할 때의 critical path는 누적기에서 캐리 신호가 발생하여 입력 버퍼에 데이터를 요구할 경우이며, 이 때의 데이터 경로는 "D 래치 + 52×26 배럴 쉬프트 + CAM 시스템

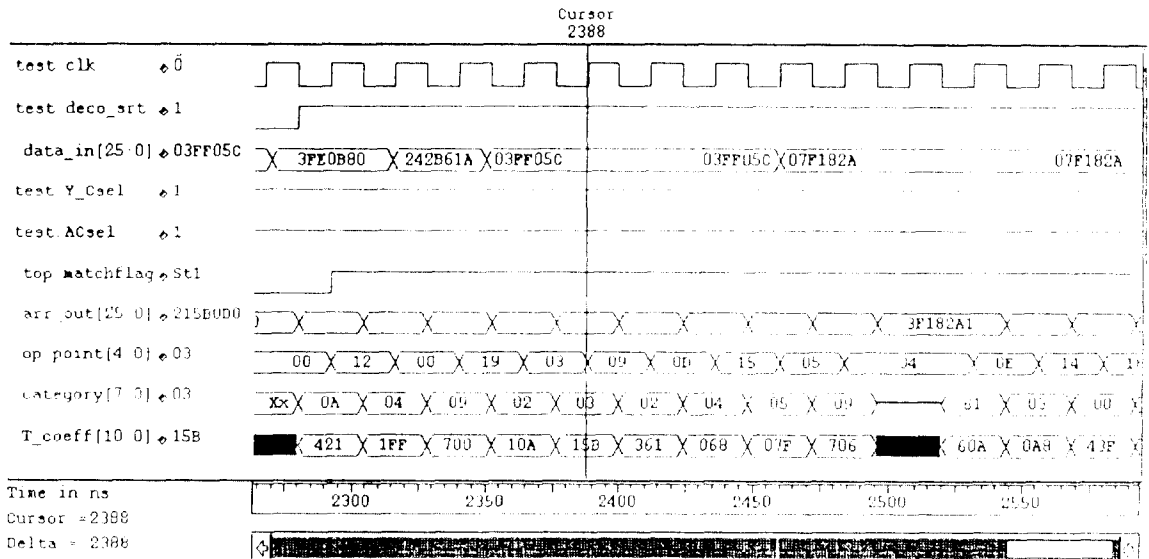


그림 5.3 허프만 디코더의 로직 시뮬레이션 출력 파형
Fig 5.3 Logic simulation waveform of Huffman decoder

표 5.1 제안한 허프만 디코더(VLD) 시스템의 지연 시간 및 전력 소비

Table 5.1 The delay time and power consumption of proposed huffman decoder

블럭	지연 시간	전력 소비
CAM 시스템	11.6 ns	25.49 mW
	12.8 ns	(81 × 2블럭화) 33.07 mW
	(디코딩 동작은 match 모드에서 수행됨)	(162 워드경우)
5 Bit Subtractor	2.74 ns	0.88 mW
5 Bit Accumulator	3.5 ns	0.864 mW
26 Bit D latch (PIPO) × 2	0.5 ns	5.668 mW
26 × 11 Barrel Shifter	1.1 ns	0.448 mW
52 × 26 Barrel Shifter	1.6 ns	0.873 mW

+누적이"이다. 이 때의 지연 시간은 17.5 ns (57MHz)이며, 출력 throughput은 456Mbps (57 × 8)이다. 이와 같은 결과는 설계한 허프만 디코더는 동작 주파수가

50MHz 정도인 고속 동작 코덱의 가변 길이 부호화기의 모듈로 쓰일 수 있음을 보여 준다. 허프만 코드 테이블의 재구성이 가능하므로, 다른 응용 분야 코덱의 가변 길이 부호화기 모듈에 허프만 디코더를 이용될 수 있다.

VI. 결 론

본 논문에서는, 동영상(MPEG)의 기초가 되며 대중매체, HDTV등에 직접 이용되는 중요한 기법인 JPEG 영상의 처리를 위한 디코더 중에서 가변길이 부호화기(VLD) 부분을 새로운 개발된 Ternary CAM과 메모리 블럭킹을 이용한 아키텍처를 적용하여 설계하였다. 허프만 테이블을 블럭화해서 저장함으로써 VLD를 설계한 기존의 회로 보다 동작 속도의 개선을 가져왔다.

설계한 허프만 디코더에서는 허프만 코드 테이블의 저장을 위해서 unknown 상태까지 고려한 Ternary CAM을 사용하였으며, 그 Ternary CAM은 블럭화의 방식으로 인하여 탐색방식에서 더 큰 효율성을 확보하게 되었다. 이와 같은 블럭화 기법은 워드-비트 모

두 병렬 방식 CAM의 단점인 많은 전력 소비와 과도한 부하로 인한 신호 지연 문제를 개선함으로써 허프만 디코더의 디코딩 속도의 향상을 가져왔다. Cadence Verilog-XL을 이용해서 설계한 허프만 디코더의 논리 동작 검증을 하였으며, HSPICE를 이용해서 회로 동작 특성을 구하였다. 0.8 μm 공정 파라미터를 이용한 SPICE 시뮬레이션 결과, 허프만 디코더의 critical path의 지연 시간이 17.5ns가 나왔다. 이것은 본 허프만 디코더가 57MHz 주파수에서 동작할 수 있음을 의미하며, 이 때의 출력 throughput은 456Mbps 이다. 이와 같은 결과는 설계한 허프만 디코더는 동작 주파수가 50MHz 정도인 고속 동작 코덱의 가변 길이 복호화기의 모듈로 쓰일 수 있음을 보여 준다.

또한 본 논문에서 제시한 새로운 Ternary CAM의 cell 구조 및 전체 구성방식은 허프만 디코더 분야 이외에도 고속 병렬 비교 동작을 필요로 하는 분야: 캐쉬 메모리의 룩업 테이블, 패턴 인식, 데이터 플로우 컴퓨터, 인공 지능 분야 등에 훌륭히 응용될 수 있다.

참 고 문 헌

1. H. Kadota et al., "An 8-kbit Content Addressable and Reentrant Memory", IEEE J. Solid state circuits, vol. sc-20, No 5, pp. 951-957, Oct. 1985.
2. Shin-Fu et al, "Designing High-Throughput VLC Decoder Part I-Concurrent VLSI Architectures", IEEE Trans. on Circuit and Systems, Vol. 2, pp187-196, No 2, June, 1992
3. E. Komoto et al, "A High-Speed and Compact-Size JPEG Huffman Decoder using CAM", Symp. VLSI ckt, pp37-38, 1993
4. Geunhoe Kim et al, "Design of Variable Length Decoder based on CAM", Proc. JTC-CSCC'94, pp950-954, 1994
5. 차 균현 외, "Content Addressable Memory 설계 및 제작에 관한 연구", MPW(I) 최종 연구보고서, 한국 전자통신연구소, 1989
6. S. I. Chae et al., "Content Addressable Memory for VLSI Pattern Inspection", IEEE J. Solid state circuits, vol. sc-23, pp. 74-78, No 1, Feb. 1988
7. Shaw-Min Lei et al, "An entropy coding system for

digital HDTV applications", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 1, No. 1, pp. 147-155, March 1991

8. "JPEG Committee Draft, Part 1", ISO/IEC JTC1/SC2, Mar 1991

이 광 진(Kwang-Jin Lee)

정회원

1970년 1월 1일생

1994년 2월: 고려대학교 전자공학과 학사

1996년 2월: 고려대학교 전자공학과 석사

1996년 2월~현재: 삼성전자 반도체 근무

※ 주관심분야: 반도체 메모리 설계, 영상신호처리 설계

김 상 훈(Sang-Hoon Kim)

정회원

1964년 7월 26일생

1987년 2월: 고려대학교 전자공학과 학사

1989년 2월: 고려대학교 전자공학과 석사

1993년~현재: 고려대학교 전자공학과 박사과정

1989년~1994년: LG반도체 근무

1994년~현재: 경인여자전문대 전산과 전임강사

※ 주관심분야: ASIC설계, neural net을 이용한 신호 처리

이 주 석(Joo-Sock Lee)

정회원

1961년 1월 26일생

1983년 2월: 서강대학교 전자공학과 학사

1985년 8월: 고려대학교 전자공학과 석사

1984년~1995년: LG반도체, LG전자 근무

1995년 3월~현재: 고려대학교 전자공학과 박사과정

※ 주관심분야: 영상신호처리 설계, 특수메모리 설계

박 노 경(Noh-Kyung Park)

정회원

1958년 1월 7일생

1984년 2월: 고려대학교 전자공학과 학사

1986년 2월: 고려대학교 전자공학과 석사

1990년 2월: 고려대학교 전자공학과 박사

1988년~현재: 호서대학교 정보통신공학과 부교수

※ 주관심분야: 영상신호처리 ASIC설계, 회로 및 시스템 설계

차 균 현(Kyun-Hyon Tchah)

정회원

고려대학교 전자공학과 교수.

한국통신학회 수석부회장