

# Content Addressable and Reentrant Memory (CARM)의 설계에 관한 연구

正會員 李 準 洙\* 正會員 白 寅 天\* 正會員 朴 商 奉\*  
正會員 朴 魯 京\* 正會員 車 均 鉉\*

## A Study on the Design of Content Addressable and Reentrant Memory(CARM)

Jun Soo LEE\*, In Cheon PAIK\*, Sang Bong PARK\*, Nho Kyung PARK\*,  
Kyun Hyon TCHAH\* *Regular Members*

**要 約** 본 논문에서는 16 워드 X 8비트 Content Addressable and Reentrant Memory(CARM)을 설계하였다. CARM은 읽기, 저장, 매칭, 리엔트런트 (Reentrant)의 4가지 동작 모드를 수행한다. CARM의 읽기와 저장 동작은 기존의 스태틱 RAM과 같다. CARM은 칩 상에서 캐잉의 회수(Garbage collection)를 조건적으로 수행할 수 있는 리엔트런트 동작을 가지고 있다. 이러한 기능은 다이내믹 데이터 플로우 컴퓨터의 고속 매칭 유닛에 사용될 수 있다. CARM은 또한 매칭 어드레스들 그들의 우선권에 따라 순차적으로 인코딩을 할 수 있는 기능을 가지고 있다. 이러한 CARM은 전체적으로 메모리 셀, 순차적 어드레스 인코더(Sequential Address Encoder, S.A.E), 리엔트런트 동작, 읽기/저장 제어, 데이터/마스킹, 레지스터, 감지 증폭기, 인코더, 디코더 등의 8개의 블록으로 구성된다. CARM은 데이터 플로우 컴퓨터, 패턴 인식, 테이블 룩업(Table look-up), 영상처리 등에 응용될 수 있을 것이다. 설계된 회로에 대해 각 동작별로 Apollo 워크스테이션의 QUICKSIM을 이용하여 논리 시뮬레이션을 하였고, 각 블록별 회로의 SPICE 시뮬레이션을 하였다. 시뮬레이션 결과 액세스 타임은 26ns였고, 매치 동작을 수행하는 때에는 41ns의 지연 시간이 소요됐다. 전체 레이아웃은 3 $\mu$ m n well CMOS 공정에 따른 설계 규칙을 이용하여 수행하였다.

**ABSTRACT** In this paper, 16word X 8bit Content Addressable and Reentrant Memory (CARM) is described. This device has 4 operation modes(read, write, match, reentrant). The read and write operation of CARM is like that of static RAM. CARM has the reentrant mode operation where the on chip garbage collection is accomplished conditionally. This function can be used for high speed matching unit of dynamic data flow computer. And CARM also can encode matching address sequentially according to their priority. CARM consists of 8 blocks(CAM cell, Sequential Address Encoder(S.A.E), Reentrant operation, Read/Write control circuit, Data/Mask Register, Sense Amplifier, Encoder, Decoder).

Designed CARM can be used in data flow computer, pattern inspection, table look-up, image processing. The simulation is performed using the QUICKSIM logic simulator and Pspice circuit simulator. Having hierarchical structure, the layout was done using the 3 $\mu$ m n well CMOS technology of the ETRI design rule.

### I. 서 론

\*高麗大學校 電子工學科  
Dept. of Electronic Eng., Korea University  
論文番號 : 91- 5 (接受1990. 10. 10)

RAM을 사용할 경우에 어떤 데이터를 탐색하

러면 어드레스에 따라 순차적으로 데이터를 탐색해야 하므로 데이터 탐색 속도가 매우 느리다. 그러므로, 데이터 플로우 컴퓨터, 패턴 인식, 테이블 룩업, 고속 변환 데이터 베이스의 정보 저장 및 수정, 레이다 신호의 추적과 처리, 영상 처리 등과 같이 광범위한 분야에서 요구되는 빠른 데이터 탐색 속도를 뒷받침해줄 수 없다<sup>1)2)6)</sup>. 따라서, 데이터의 고속 병렬 탐색을 할 수 있는 연상 메모리에 관한 연구는 지난 30여년간 많은 주목을 받아 왔다. 그리고, 지금까지 여러 응용분야에서 이용할 수 있는 다양한 종류의 연상 메모리가 연구되어 왔다.

본 논문에서는 광범위한 응용 분야를 가지고 고속 병렬 데이터 탐색을 수행하는 16워드×8비트 Content Addressable and Reentrant Memory (CARM)를 설계하였다.

CARM은 전체적으로 8개의 블럭으로 구성된다. 즉 메모리 셀, 데이터/마스크 레지스터, 순차적 어드레스 인코더, 리엔트런트 동작 회로, 감지 증폭기, 읽기/저장 제어, 인코더, 디코더 회로등이다. 여기에서 데이터/마스크 레지스터는 저장과 매치동작시에 입력 데이터를 받아들이는 부분이며, 8개의 비트로 구성된 한 워드 중에서 MSB에 대해서만 마스크를 할 수 있도록 하였다. 순차적 어드레스 인코더는 여러 워드에서 매칭이 발생했을때 우선권에 따라 순차적으로 인코딩을 할 수 있도록 해준다. 그리고, 리엔트런트 동작 회로는 선택적으로 캐영역 회수를 할 수 있는 회로이다. 위와 같은 8개의 블럭에 대해 회로설계를 완료한 후 논리 시뮬레이션 및 회로 시뮬레이션을 마치고 3 $\mu$ m CMOS n-well 공정에 따라 레이아웃을 마쳤다.

각 장별로 내용을 살펴보면 다음과 같다. II장에서는 메모리 셀, 매치 동작, 데이터/마스크 레지스터, 읽기/저장 제어, 순차적 어드레스 인코더, 리엔트런트 동작, 감지 증폭기, 디코더, 인코더 블럭들에 대한 회로구성과 동작 설명을 하였다. III장에서는 읽기, 저장, 매칭, 리엔트런트 동작에 관하여 설명하였다. IV장에서는 Apollo 워크스테이션의 Quicksim에서 수행한 논리 시뮬

레이션, Pspice를 사용한 회로 시뮬레이션 결과를 설명한 후 ETRI의 3 $\mu$ m CMOS n-well 공정에 따른 레이아웃 결과를 설명하였다. V 장에서는 연구결과를 기술하였다.

## II. CARM의 구조 및 설계

본 논문에서는 어드레스가 4비트이고 데이터가 8비트로 구성된 8비트×16워드 CARM을 설계하였다.

CARM은 메모리 셀, 순차적 어드레스 인코더, 리엔트런트 동작, 읽기/저장 제어회로, 데이터/마스크 레지스터, 감지 증폭기, 인코더, 디코더 등으로 이루어졌으며, 전체 블럭도는 그림 2.1에 나타내었다. 그림 2.1에서 각 블럭의 동작 설명은 다음과 같다.

메모리 셀은 6개의 트랜지스터로 구성된 스테틱 RAM 회로와 매치 동작을 수행하기 위한 Exclusive OR 구조로 구성된다<sup>2)</sup>. 순차적 어드레스 인코더는 매칭 데이터가 여러 워드에서 발생된 경우 우선권에따라 순차적으로 인코딩을 수행하는 회로이다<sup>1)2)3)</sup>. 여기에서의 출력은 인코더로 입력된다. 리엔트런트 동작은 On chip 캐영역 회수를 수행하는 회로이다<sup>2)</sup>. 데이터/마스크 레지스터는 저장 또는 매치 동작시에 입력 데이터를 받아들이는 부분이고, 여기서는 MSB에 대해 마스크할 수 있도록 설계하였다. 읽기/저장 제어 회로는 칩 외부로부터 명령을 받아서 입력 데이터 또는 Precharge 신호를 Bit 선과 Bit 선으로 연결시키는 데이터/마스크 레지스터를 제어한다.

감지 증폭기는 읽기 동작시 속도를 증가시키기 위한 회로로서, 기본적인 차동 증폭기의 구조를 이용하였다. 인코더는 기본적인 ROM 구조를 이용하였다. 디코더는 NAND 디코더를 사용하였다.

각 블럭에 대한 회로와 수행기능은 다음과 같다.

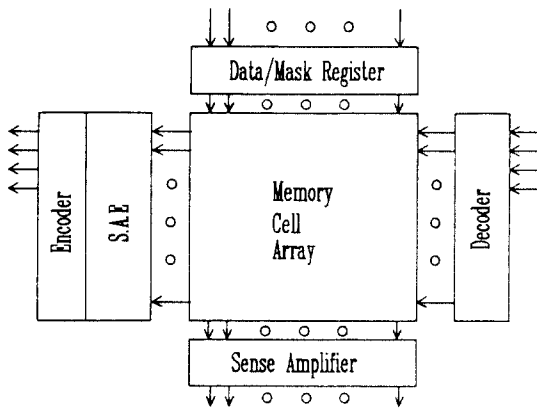


그림 2.1 CARM의 전체 블럭도  
Fig. 2.1 Block diagram of CARM

2.1. 메모리 셀

메모리 셀 회로는 6개의 트랜지스터를 사용하는 스테틱 RAM과 데이터 매칭 동작을 위한 4개의 트랜지스터로 구성된 Exclusive OR 구조를 지닌다<sup>(2)(6)</sup>. 그림 2.2는 메모리 셀의 회로이다.

이 회로에서 읽기나 저장 할 때의 동작은 기존의 RAM과 같다. 매칭동작인 경우, 매치 Precharge 선은 미리 5V로 Precharge되고 메모리 셀에 1이 저장되어 있을 때 Bit 선으로 1이 입력

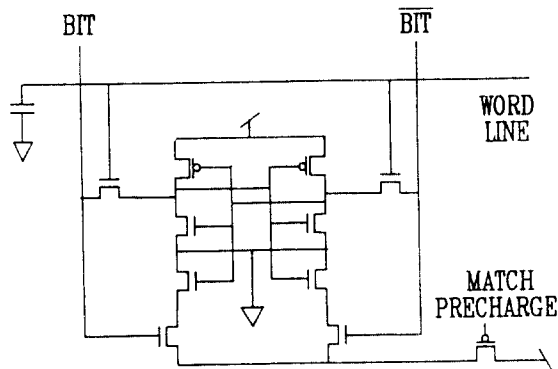


그림 2.2 메모리 셀 회로도  
Fig. 2.2 Memory cell

되면 회로 아래 부분의 4개의 트랜지스터가 형성하는 두 개의 경로를 끊어서 매치선은 1로 유지된다. 만약 메모리 셀의 내용과 Bit 선의 내용이 다르면 두 개의 경로중 하나가 접지로 연결되어 매치 선은 0으로 된다.

2.2. 읽기/저장 제어 회로

읽기/저장 제어회로는 칩의 외부로부터 읽기, 저장 또는 매치 명령을 받아 데이터 또는 Precharge 신호를 Bit 선과 Bit 선으로 넣어주게 위해 데이터/마스크 레지스터를 제어하는 회로이다. 이 회로가 동작하는 방법은 아래의 3가지 기준에 의거한다.

- 읽기 명령 : 입력 데이터 선과 Precharge 신호를 모두 끊어주어야 한다.
- 저장 명령 : 입력 데이터의 경로를 열어주고 Precharge 신호는 끊어주어야 한다.
- 매치 명령 : 저장 명령때와 같이 입력 데이터의 경로를 열어주고 precharge 신호는 끊어주어야 한다.

위의 3가지 기준에 의해 진리표를 작성하면 표 2.1과 같다.

표 2.1 읽기/저장 제어 회로의 진리표

INPUT			OUTPUT	
읽기	저장	매치	Enable	Precharge
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	0	0

위의 진리표에서 Enable과 Precharge 신호는 (2.1)과 같다.

$$E = r + \overline{w} \oplus m = \overline{\overline{r} + w \oplus m} \quad (2.1)$$

$$P = \overline{r}(w \oplus m) + \overline{r}w$$

(2.1)식을 게이트회로로 구현하면 그림 2.3과 같다.

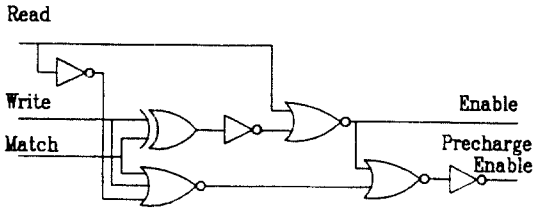


그림 2.3 읽기/저장 제어 회로도  
Fig. 2.3 Read/Write control circuit

데이터/마스킹 레지스터는 읽기/저장 제어 회로에서 Enable과 Precharge Enable 신호를 받아서 입력 데이터 Q 또는 Precharge 신호를 4×2 MUX를 통해 메모리 셀의 Bit 선과  $\overline{\text{Bit}}$  선으로 입력시켜준다. 회로는 그림 2.4와 같다.

이 회로의 동작을 설명하면 다음과 같다. Enable 입력으로 1이 들어오고 Precharge Enable 신호로 1이 들어오는 경우, 저장 또는 매치 명령이 들어오는 것으로서 입력 데이터 Q는 위쪽의 두 개의 전송 게이트를 통해 Bit 선과  $\overline{\text{Bit}}$  선으로 전해진다. Enable 신호가 0이 입력되고 Precharge Enable 신호로 1이 입력되는 경우, 읽기 명령이 들어오는 경우로서 입력데이터의 경로는 끊어지고 Precharge Enable 신호도 끊어져서 Bit 선과  $\overline{\text{Bit}}$  선은 메모리 셀 내부의 내용에 따라 선택적으로 방전하게 된다.

이 회로에서의 마스킹 기능은 Bit 선과  $\overline{\text{Bit}}$  선 양쪽 모두에 nMOS 트랜지스터를 통해 선택적으로 접지시켜서 이루어진다. 이런 식으로 Bit 선과  $\overline{\text{Bit}}$  선을 접지시켰을 경우, 메모리 셀의 XOR 회로의 방전 경로가 모두 끊어지므로 매치 선은 자동적으로 High가 되고, 이 경우 입력된 데이터와 매치가 된 것으로 생각할 수 있다.

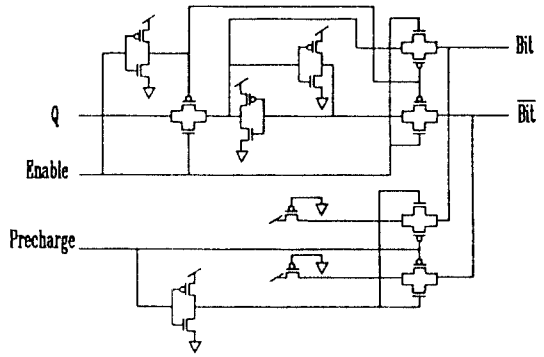


그림 2.4 데이터/마스킹 레지스터 회로도  
Fig. 2.4 Data/Mask Register circuit

#### 2.4. 감지 증폭기

CARM에서는 Bit 선과  $\overline{\text{Bit}}$  선을 모두 Precharge 하는 방법을 사용하는데 읽기 동작시 메모리 cell 안의 트랜지스터들은 매우 작고 Bit 선과  $\overline{\text{Bit}}$  선의 커패시턴스는 매우 크므로 두 선중 하나를 pull down 시키는데 많은 시간이 필요하다. 이것은 액세스 시간을 매우 느리게 한다. 감지 증폭기는 두 선간의 전압차를 감지해서 증폭하여 읽기 동작에서의 속도를 증가시킨다<sup>5)</sup>.

그러한 필요에 따라 그림 2.5(a)의 회로를 감지 증폭기로 사용하였다. 이 회로는, 먼저 Bit 선과  $\overline{\text{Bit}}$  선이 모두 똑같이 5V로 Precharge 되어 있을 때에는 오른쪽과 왼쪽의 경로에 똑같이 I의 전류가 흐른다. 그러나, 만약 Bit 선이 방전하기 시작하면,  $\overline{\text{Bit}}$  선에서 들어가는 오른쪽 아래 n 트랜지스터의  $V_{GS}$  값이 상대적으로 증가하고 그쪽 경로에 I가 흘러서 출력값으로 '0'이 출력된다. 그 반대의 경우도 똑같은 방식으로 수행된다.

그림 2.5에 보통 인버터와  $\beta_p/\beta_n$  값을 크게 하여 인버팅 전압을 크게한 인버터의 경우, 그리고 CARM에서의 감지 증폭기의 속도 비교를 나타내었다. 그림에서 알 수 있듯이, 보통 인버터의 경우 약 16 nsec 정도로 지연시간이 가장 컸고,  $\beta_p/\beta_n$  값이 큰 인버터는 약 11 nsec 정도, 감지

증폭기는 약 6 nsec로 감지 증폭기와 보통 인버터의 속도 차이는 10 nsec인 것을 알 수 있다<sup>12)</sup>.

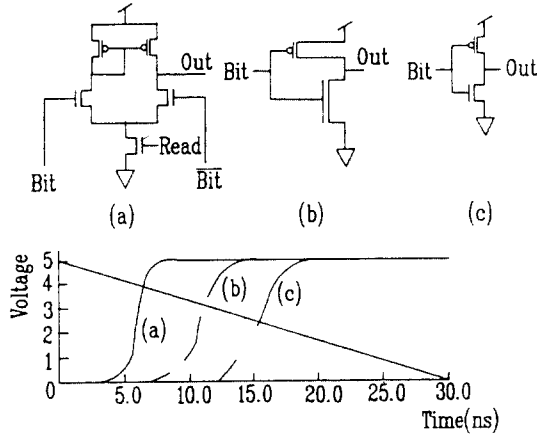


그림 2.5 감지증폭기의 속도 비교  
Fig. 2.5 Speed comparison of Sense amplifiers

2.5. 순차적 어드레스 인코더

CARM에서 매칭된 워드가 여러개인 경우 매칭된 모든 어드레스를 인코딩할 수 있어야 한다. 이때 사용되는 회로가 순차적 어드레스 인코더이다. 이 회로는 매칭 데이터를 우선권에 따라 순차적으로 인코딩 해주는 역할을 한다<sup>13)</sup>.

그림 2.6(a)와 (b)에 순차적 어드레스 인코더의 게이트 회로와 트랜지스터 회로를 각각 나타내었다.

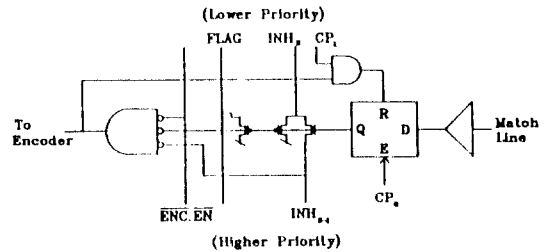
이 회로의 동작은 다음과 같다.

- i) "N" 워드에서 매치가 발생하면, 매치선의 출력 전압은 high가 된다. 이 전압은 인버팅되어 D-래치에 저장된다. Enable pulse CP1은 각 매치 동작의 초기에 온다.
- ii) 이 래치의 출력은 FLAG와 INH<sub>N</sub>을 high로 만들어 준다. INH<sub>N</sub>은 낮은 우선권의 워드 인코딩을 불가능하게 한다.
- iii) 어드레스 인코딩이 가능한 경우는 INH<sub>N-1</sub>과 ENC.EN이 low일때이다. 이것은 높은

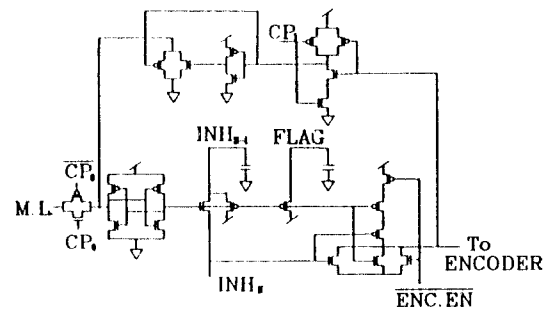
우선권의 매칭된 모든 워드들이 어드레스 인코딩이 끝났음을 의미한다.

iv) "N" 워드에서 어드레스 인코딩이 끝난 후에 신호는 래치로 되먹임되고 각 클럭 사이클(CPO)의 초기에 매치 정보를 리셋시킨다.

이러한 FLAG, INH, ENC.EN은 이 테마이스의 동작속도에 중요한 요소로 작용한다. 본 논문에서는 CARM을 16워드×8비트로 설계하였으므로 그림 2.6과 같은 회로를 계층적 구조에 적용할 필요는 없지만, 메모리 용량이 큰 회로에서는 계층적 구조를 사용하여 인코딩 속도를 증가시키는 등 해야 한다.



(a) 순차적 어드레스 인코더 게이트 회로



(b) 순차적 어드레스 인코더 회로

그림 2.6 순차적 어드레스 인코더 회로도  
Fig. 2.6 Sequential Address Encoder circuit

2.6. 인코더

인코더는 매치 동작시에 순차적 어드레스 인코더를 통해 나온 출력을 받아서, 매칭된 데이터의

어дрес을 칩 외부로 출력시켜주는 역할을 한다. CARM에서 사용한 구조는 기본적인 ROM을 이용하였다<sup>10)</sup>. 회로도에는 그림 2.7에 나타내었다.

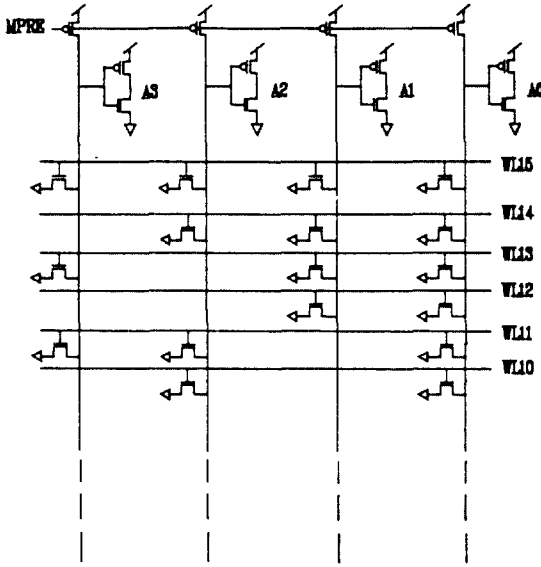


그림 2.7 어드레스 인코더 회로도.  
Fig. 2.7 Address Encoder circuit

### 2.7. 디코더

디코더는 읽기나 저장할 때 필요한 회로로서, 필요한 데이터의 어드레스를 입력으로 받아서, 거기에 해당하는 워드선을 high로 만들어준다. CARM에서는 4-입력 NAND 디코더를 사용하였다<sup>11)</sup>. 회로도에는 그림 2.8에 나타내었다.

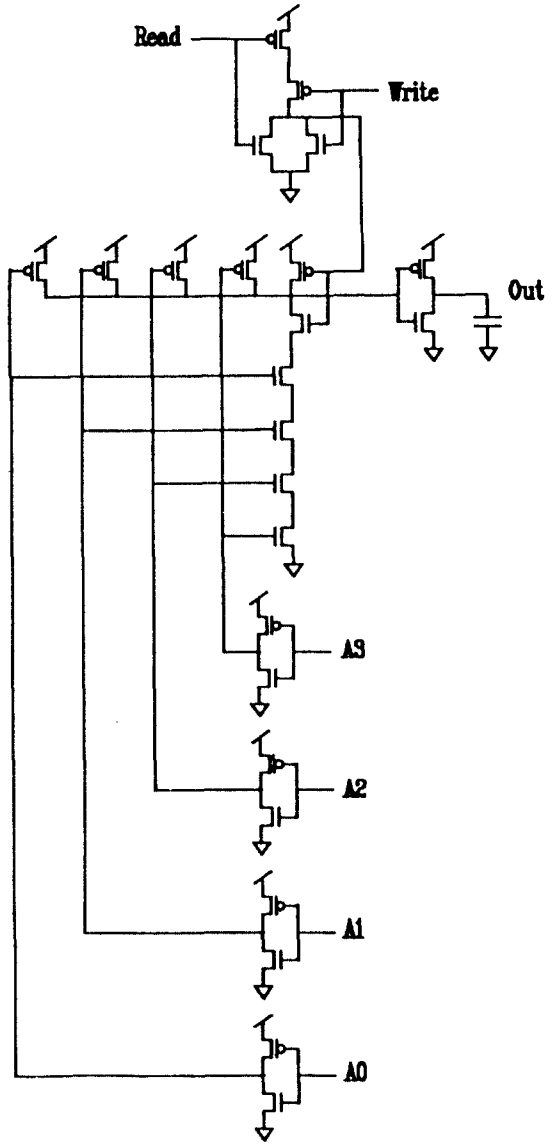


그림 2.8 어드레스 디코더 회로도.  
Fig. 2.8 Address Decoder circuit

### 2.8. 리엔트런트 동작 회로(지연 저장 회로)

리엔트런트 동작중 매칭이 발생하였을때 어드레스 포인터가 가리키고 있던 워드의 내용이 매칭워드로 지연 저장되도록 해주는 회로이다. 지연 저장 기법(Delayed write technique)을 위한 회로는 그림 2.9에 나타내었다. 동작 설명은 다음과 같다.

첫번째 싸이클 동안, 순차적 어드레스 인코더 출력에서 매칭정보는 D-래치에 저장된다. 두번째 싸이클의 시작에서, 최상위 워드(N-1)은 액세스되고 최상위 데이터는 Bit 선으로 읽혀 출력된다. Δ 후에 매치된 워드 "X"는 기억된 매칭 정보에 따라 선택되고 액세스된다. 이때, 두

개의 워드선 전압은 high에 있다. 그러나, Bit 선과 Bit 선의 전압은 저장하기에 충분한 정도로 바뀌어서 저장동작이 수행된다. 즉, Bit 선과 Bit 선의 최상위 데이터는 워드 "X"로 저장된다.

그림 2.10은 지연 저장 기법의 시뮬레이션 결과이다.

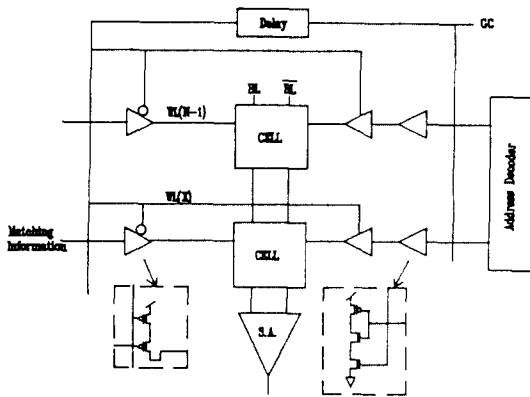
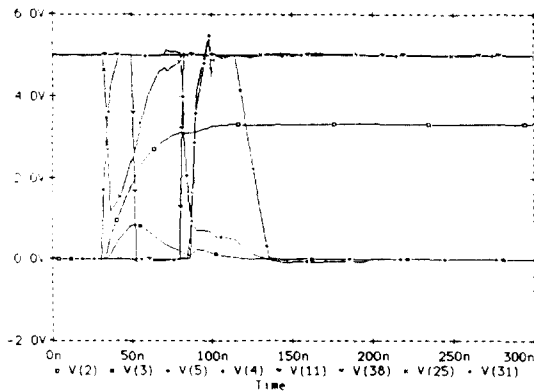


그림 2.9 지연 저장 기법의 회로도.  
Fig. 2.9 Circuit for Delayed write technique



V(4) Word line(N-1)  
V(5) Word line(X)  
V(11) Garbage Enable  
V(25) Word(N-1)  
V(31) Word(X)

그림 2.10 지연 저장 기법의 시뮬레이션  
Fig. 2.10 Simulation of Delayed write technique

### III. CARM의 동작

#### 3.1. 읽기/저장 동작

이 절에서는 읽기, 저장 동작에 대해 기술하였다. 먼저 읽기 동작은 다음과 같은 순서로 이루어진다.

- i) 디코더로 어드레스 신호를 입력시킨다.
- ii) 읽기/저장 제어 회로와 디코더, 감지 증폭기로 읽기 신호를 입력시킨다.
- iii) 어드레스에 해당하는 데이터를 감지 증폭기를 통하여 출력시킨다.

저장 동작은 다음과 같은 순서로 이루어진다.

- i) 데이터/마스크 레지스터로 입력 데이터를 Q를 입력시킨다.
- ii) 읽기/저장 제어회로와 디코더로 저장 신호를 입력시킨다. 디코더로 저장 하려는 워드의 어드레스를 입력시킨다.

#### 3.2. 매치 동작

매치 동작은 다음과 같은 순서를 따른다.

- i) 탐색 데이터를 데이터/마스크 레지스터에 입력시켜준다.
- ii) 매치 신호를 읽기/저장 제어 회로에 입력시킨다.
- iii) 매치 Precharge 신호를 끊어준다.

매치 동작시에 하나의 워드선에 걸려있는 모든 메모리 셀의 매치신은 wired-AND 이다.

#### 3.3. 리엔트런트 동작

리엔트런트 동작은 고속 연상탐색과 데이터의 재기/지정이 필요한 다이내믹 데이터 플로우 컴퓨터의 고속 매칭 유닛을 위해 사용된다<sup>(2)</sup>.

리엔트런트 동작을 설명하기 위한 동작도는 그림 3.1에 나타내었다.

리엔트런트 동작의 동작 설명은 다음과 같다.

##### 1) 매칭이 발생한 경우

첫번째 싸이클 동안, "X" 워드에서 매치가 감지되고, 대응하는 어드레스가 출력된다. 데이터가 비러지고, "X" 위치는 비어있게 된다. 다음

사이클 동안, 이 빈공간을 채우기 위해, 또는 레퍼런스 회수를 수행하기 위해, 어드레스 포인터에 의해 지정된 (N-1) 어드레스인 최상위 데이터는 지연 저장 기법에 의해 비어있는 위치로 대체된다. 이 동작후에, 기억소자의 기록이 끝나고, 채워진 공간은 연속으로 된다. 타이밍도는 그림 3.2(a)에 나타내었다. 데이터 버스에서의 레퍼런스 데이터와 어드레스 버스의 최상위 어드레스 (N-1)은 데이터와 어드레스 레지스터로 입력된다. 데이터가 매치 동작에 사용되고, 어드레스는 디코딩되어서 지연저장 동작에 사용된다.

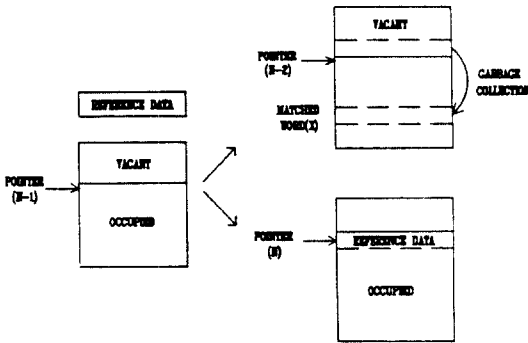
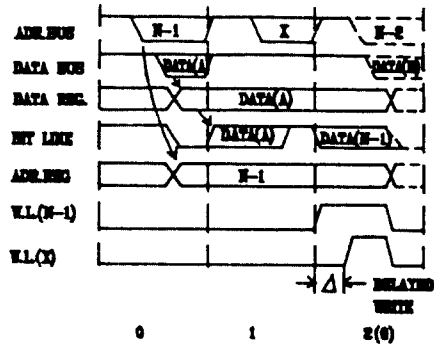
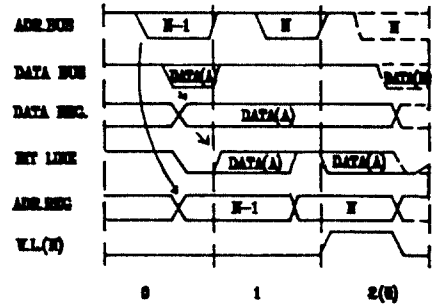


그림 3.1 리엔트런트 동작 설명도  
Fig. 3.1 Description of Reentrant operation



(a) 매칭



(b) 미스매칭

그림 3.2 리엔트런트 동작의 타이밍도  
Fig. 3.2 Timing diagram of Reentrant operation

ii) 매치가 발생하지 않은 경우

첫번째 사이클에서 매치 동작후에, 들어오는 레퍼런스 데이터는 채워있는 부분에 스택처럼 쌓인다. 대응하는 타이밍 도는 그림 3.2 b)에 나타내었다.

첫번째 사이클 동안은 매치인 경우와 동일하다. 그러나, 두번째 사이클에서 매치 플래그가 발생하지 않으면, "N"으로 미리 증가된 어드레스 포인터의 내용은 어드레스 버스로 출력되고, 어드레스 레지스터로 옮겨진다. 다음 사이클에서 어드레스 "N"이 디코딩되고 액세스 된다. 마지막으로, Bit 선에서 레퍼런스 데이터는 "N" 워드로 쓰여진다. 두 가지 모든 경우에 리엔트런트 동작은 두 매치 사이클 동안 효과적인 실행을 구현하

기 위해 파이프 라인될 수 있다.

IV. 시뮬레이션 및 레이아웃

II, III 장에서 설명한 각 회로들에 대해서 논리 시뮬레이션과 Spice 시뮬레이션을 수행하였다. 논리 시뮬레이션은 Apollo 워크스테이션상의 Quicksim을 이용하여 인코더, 디코더, 읽기/저장 제어, 순차적 어드레스 인코더, 데이터/마스크 레지스터 등 각 블록에 대해 수행하였고, 읽기/저장 제어, 데이터/마스크 레지스터, 메모리 셀 등을 붙여서 저장 시뮬레이션을 수행하였다. 읽기 동작은 감지 증폭기가 야날로그 회로이



어서 논리 시뮬레이션은 불가능하므로 회로 시뮬레이션만을 수행하였다.

회로 시뮬레이션은 IBM PC에서 Pspice를 이용하였으며, 메모리 셀, 감지 증폭기, 읽기/저장 제어, 디코더, 인코더, 지연 저장 기법, 순차적 어드레스 인코더, 데이터/마스크 레지스터 등 각 블록별로 시뮬레이션을 하였다. 시뮬레이션 결과는 표 3.1에 나타내었다.

시뮬레이션 결과 읽기, 저장, 배치의 각 동작에 대한 지연은 다음과 같다.

- 읽기 : 26ns(디코더+메모리 셀+감지 증폭기)
- 저장 : 33ns(읽기/저장 제어+데이터/마스크 레지스터+메모리 셀)
- 배치 : 41ns(읽기/저장 제어+데이터/마스크 레지스터+순차적 어드레스 인코더+인코더)

레이아웃은 ETRI 3  $\mu\text{m}$  CMOS n well 공정에 따라 Apollo 워크스테이션 상의 Chipgraph 소프트웨어를 사용하여 각 블록별 레이아웃을 수행하였다.

각 블록별 레이아웃을 수행한 후, 배열시키고 배치 및 배선을 하였다. 결과적인 전체 레이아웃은 그림 4.1과 같다.

칩의 총 면적은  $4986\mu\text{m} \times 4986\mu\text{m}$ 이고, 입력 보호 회로와 출력 버퍼회로를 제외한 내부의 사용한 면적은  $3714\mu\text{m} \times 3714\mu\text{m}$  였다. 여기에서 입력 핀수는 33개, 출력 핀수는 21개로 구성되었다. 또한 테스트를 하기 위하여 각 블록별 회로를 따로 배치하였다.

## V. 결 론

본 논문에서는 고속 병렬처리를 수행하는 기본적인 CAM에 매칭데이터의 순차적인 인코딩을 수행하기 위한 순차적 어드레스 인코더, On chip 페영역 회수를 수행하기 위한 리엔트런트 동작, 어떤 워드에 대해서 부분적으로 배치 여부를 수행하기 위한 마스크 동작 등을 추가한 16워드  $\times$  8비트 Content Addressable and Reentrant

표 3.1. 각 회로의 회로 시뮬레이션 결과

회로 및 동작	지연 시간
메모리 셀	2ns
감지 증폭기	6ns
읽기/저장 제어	
지장	16ns
배치	15ns
읽기	16ns
디코더	18ns
인코더	5ns
순차적 어드레스 인코더	5ns
데이터/마스크 레지스터	15ns

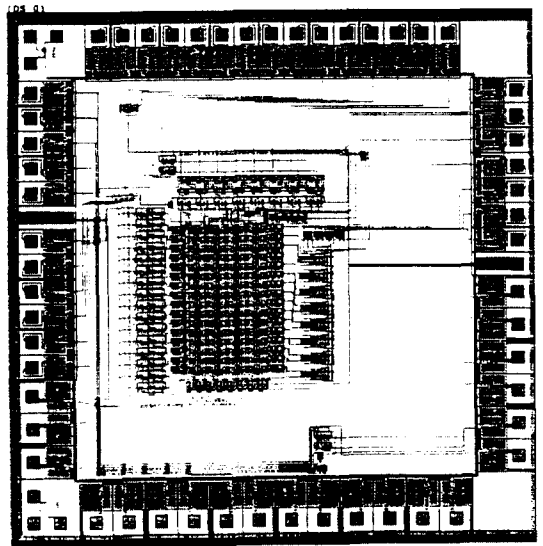


그림 4.1 CARM 레이아웃  
Fig. 4.1 Layout of CARM

Memory(CARM)을 설계하였다.

전체적으로 메모리 셀, 순차적 어드레스 인코더, 읽기/저장 제어, 데이터/마스크 레지스터, 디코더, 인코더, 감지 증폭기, 리엔트런트 동작으로 구성되었으며, 위 회로의 설계가 끝난 후 Apollo 워크스테이션상의 Quicksim에서 논리 시뮬레이션, Pspice를 이용하여 회로 시뮬레이션

을 수행하여 정상동작을 확인하였다. 설계한 CARM의 시뮬레이션 결과 읽기 동작의 경우 26ns, 저장 동작의 경우 33ns, 매치 동작의 경우 41ns로 측정되었다.

그리고, 위의 결과를 바탕으로 3- $\mu$ m CMOS n-well 공정에 따라서 레이아웃을 하였다.

參 考 文 獻

1. T. Kohonen, "Content Addressable Memories", Springer-Verlag, 1987.
2. H. Kadota, "An 8-kbit Content Addressable and Reentrant Memory", IEEE J. Solid state circuits, vol. sc-20, No. 5, pp. 951~957, Oct. 1986.
3. T. Ogura, "A 4-kbit Associative Memory LSI", IEEE J. Solid state circuits vol. sc 20, No. 6, pp. 1,277~1,282, Dec. 1985.
4. J.T. Koo, "Integrated-Circuit Content Addressable Memories", IEEE J. Solid state circuits, vol. sc-5, pp. 208~215, Oct. 1970.
5. L.F. Childs, "An 18ns 4k $\times$ 4 CMOS SRAM", IEEE J. Solid state circuits, vol. sc-19, No. 5, pp. 545~55

- 1, Oct. 1984.
6. S.R. Jones, "A 9-kbit Associative Memory for High Speed Parallel Processing Applications", IEEE J. Solid state circuits, vol. sc-23, No. 2, Apr. 1988.
7. S.I. Chae, "Content Addressable Memory for VLSI Pattern inspection", IEEE J. Solid state circuits, vol. sc 23, No. 1, Feb. 1988.
8. T. Nikaido, "A 1-kbit Associative Memory LSI", Japanese J. Applied Physics, vol. 22, pp. 51-54, 1983.
9. T. Ogura, "A 20-kbit Associative Memory LSI for Artificial Intelligence Machines", IEEE J. Solid state circuits, vol. sc-24, No. 4, Aug. 1989.
10. A.G. Halton, "Content Addressable and Associative Memory System - A survey", IEEE Trans. Electronic Computers, vol. 15, No. 4, Aug. 1966.
11. N. Weste, "Principles of CMOS VLSI Design", Addison Wesley, 1985.
12. M. Annaratone, "Digital CMOS circuit design", Kluwer Academic Publishers, 1986.
13. 차균현 외, "Content Addressable Memory 설계 및 제작에 관한 연구", MPW(1) 최종연구보고서, 한국전자통신연구소, 1989.



李 準 洙 (Jun Soo LEE) 正會員  
 1967年 3月 11日生  
 1989年 2月: 高麗大學校 電子電算工學科 卒業(工學士)  
 1989年~現在: 高麗大學校 大學院 電子工學科 碩士課程  
 ※주관심분야: VLSI Circuit design, CAD tool등임



白 寅 天 (In Cheon PAIK) 正會員  
 1963年 1月 14日生  
 1985年: 高麗大 電子科 卒業  
 1987年: 高麗大 大學院 電子科(碩士)  
 1987年~現在: 高麗大 大學院 電子科 博士課程



朴 商 奉 (Sang Bong PARK) 正會員  
 1962年 3月 8日生  
 1985年: 慶尙대 電子材料科 卒業  
 1987年: 高麗大 大學院 電子工學科 卒業(工學碩士)  
 1987年~現在: 高麗大 大學院 電子工學科 博士課程



朴 魯 京 (Nho Kyung PARK) 正會員  
 1958年 1月 8日生  
 1984年 2月: 高麗大學校 電子工學科 卒業  
 1986年 2月: 高麗大學校 工學碩士學位 取得  
 1990年 2月: 高麗大學校 工學博士學位 取得  
 1988年~現在: 湖西大學校 情報通信工學科 助教授  
 ※主關心分野는 VLSI/CAD, 通信 回路 및 시스템 自動設計 등



車均鉉(Kyun Hyon TCHAH) 正會員

1939年3月26日生

1965年：서울大學校 工學士

1967年：美國伊利諾伊大學校 工學碩士學位 取得

1976年：서울大學校 工學博士學位 取得

1987年～現在：高麗大學校 電子電算工學科 教授

※主關心分野：CAD 및 通信 시스템等