

논문 2004-41SD-10-14

비동기식 시스템을 위한 메모리의 동작 완료 신호 생성 회로

(A Design Method of a Completion Signal Generation Circuit of Memory for Asynchronous System)

서 준 영*, 이 제 훈*, 조 경 록*

(Joon-Young Seo, Je-Hoon Lee, and Kyoung-Rok Cho)

요 약

본 논문은 D-I (delay insensitive) 모델을 사용하는 비동기 프로세서의 메모리 동작 완료 신호 생성 회로를 제안한다. 제안된 설계 방법은 더미셀과 완료 신호 생성 회로를 이용하여 메모리의 읽기 및 쓰기 동작의 완료 신호를 생성한다. 비트라인과 메모리 셀의 지연을 고려하여 메모리를 지수적 블록 크기로 나누어 최소의 완료 신호 회로를 추가하여 D-I 모델로 동작하는 메모리를 설계하였다. 각 구역의 크기가 지수적으로 증가하도록 메모리를 분할하는 제안된 분할 알고리즘은 기존의 동일한 크기를 갖는 구역들로 메모리를 분할하는 방법에 비해 약 40% 정도 동작 지연을 개선하였다.

Abstract

This paper presents a design method for an asynchronous memory with a completion signal generation circuit meeting D-I model. The proposed design method is to generate a completion signal with dummy cell and a completion signal generation circuit to indicate completion of the required read or write operation to the processor. Dividing a memory exponentially to consider delay of a bit-line and a memory cell makes memory operate as a D-I model with minimum addition of redundant circuit. The proposed memory partitioning algorithm that divides entire memory into the several partitions with an exponentially increased size reduces the memory access delay by 40% compared with a conventional partitioning method to the same size.

Keywords: Asynchronous memory, Self-Timed, Completion signal, Delay-insensitive

I. 서 론

최근 VLSI 공정 기술의 발달로 게이트 지연은 감소되나 집적 가능한 회로의 크기가 증가함에 따라 상대적으로 배선 지연이 증가되고 있다. 동기식 시스템의 경우 시스템이 점점 복잡해지고 고속화됨에 따라, 시스템 클럭 신호의 동기화가 어려워지고 클럭 스퀘 등의 문제가 발생된다. 이러한 동기식 회로 설계의 단점을 극복

하기 위해 시스템 클럭 대신 핸드셰이킹 프로토콜을 사용하는 비동기식 회로에 관한 연구가 진행되고 있으며 현재까지 많은 비동기 프로세서들이 개발되었다^[1-2].

동기식 시스템은 서로 다른 데이터패스를 갖는 여러 모듈로 구성되며, 일반적으로 가장 긴 데이터패스를 갖는 모듈의 동작 지연 시간이 시스템 클럭의 기준이 되고 클럭 신호에 따라 전체 모듈이 모두 동시에 구동된다. 그러나 비동기식 시스템은 핸드셰이킹 프로토콜에 의해 동작이 요구되는 모듈만을 구동시키고 동작이 완료된 모듈은 동작 완료를 지시한 후 휴지(idle) 상태에 머물며 이 경우 전력을 소비하지 않는다. 따라서 각 모듈은 각각의 데이터패스의 길이에 따라 서로 다른 동작 시간을 갖기에 전체 시스템은 평균 동작 지연 시간으로 구동되고, 동작이 요구될 때만 모듈을 구동하므로 고속, 저전력 시스템에 유리하다. 그러나 각 모듈의 동작 완

* 정회원, 충북대학교 정보통신공학과, 컴퓨터정보통신연구소

(Dept. of Computer and Communication and Research Institute for Computer and Information Communication, Chungbuk National University)

※ 본 연구는 반도체설계교육센터(IDECE)의 설계 환경 지원과 정통부가 지원하는 ITSOC 사업단의 ITSOC 설계실습사업의 지원으로 수행되었음.

접수일자:2004년6월17일, 수정완료일:2004년 8월16일

료시간을 검출하기 위한 부가적인 회로가 필요하다는 단점을 갖는다.

비동기 회로 설계시 동작 완료 검출을 위해 지연 모델을 사용하며 크게 고정 지연 (bounded delay) 모델과 미지 지연 (delay insensitive, DI) 모델로 구분된다. 고정 지연 모델을 적용할 경우 각 모듈을 합성한 후 레이아웃된 상태에서 정밀한 시뮬레이션을 통해 회로의 지연값을 예측하고 이 지연값을 갖는 지연 소자를 회로에 삽입하여 동작 완료 신호를 생성해야 하며, 미지 지연 모델을 적용할 경우 각 모듈의 동작 지연의 크기와는 무관하게 출력 신호의 유효 여부에 따라 동작 완료를 판단하는 검출 회로를 추가해야 한다.

기존에 개발된 대부분의 비동기 프로세서들은 모듈 내의 데이터패스의 지연값을 미리 알고 있다고 가정하고 고정 지연 모델을 적용하여 비동기식 메모리를 개발하였고 일부만이 회로의 지연값을 예측하는 대신 동작 완료 신호를 생성하는 미지 지연 (delay insensitive, DI) 모델을 적용한 비동기식 메모리를 사용한다. Amulet2e 프로세서의 캐쉬 메모리는 데이터 저장 메모리인 RAM과 주소 지정 메모리인 CAM으로 구성되며 각각은 DI 지연 모델을 적용하여 메모리의 동작 지연과 무관하게 동작하도록 비트의 변화를 감지할 수 있는 더미라인을 추가하여 읽기 및 쓰기 동작의 완료를 검출한다^[3]. Vincent의 비동기식 SRAM은 고정 지연 모델을 적용하여 각 메모리 어레이의 동작 시간을 예측하여 지연값을 결정한다. 그러나 전체 메모리의 워드라인별로 지연값을 결정할 경우 복잡한 예측과정을 거쳐야 하기 때문에 메모리를 몇 개의 구역으로 나누어 각 구역별로 최대 지연 시간을 예측하고 이를 지연소자로 삽입하여 각 구역에 따라 서로 다른 메모리 액세스 시간을 갖게 하였다^[4]. 그러나 본 논문에서 제안한 DI 지연 모델한 완료 신호 검출 회로를 기존 메모리 구조에 적용할 경우 지연과 무관하게 회로 동작에 따라 동작 완료 신호를 검출하기 때문에 메모리 액세스 시간의 예측이 불필요하며 다른 비동기 프로세서와 쉽게 연결될 수 있다.

본 논문에서는 메모리의 각 워드별 비트라인 지연을 고려하여 읽기 및 쓰기 동작의 완료시점을 검출하여 완전한 DI 지연 모델의 비동기식 메모리 설계 방안을 제시하였고 효율적으로 전체 메모리를 구역별로 구분하는 방법을 제안하였다. II장은 비동기식 프로세서의 메모리 접근 방법에 대해서 설명하고 III장은 제안된 메모리 완료 신호 검출 회로와 DI 동작을 위한 메모리 구조를 제시하고 IV장은 전체 메모리의 블록별 지연 시간에 따

른 분할 방안을 제시한다. V장에서 제안된 회로에 대한 실험 결과를 토대로 VI장에서 결론을 맺는다.

II. 비동기식 프로세서의 메모리 접근 방식

1. 비동기 프로세서의 메모리 액세스

비동기식 시스템은 그림 1과 같이 데이터 전송을 위한 전송선외에 추가적으로 요구 신호(Req)와 확인 신호(Ack)의 두 제어 신호를 사용한다. 요구 신호는 동작이 필요한 모듈로 데이터를 전송함과 동시에 송신한 데이터에 대한 구동을 요구하는 제어 신호로 사용되고 수신단은 요구 신호를 수신하면 휴지 상태에서 벗어나 입력 데이터를 받아들여 요구된 동작을 수행한다. 동작이 완료된 후 유효 데이터를 출력하고 송신단으로 동작 완료를 지시하는 확인 신호를 전송함으로써 송신단과 수신단측의 핸드셰이킹을 종료한다^[5].

이처럼 비동기식 시스템은 클럭과 같은 기준 신호가 없기 때문에, 각 모듈별로 연산이 완료된 후 유효 데이터를 출력한 후 이의 상태를 검출하여 다음 블록에 지시할 수 있어야 한다. 비동기식 회로내의 각 지연 요소를 고려한 지연 모델에 따라 동작 완료의 검출 방법과 데이터 전송 방법이 달라진다. 고정 지연 모델을 적용할 경우 그림 1(a)에서 나타난 것처럼 싱글-레일로 데이터를 전송한다. 조합 회로의 동작이 시작됨과 동시에 지연 소자에도 요구 신호가 입력되며 조합 회로의 유효 데이터가 출력과 동시에 지연 소자로부터 동작 완료 신호가 출력된다. 반면에, 그림 1(b)과 같이 DI 지연 모델을 적용한 비동기식 회로는 동작 완료를 판단하기 위해

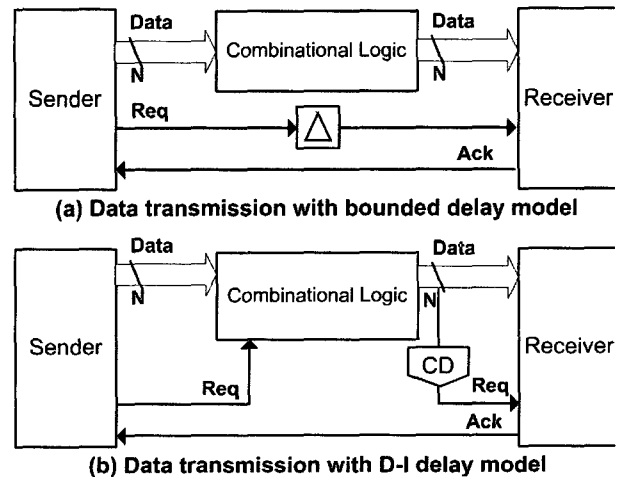


그림 1. 지연 모델에 따른 데이터 전송 방식
Fig. 1. Data transmission method according to the delay model.

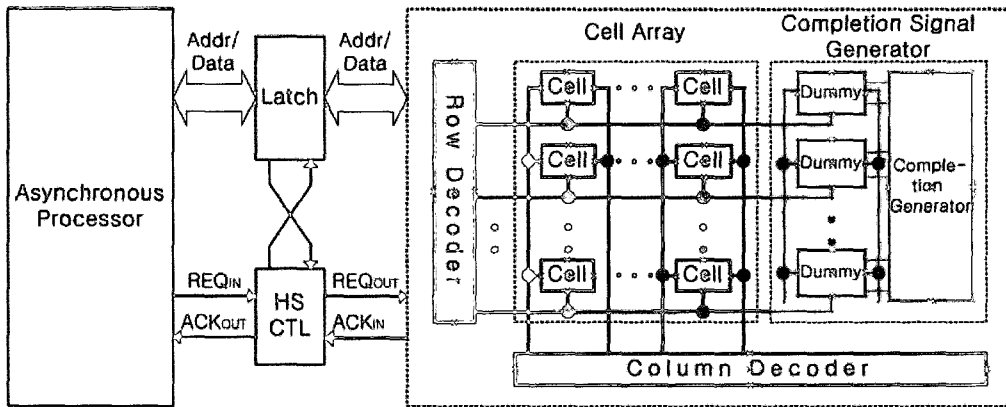


그림 2. 프로세서와 비동기식 메모리의 핸드셰이킹 모델
 Fig. 2. Handshaking model between processor and asynchronous memory.

듀얼 레일 (dual-rail) 데이터 전송과 조합 회로 내부에 동작 완료 검출 회로 (completion detector, CD)가 필요하다.

각 모듈의 동작이 완료된 후 데이터 출력이 유효하면 이를 검출하여 완료 신호를 생성한다. 따라서 데이터패스의 길이에 따른 지연과 무관하게 동작 완료 시점을 검출할 수 있다.

2. 제안된 비동기식 메모리 액세스

DI 지연 모델을 적용한 비동기 프로세서는 내부의 배선 및 소자 지연과 무관하게 자기동기식으로 구동되고 외부 메모리와 같은 주변장치의 연결시에도 정확한 메모리 액세스 시간과 무관하게 동작하도록 DI 지연 모델을 적용하는 것이 효율적이다. 그림 2는 비동기식 프로세서와 DI 지연 모델을 적용한 외부 메모리간의 핸드셰이킹 인터페이스를 나타낸다. 본 논문에서 제안하는 DI 지연 모델을 적용한 메모리는 내부에 완료 신호 생성 회로를 추가하여 읽기 및 쓰기 동작을 수행한 후 즉시 완료신호를 생성하여 핸드셰이킹한다. 더미셀은 워드라인의 가장 긴 지연을 갖는 위치에 배치시켜 더미셀로의 읽기 및 쓰기 동작의 완료는 해당 워드라인의 읽기 및 쓰기 동작의 완료를 보증한다. 더미셀은 유효 데이터가 읽혀지거나 쓰여진 경우 bit와 \overline{bit} 신호선을 통해 서로 논리역인 값을 출력하고 그렇지 않은 경우 동일한 논리값을 갖도록 함으로서 bit와 \overline{bit} 의 두 신호의 XOR 연산의 결과가 1일 경우 읽기 완료를 의미하고 쓰기 동작의 완료는 두 신호선의 XNOR 연산의 결과로 알 수 있다.

메모리의 읽기 동작은 주소 지정된 메모리 셀과 더미셀로부터 데이터를 읽어 메모리 출력단을 통해 프로세서로 전송한다. 더미셀이 메모리 어레이에 비해 긴 지

연을 갖기 때문에 더미셀로부터 유효 데이터가 전송되면 메모리 셀의 읽기 동작이 완료됨을 지시하는 완료 신호를 생성한다. 워드라인의 종단에 위치한 더미셀의 bit와 \overline{bit} 신호의 XOR 연산을 실행하여 1이면 유효 데이터가 출력됨을 의미하므로 읽기 동작 완료 신호를 출력한다. 메모리의 쓰기 동작은 프로세서로부터 수신된 데이터 워드를 메모리 셀로 저장한다. 따라서 동작 완료를 검출하기 위해 각 워드 라인의 가장 긴 배선 지연을 갖는 위치에 더미셀을 배치하고 메모리 셀뿐만 아니라 더미셀에도 입력 데이터를 저장시킨다. 더미셀에 데이터가 쓰기 완료될 경우 해당 메모리 워드 라인에 모두 데이터가 저장되었음을 의미하므로 쓰기 동작 완료 신호를 프로세서로 전송한다.

동작 지연 시간과 무관하게 메모리를 설계하기 위해 각 워드라인별로 더미셀을 배치하여 워드라인별로 메모리 동작 완료 신호를 생성하도록 하였다. 메모리의 비트라인 지연과 무관하게 동작 완료를 검출할 수 있으나 워드라인별로 더미셀을 배치함에 따라 더미셀 수 증가에 의한 회로 크기가 급격히 증가한다는 단점이 있다. 또한 지연소자를 삽입하는 방법은 메모리 회로의 크기는 동기식과 동일하나 동기식과 같이 최대 비트라인 지연에 의해 성능이 제한되기 때문에 메모리 액세스 속도가 크게 감소된다. 본 논문에서는 전체 메모리를 몇 개의 동일한 크기를 갖는 구역으로 나눈 후, 각 구역별 최대 지연값을 반영할 수 있는 위치에만 더미셀을 배치하였다. 그러나 비동기 메모리내의 각 워드라인의 비트라인 지연의 크기가 비트라인 길이에 비례하지 않고 지속적인 증가를 갖기 때문에 기존에 제안된 것과는 달리 분할된 각 구역의 크기는 메모리 셀 위치에 따른 지연 값에 따라 서로 다른 크기를 갖도록 하였다. 따라서 각 워드라인별로 더미셀을 배치하는 방식에 비해 회로 크

기는 크게 감소되고 구역분할을 사용한 기존 방법에 비해 메모리 액세스 시간이 빠른 장점을 갖는다.

III. 제안된 더미셀 및 완료 신호 생성회로

자연과 무관하게 자기동기식으로 동작하는 비동기식 메모리 구현을 위해 메모리로의 읽기 및 쓰기 동작이 완료된 후 이를 프로세서에 지시하는 완료 신호를 생성하는 회로가 필요하다. 본 논문에서 제안한 완료 신호 생성 회로는 읽기 및 쓰기 동작을 수행할 수 있는 더미셀과 각 더미셀의 읽기 및 쓰기 동작의 완료를 검출하고 완료 신호를 생성하는 완료 신호 생성 회로로 구성된다. 더미셀의 동작 완료가 해당 워드라인의 읽기 및 쓰기 동작이 완전한 완료를 보장하기 위해 각 더미셀은 워드라인의 가장 긴 지연을 갖는 위치에 배치시킨다. 그리고 메모리로의 읽기 및 쓰기는 해당 메모리 셀뿐만 아니라 워드 라인의 종단에 연결된 더미셀에서도 수행된다.

따라서 더미셀의 읽기 및 쓰기 동작의 완료 검출은 해당 워드라인의 읽기 및 쓰기 동작이 완료를 의미한다. 완료 검출 회로는 더미셀에 대해 읽기 혹은 쓰기 동작이 수행되었는지를 판단하고 완료되었다면 동작 완료 신호를 생성하여 프로세서에 동작 완료를 알리는 핸드셰이킹 신호를 전송한다.

1. 제안된 더미셀 구조

그림 2에서 제안된 비동기식 메모리는 기존의 메모리 구조를 사용하며 메모리 셀의 각 워드라인이나 구역별로 더미셀과 완료 신호 생성 회로만을 추가한다. 그림 3은 기존의 6-Tr SRAM 셀을 나타낸다. 기존의 셀은 비트 신호 (Bit와 \overline{Bit})와 워드신호에 의해 동작하며 워드신호가 high가 되면 선택된 워드의 셀 내부에 저장된 데이터가 비트라인을 통해 셀 외부로 읽혀지거나 외부로부터 전송된 데이터가 셀 내부에 저장된다. 따라서 비동기식 메모리로의 읽기 혹은 쓰기 동작이 요구될 경우, 주소 지정된 메모리 워드라인뿐만 아니라 종단에 연결된 더미셀로도 읽기 및 쓰기 동작이 동시에 수행된다. 더미셀로의 읽기 및 쓰기 동작을 위해 읽기 동작이 시작되기 전에 셀에 읽을 데이터를 미리 써놓아야 하고, 쓰기 동작 전에 셀에 이미 저장된 데이터를 미리 지울 수 있어야 한다. 따라서 제안된 더미셀 회로는 그림 4와 같이 PW (Pre_Writing) 회로와 PR (Pre_Reset) 회로를 포함한다. PW 회로는 읽기 완료 상태를 검출하기

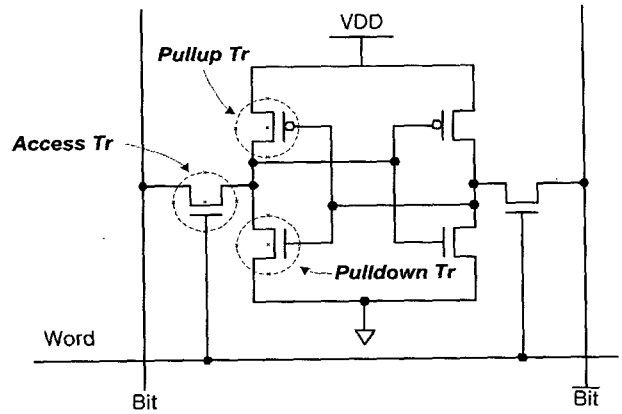


그림 3. 일반적인 SRAM Cell 구조
Fig. 3. General architecture of SRAM Cell.

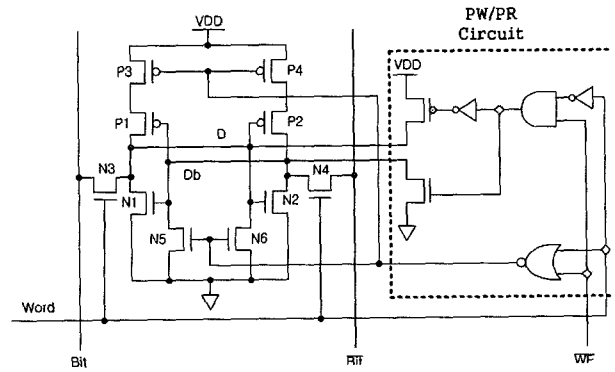


그림 4. 제안된 더미셀과 PW (Pre_Write) 및 PR (Pre_Reset) 회로
Fig. 4. Proposed dummy cell architecture including the PW(Pre_Write) and PR(Pre_Reset) circuit.

위하여 읽기 동작 이전에 '1'의 데이터를 메모리 셀에 미리 저장하도록 하고, PR 회로는 쓰기 완료 상태를 검출하기 위하여 쓰기 동작 이전에 셀 내부에 저장되어 있던 데이터를 미리 리셋한다.

완료 신호 생성 회로는 읽기 동작의 경우 더미셀의 출력을, 쓰기 동작의 경우 더미셀의 내부 상태값을 검사한다. 더미셀은 동작이 완료되지 않은 경우 bit와 \overline{bit} 신호선을 통해 모두 '0'의 값을 출력하고 동작이 완료되면 bit와 \overline{bit} 신호선에 논리역인 유효 데이터가 출력된다. 따라서 더미셀과 연결된 완료 신호 생성 회로는 읽기 동작일 때는 더미셀의 출력 비트인 bit와 \overline{bit} 신호선이 서로 다른 논리값을 전송하면 완료 신호를 생성하고, 쓰기 동작의 경우에 비트라인을 통해 더미셀로 입력되는 비트와 그의 논리역인 비트가 더미셀 내부에 저장된 전위와 일치할 경우 완료신호를 생성한다.

처음에 쓰기 동작신호(\overline{WE})가 high 상태이고 워드신호가 low일 때, 더미셀 내부에는 '1'의 데이터가 저장되어 있다. 쓰기 동작신호가 low가 되고, 워드신호가 low

면 더미셀 내부는 N5, N6가 온이 되어 D, Db의 전위가 '0'으로 리셋된다. 이 때 워드신호가 high가 되면 N5, N6는 오프가 되고 P3, P4가 온이 되어 리셋 동작은 중지되고 더미셀은 기존 SRAM셀처럼 동작하여 비트라인을 통해 전송된 데이터가 셀 내부에 쓰여진다. 셀 내부로의 데이터 전송 후 워드신호가 low가 되고, 쓰기 동작신호가 high가 되면 더미셀의 내부 전위는 D, Db가 각각 전원전압과 그라운드로 연결되어 데이터 '1'을 저장한 상태로 회복된다. 읽기 동작의 경우에는 워드 신호만 high가 되어 셀 내부에 데이터 '1'을 저장하기 위한 미리 쓰기 동작은 중지되고 쓰기 동작에서와 같이 기존 SRAM셀과 같은 동작을 수행하여 저장되어 있던 '1'의 데이터가 비트라인을 통하여 셀 외부로 전송된다.

2. 완료신호 생성회로 및 검출 방법

그림 5와 같이 구성된 더미셀과 완료신호 검출회로를 사용하여 동작의 완료 시점을 검출할 수 있다. 더미셀 옆은 쓰기 완료신호의 생성 오류를 방지하기 위해 읽기 및 쓰기 동작 이전에 방전되며, 더미셀에 연결된 읽기 및 쓰기 완료 신호 생성회로를 통해 각 동작의 완료를 검출하고 완료신호를 생성한다. 입력 비트신호(Di와 Dib)와 더미셀 내부 비트신호(D와 Db)를 감지 증폭기를 통해 증폭하여 완료신호 생성을 위해 사용한다. 워드라인의 종단에 위치한 더미셀은 읽기 동작 이전에 저장되어 있던 '1'의 데이터를 가지고 있다. 읽기 동작이 실행되면 더미셀의 액세스 트랜지스터가 활성화되어 더미셀에 저장되어 있던 비트신호(D, Db)가 비트라인을 통해 쓰기 완료 신호 생성 회로로 전송된다. 그림 6은 읽기 완료 신호 생성 회로이며 더미셀로부터 전

송받은 비트신호를 증폭한 비트 신호, SB와 이의 반전 신호, SBb를 비교하여 두 신호가 일치하지 않으면 읽기 동작의 완료로 판단, 완료 신호를 생성한다.

쓰기 동작의 경우에 더미셀은 쓰기 동작 이전에 low로 리셋 되어있다. 쓰기 동작이 실행되면 비트라인에 전송된 비트신호(bit, \bar{bit})의 전위가 내부 전위 상태가 모두 low로 리셋 되어 있는 더미셀의 내부에 나타나게 되고 (D, Db) 더미셀 내부와 연결된 감지 증폭기에 의해 증폭된다. 그림 7은 쓰기 완료 신호 생성 회로이며 증폭된 더미셀 내부의 데이터 신호(SD, SDb)와 비트라인으로 전송되는 비트신호(Di, Dib)를 증폭한 신호(Si, Sib)를 비교하여 만약, 이 두 데이터가 서로 일치할 경우 쓰기 동작의 완료로 판단, 이의 완료 신호를 생성한다.

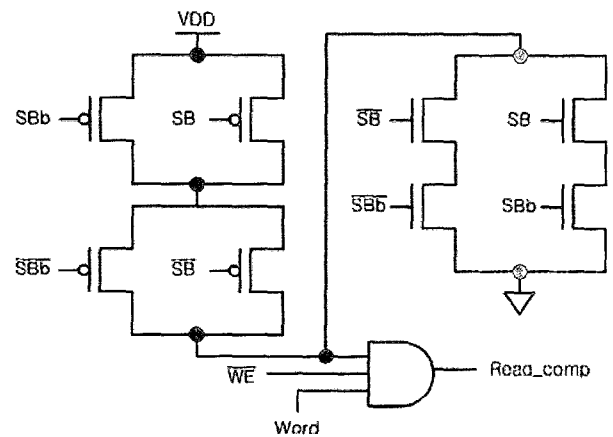


그림 6. XOR 연산을 이용한 읽기 완료 생성 회로
Fig. 6. Read completion signal generator using the XOR operation.

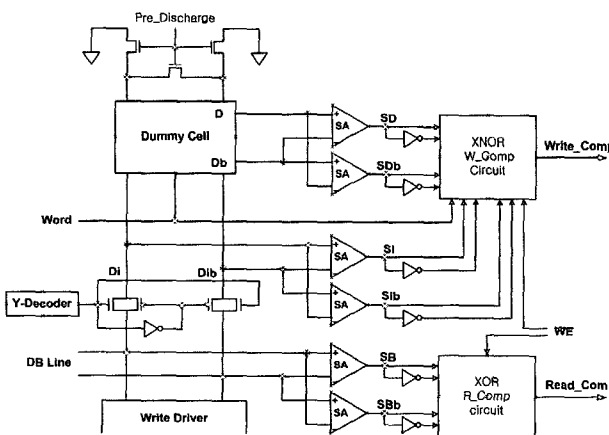


그림 5. 더미셀과 읽기/쓰기 완료 신호 생성회로
Fig. 5. The connection of between the dummy cell and the read/write completion signal generator.

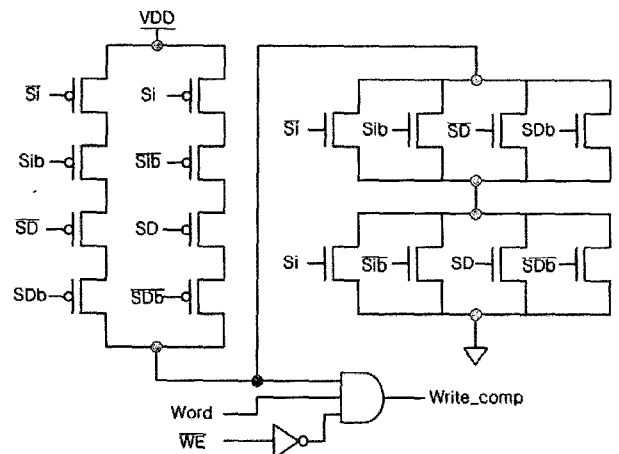


그림 7. XNOR 연산을 이용한 쓰기 완료 생성 회로
Fig. 7. Write completion signal generator using the XNOR.

IV. 더미셀 배치에 따른 완료 신호 검출

더미셀 구조뿐만 아니라 더미셀의 다양한 배치를 통한 효율적이고 빠른 완료 신호 검출이 가능하다. 지금까지의 더미셀의 배치는 전체 메모리 어레이에 하나의 더미셀과 완료신호 생성회로를 두어 메모리의 최대 비트라인 지연을 기준으로 완료신호를 생성하거나 각 워드라인별로 더미셀을 배치하여 워드라인별로 서로 다른 지연 시간에 따라 완료신호를 생성하는 방법을 제시하였다. 그러나 전체 메모리 모듈에 대해 하나의 최대 비트라인 지연값을 이용할 경우 더미셀 및 완료 신호 검출회로를 하나만 사용하기에 회로 크기의 증가 없이 비동기식 메모리 구현이 가능하나 그림 8(a)에서와 같이 각 메모리 셀의 위치에 따라 비트라인 지연시간이 다름에도 불구하고 메모리 액세스 속도는 최대 비트라인 지연 시간에 의해 제한되는 단점을 갖는다. 또한 각 워드라인별로 더미셀을 배치하여 완료 신호를 검출하는 방법은 그림 8(b)과 같이 각 워드라인별 비트라인 지연만이 소요되기 때문에 빠른 완료 신호 검출이 가능하나 각 워드라인별로 모두 더미셀과 완료신호 생성회로가 필요하므로 회로 크기가 크게 증가한다.

따라서 실제적인 비동기 메모리 설계시 전체 메모리 모듈을 여러 구역으로 나누어 각 구역별로 더미셀과 완료신호 생성회로를 연결하는 방법을 사용한다. Vincent의 비동기식 SRAM 설계는 동일 크기를 갖는 메모리를 여러 개의 구역으로 나눌 수 있는 방법을 모델링하여 메모리 액세스 속도와 회로 크기의 최적값을 구하였다.

비트라인 지연은 비트라인상에 존재하는 부하에 의해 발생하므로 비트라인에 연결된 워드라인의 수에 따라 지연값이 증가된다. 따라서 Vincent의 동일한 크기를 갖는 구역들로 전체 메모리를 분할하는 방법에 비해 실제적인 메모리 셀의 각 비트라인의 지연을 예측하여 그림 8(d)와 같이 서로 다른 크기를 갖는 여러 구역으로 전체 메모리를 분할하는 것이 회로의 크기면에서 뿐만 아니라 메모리 액세스 속도면에서 유리하다.

본 논문에서 제안된 메모리 구역 분할은 메모리를 여러 구역으로 분할할 때 각 구역에 배치된 더미셀에 의해 생성된 완료신호는 비트라인 지연의 증가 형태와는 상관없이 구역내 최대 비트라인 지연을 반영한다. 따라서 각 워드라인별로 더미셀을 배치하는 것에 비해 적은 더미셀이 필요하며, 각 구역은 서로 상이한 크기의 메모리 어레이를 포함하나 동일한 비트라인 지연시간을 갖는다. 동일한 비트라인 지연시간을 갖는 구역을 결정하기 위해 메모리 어레이의 비트라인의 부하를 Elmore 지연에서와^[6] 같이 R-C 연결선으로 모델링하고 시뮬레이션을 통해 계산된 지연을 비트라인 지연에 적용하였으며, 더미셀을 이용해 완료 신호를 검출하여 제안된 회로의 성능을 분석하였다. R-C 연결선 모델에 적용된 각각의 파라미터 값은 설계된 셀을 0.35 μ m CMOS 공정에서 레이아웃 후 비트라인 상에서 추출하여 적용하였다. 완료신호 지연 측정을 위해 R-C 연결선 모델로 각 셀의 비트라인을 구성하고 이를 어레이로 구성하여 비트라인 지연을 측정하였으며, 설계된 메모리 셀과 완료신호 생성회로에서의 동작시간을 반영하였다. 쓰기 완

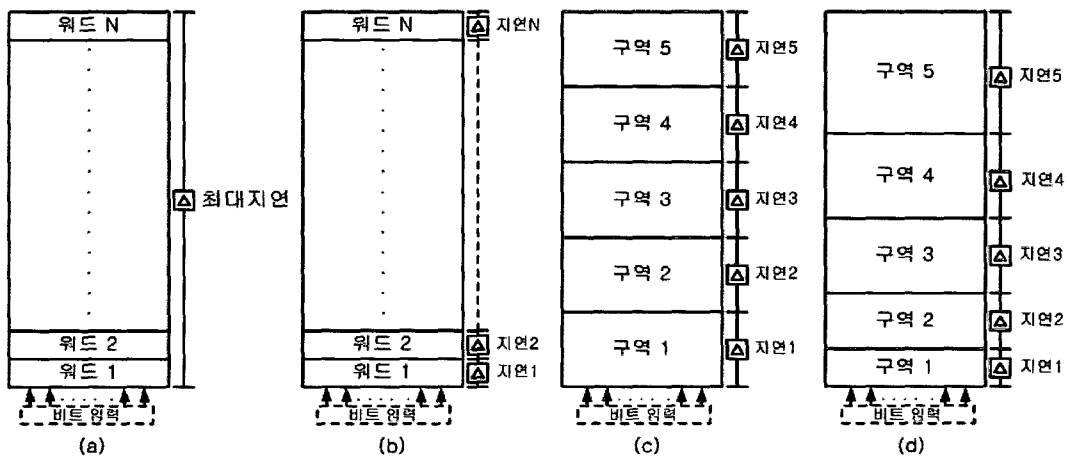


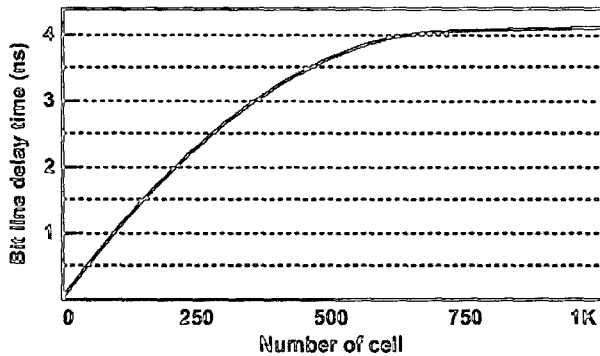
그림 8. 더미셀 배치에 따른 메모리 블록상의 비트라인 지연 모델 (a) 최대 지연 적용, (b) 모든 워드에 더미셀 배치, (c) 고정 구역 분할, (d) 지연시간에 따른 구역 분할시.

Fig. 8. Bitline delay model on the memory block according to the (a) maximum delay model, (b) dummy cell located on the every wordline, (c) dummy cell located the every block with fixed length (d) dummy cell located the every block with variable length.

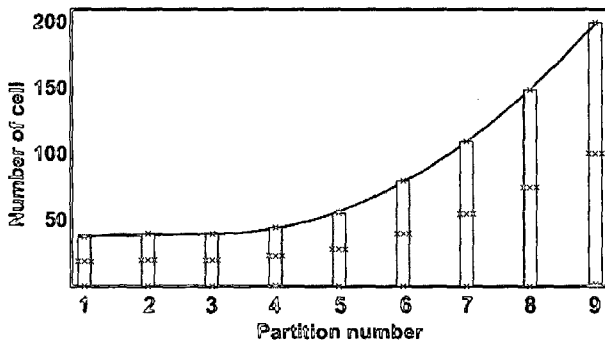
료신호 지연(T_{WC})은 식 6.1과 같이 계산된다. T_{BL} 은 비트라인 지연시간을, T_{WR} 은 쓰기 동작 시간을 그리고 T_{WCG} 는 쓰기 완료신호 생성회로의 동작시간을 의미한다.

$$T_{WC} = T_{BL} + T_{WR} + T_{WCG} \quad (6.1)$$

일례로 1Kbyte 셀 어레이에 대해 Elmore 지연 모델을 적용한 비트라인의 실험 결과와 실제 회로를 합성한 후 netlist를 추출한 실험값을 토대로 비트라인 상의 지연을 추출하였고 셀 위치에 따른 지연값의 크기는 그림 9(a)와 같이 도출하였다. 비트라인 지연 시간은 로그함수 형태로 셀 크기 증가에 따라 지연 시간 증가량이 점차 줄어들고 결국 일정해진다. 따라서 메모리를 여러 구역으로 분할할 때 각 구역의 최대 지연 시간이 같게 할 경우 각 구역은 그림 9(b)와 같이 지수함수적인 증가를 갖는 서로 다른 크기의 메모리 어레이를 갖는 여러 구역으로 나뉘게 된다. 그림 9의 경우 1KB의 메모리를 총 9개의 구역으로 분할하였고 각각은 최대 0.5ns의 메모리 액세스 지연시간을 갖는다. 또한 각 구역별 메모리의 크기는 비트라인의 위치에 따른 지연 시간 그



(a) Bit line delay time according to the number of cells



(b) the size of the each partition

그림 9. 1Kbyte 셀 어레이의 비트라인 지연에 따른 구역 분할 및 구역별 크기 비교
Fig. 9. Block size comparison according to the bitline delay in 1Kbyte memory cell array.

래프의 역함수를 이용하여 그림 9(b)와 같이 결정된다. 따라서 본 논문에서 제안한 것처럼 간단한 비트라인 지연 시간 모델링을 통해 동일한 최대 지연 시간을 갖고 서로 다른 크기를 갖는 여러 구역으로 메모리를 분할하여 각 구역별로 더미셀과 완료 신호 생성 회로를 삽입하는 방식이 기존의 비동기식 메모리에 비해 전체 회로의 크기와 메모리 액세스 시간면에서 효과적이다.

V. 시뮬레이션 결과

제안된 비동기식 메모리 시스템은 그림 10과 같이 메모리 블록을 구성하였다^[7]. 본 논문에서 제안된 더미셀을 이용하여 지역 메모리 설계에 적용 가능한 메모리 어레이를 설계하였다. 비트라인에 따라 한 워드씩 연결하여 메모리 어레이를 구성하고, 제안된 분할 방법을 적용하여 더미셀을 배치하였다. 구성된 어레이의 크기는 각각 2Mbyte, 8Mbyte, 32Mbyte, 128Mbyte, 512Mbyte, 2Gbyte로 구성하였다. 비교를 위해 고정된 크기의 구역을 갖는 어레이와 제안된 구역분할법에 따라 분할한 메모리 어레이를 구성하여 더미셀 배치에 따른 쓰기 완료신호에 대한 지연을 측정하였다. 표 1은 메모리 어레이

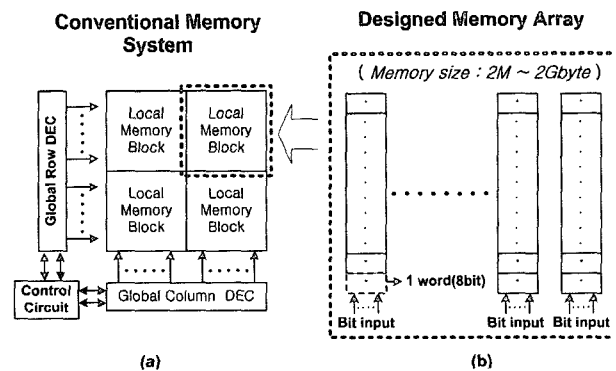


그림 10. 제안된 비동기식 메모리 시스템
Fig 10. Proposed asynchronous memory system.

표 1. 고정 구역분할 및 모든 워드에 더미셀 배치 시 쓰기 완료신호 지연시간

Table 1. Write completion signal delay time in case of both fixed segment division and arrangement of dummy cell on all word.

메모리 크기	최대지연 적용	모든 워드에 더미셀 배치	고정 구역분할 시				단위
			2구역	4구역	8구역	10구역	
2M	56.62	31.5	47.1	39.3	35.6	35.1	ns
8M	220.5	125	183	153	140	137	
32M	910.9	505	736	618	562	551	
128M	3.373	1.84	2.72	2.26	2.05	2.01	us
512M	10.91	5.75	8.42	7.09	6.42	6.29	
2G	43.49	22.8	33.5	28.2	25.5	25.0	

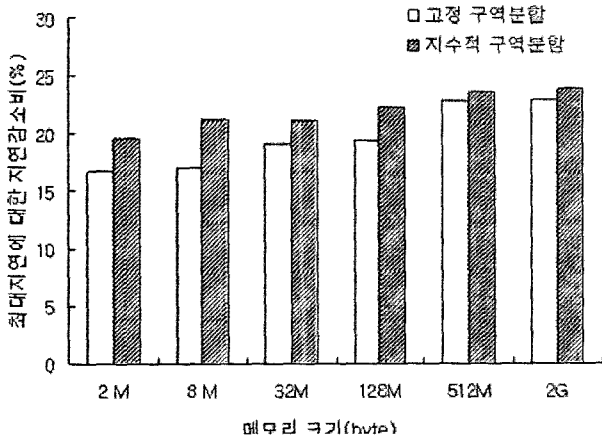


그림 11. 각 분할방법으로 2-구역 분할시 지연감소율.
Fig. 11. Delay reduction rates on two memories divided into two segments with each partitioning method.

로의 비트입력을 기준으로 최대 지연을 적용한 경우와 모든 워드에 더미셀을 배치한 경우, 그리고 고정 구역 분할방법을 적용한 경우에 각각의 쓰기 완료신호에 대한 지연시간을 나타낸다.

성능 측정결과 모든 워드에 더미셀을 배치한 경우, 최대 지연에 대해 지연시간이 메모리 크기별 평균, 44.7% 정도 감소되었다. 그림 11에서는 각각의 구역분할 방법을 이용하여 2개 구역으로 설계한 후 측정된 각 쓰기 완료신호 지연을 비교하여 최대 지연을 적용한 경우에 대한 지연 감소비로 나타냈다. 고정 구역분할의 경우에 평균, 19.4% 정도의 지연 감소를 나타냈고 지수적 구역분할은 최대 지연에 대해 약 2.1% 정도 더 높은 지연 감소를 보였다. 그림 12는 각 분할방법을 적용하여 각각 10개 구역으로 메모리를 분할한 경우와 메모리 어레이 내 모든 워드에 더미셀을 배치한 경우에 대한 지연 감소비를 나타낸다. 고정 구역분할의 경우 평균 39.7% 정도의 지연 감소를 보이며 그림 11의 2개 구역으로 분할한 경우와 비교해 약 2 배 정도 더 높다. 위의 결과에서 볼 수 있듯이 지연시간은 분할된 구역의 수가 증가됨에 따라 감소되고 지수적 구역분할의 경우 더 높은 감소율을 나타낸다. 그러나 분할된 구역의 수가 증가할수록 두 분할방법간의 격차는 감소하며, 이는 분할된 구역의 수가 증가되면 더 많은 구역의 지연을 고려하게 되고, 두 분할방법 간에 분할된 구역 간 경계의 위치가 근접해가기 때문이다. 표 1에 나타난 바와 같이 더미셀 배치에 의한 지연 시간의 감소는 메모리 어레이 내에 더미셀의 배치 빈도, 즉 분할된 구역의 수에 따라 증가하게 되며 가장 적은 지연시간을 나타내는

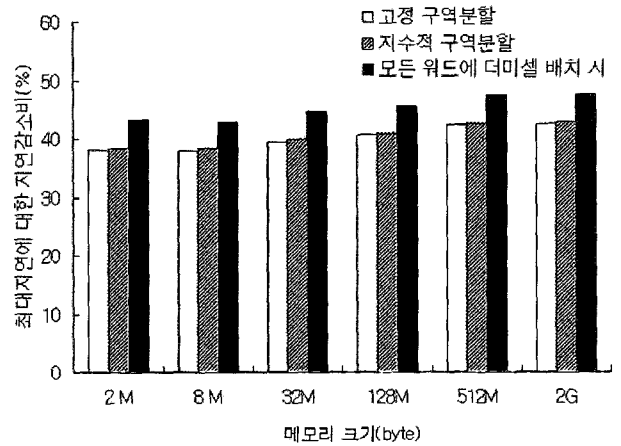


그림 12. 각 구역분할 방법을 적용한 10-구역 분할시 및 모든 워드에 더미셀 배치시 지연 감소비.
Fig. 12. Delay reduction rates on three memories divided into ten segments with each partitioning method and dummycell arrangement on every wordline.

경우는 모든 워드에 더미셀을 배치한 경우이다. 또한 제시된 시뮬레이션 결과 이외에 메모리 블록을 100개 구역까지 분할하여 실험하였다. 100개 구역으로 분할시 모든 워드에 더미셀을 배치한 경우와 거의 동일한 지연 감소를 보이나 10번째 구역 이후부터 2개 구역에서 10개 구역까지 분할한 경우와 같이 지연 감소가 크게 향상되지는 않는다. 따라서 지연감소에 대해서만 최상의 분할 구역 수를 정의하는 것은 매우 어렵고 설계할 메모리 어레이의 크기와 지연 시간을 함께 고려하여 적정 분할 구역수에 따라 더미셀 배치를 결정해야 한다. 더미셀 배치에 따른 면적의 증가는 모든 메모리 블록의 모든 워드에 배치할 경우, 200%에 가까운 증가를 보이나 10구역 분할 시에는 약 0.5% 미만이다.

VI. 결 론

본 논문에서는 더미셀과 완료신호 생성회로를 사용하여 지연소자의 사용 없이 완료신호를 생성하는 방법을 제안하였다. 또한 지수적 구역분할 방법을 적용시 비트라인 지연 감소를 통해 완료신호 생성에 대한 동작 지연을 감소시킴으로써 높은 성능향상을 얻을 수 있었다. 각 워드들에 대해 개별적으로 비트라인 지연을 적용하거나 메모리를 몇 개의 구역으로 분할하고 분할된 각 구역에 대해 블록의 대표 비트라인 지연을 적용하여 메모리를 설계할 경우, 비트라인 지연을 크게 감소시킬 수 있다. 본 논문에서 제안하는 지수적 구역분할 방법을 사용하여 2 M~2 Gbyte 어레이를 설계한 후 읽기

완료 신호와 각 구역별로 비트라인 지연이 적용된 쓰기 완료신호를 검출하고 시뮬레이션을 통해 제안된 회로의 성능을 비교, 분석 및 검증하였다. 분석결과, 메모리 어레이 내 모든 워드에 더미셀을 배치하여 메모리 어레이를 설계 했을 경우 각 워드의 개별적인 비트라인 지연만이 적용되어 최대 비트라인 지연을 고려하여 설계한 경우에 대해 쓰기 완료신호 지연시간이 메모리 크기별 평균, 약 45% 정도 감소되며, 제안된 더미셀 배치 방법 중 가장 높은 감소를 보였다. 또한 메모리 어레이를 10개 구역으로 분할했을 때 고정 구역분할을 적용한 경우에 메모리 크기별 평균, 약 40% 정도의 지연감소를 나타냈고 지수적 구역분할의 경우, 더 높은 지연감소를 나타냄을 확인하였다. 본문에서 제안된 더미셀 및 완료 신호 생성회로와 지수적 구역분할 방법을 사용하여 설계한 비동기식 메모리는 메모리 분할 수에 따른 더미셀과 동작 완료 신호 생성 회로만을 추가하여 완전한 미지지연 (DI) 모델로 동작한다.

[2] A. Takamura et al., "TITAC-2: A 32-bit asynchronous microprocessor based on scalable delay-insensitive model", in Proc. ICCD'97, pp. 288-294, Oct. 1997.

[3] J. D. Garside, S. Temple and R. Mehra., "The AMULET2e cache system", in Proc. Advanced Research in Asynchronous Circuits and systems, pp. 208-217, Mar. 1996.

[4] Vincent Wing-Yun Sit, Chui-Sing Choy, and Cheong-Fat Chan., "A four-phase handshaking asynchronous static RAM design for self-timed systems", IEEE J. Solid-State Circuits, vol. 34, pp. 90-96, Jan. 1999.

[5] Seteve Furber, Principle of Asynchronous Circuit Design, Kluwer Academic Publishers, 2001.

[6] 김석윤, VLSI 시스템 회로연결선의 모형화 및 해석, 시그마프레스, 1999.

[7] Betty Prince, SEMICONDUCTOR MEMORIES, John Wiley & Sons, 1991.

참 고 문 헌

[1] A. J. Martin, A. Lines et al., "The design of an asynchronous MIPS R3000 microprocessor", in Proc. Advanced Research in VLSI, pp. 164-181, Sep. 1997.

저 자 소 개



서 준 영(정회원)
 2000년 청주대학교
 전기전자공학부 학사.
 2003년 충북대학교 정보통신
 공학과 석사과정수료.
 <주관심분야: 반도체 메모리설계, 비
 동기식 시스템을 위한 제어회로 설계>



이 제 훈(정회원)
 1998년 충북대학교
 정보통신공학과 학사
 2001년 충북대학교
 정보통신공학과 석사
 2003년 충북대학교 정보통신
 공학과 박사과정수료.

<주관심분야: 고속 마이크로프로세서 설계, 저전
 력 디지털 회로 설계>



조 경 록(정회원)
 1977년 경북대학교 전자공학과
 공학사.
 1989년 일본 동경대학교
 전자공학과 공학석사
 1992년 일본 동경대학교
 전자공학과 공학박사.

1979년~1986년 (주)금성사 TV연구소 선임연구원
 1992년~현재 충북대학교 정보통신공학과 부교수
 <주관심분야: VLSI 시스템설계, 통신 시스템용
 LSI 개발, 고속 마이크로프로세서 설계>