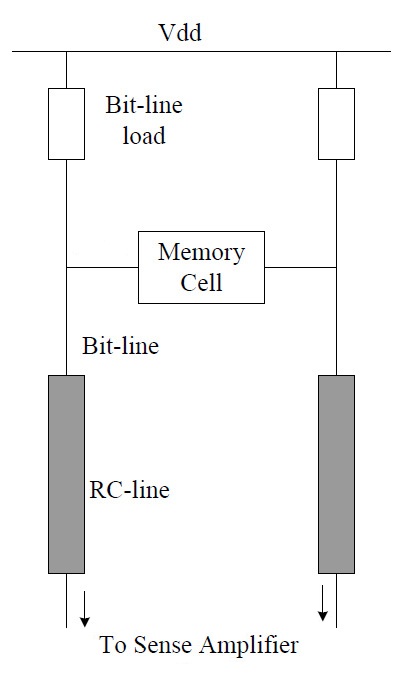
디지털회로설계 HW#4

DRAM, SRAM과 ROM

20161453 전자공학과

김규래

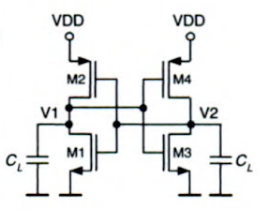
모든 정류의 메모리 반도체들은 <Figure 1>과 같은 기본적인 구조를 갖고 있다. 데이터를 저장하는 동작에 필요한 전원을 공급하는 Vdd, ‘Memory Cell’, 메모리의 출력을 증폭하는 ‘Sense Amplifier’, 데이터가 입력되는 ‘Bit Lines’, 데이터의 참조 위치를 정하는 ‘Address Lines’. 이러한 기본적인 구조는 Core Memory 시절부터 이어지고 있다.

ROM을 제외한 메모리들은 대부분 이러한 구조로 이루어져 있다. DRAM, SRAM 간의 큰 차이는 Memory Cell 구현 방법이다.

Figure

1. Sense Amplifier

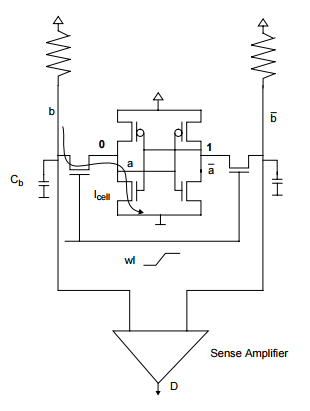
일반적으로 SRAM의 메모리 셀이나 DRAM의 메모리셀에서 출력된 신호는 지나치게 낮은 경우가 많다. 이를 정상적인 논리회로 레벨로 높이기 위해서 Sense Amplifier를 사용한다.

 각 Bit Line의 끝에는 Sense Amplifier가 연결돼 있으며 Bit에서 읽혀진 신호를 Sense Amplifier가 증폭한다. Sense Amplifier 의 증폭 방법에는 크게 두 가지가 있으며 하나는 Voltage Sense랑 Current Sense가 있다.

Figure

좌측의 <Figure 2>는 DRAM과 SRAM에서 공통적으로 가장 많이 쓰이는 CMOS Voltage Sensing Amplifier 회로이다[[1]](#footnote-1). Voltage Sensing 보다 복잡하지만 Current Sensing 방식도 존재하는데, Voltage Sensing보다 속도가 17~20% 빠르다는 장점을 가지고 있다[[2]](#footnote-2).

1. SRAM의 메모리 셀

SRAM의 단일 메모리 셀은 두개의 인버터 회로로 루프를 구성한 형태를 이루고 있다.

Figure

SRAM의 셀에서는 <Figure 3>에서 의 상태에 대해서 오직 또는 일 때만 안정적인 상태를 가질 수 있다.

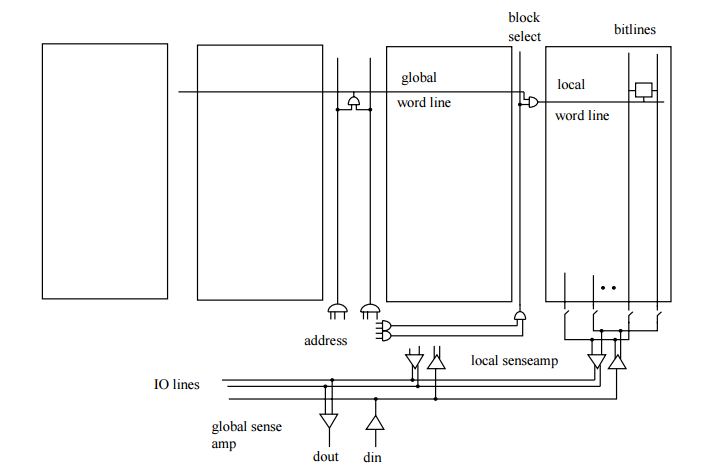
Read 나 Write를 할 때는 wl 워드라인에 신호가 생기면서 비트라인으로 신호가 통하거나 비트라인에서 신호가 셀로 흘러 들어간다.

<Figure 4>는 SRAM의 셀 레이아웃의

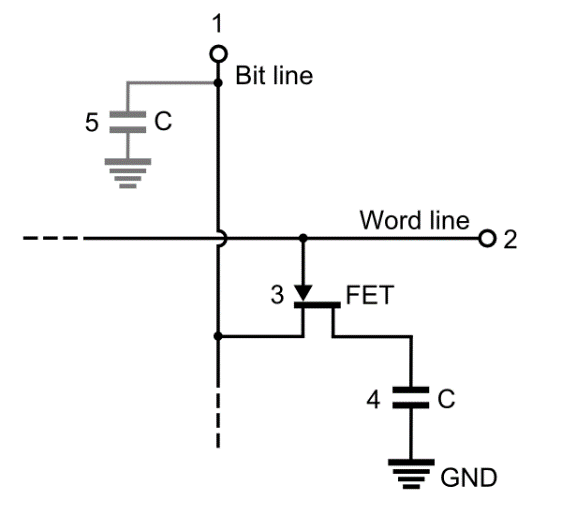
모습이다. 레이아웃에는 크게 3가지 변수들이 있는데 바로 공간, 속도, 전력 소모

량이다. 속도를 최대화하기 위해서는 메모리 블록을 짧게 설계하고, 전력 소모량

을 줄이기 위해서는 메모리 블록을 길고 좁게 설계해야 한다.



Figure

1. DRAM 메모리셀

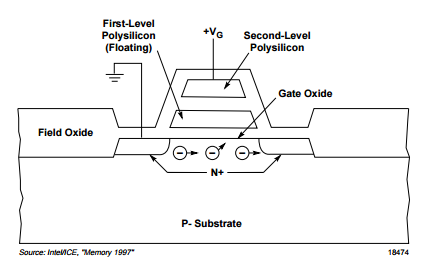
Figure

<Figure 5>는 DRAM 메모리셀의 모습이다. DRAM 에서는 트랜지스터 하나에 캐패시터가 커플링된 형태를 단일 셀로 사용한다.

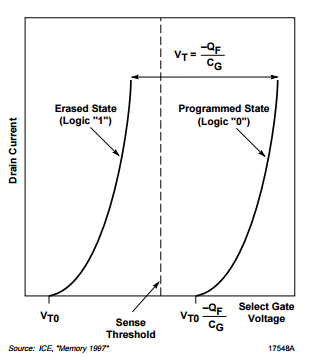
Write를 할 경우에는 캐패시터(4)에 전하들이 저장된다. 이 상태에서 Read를 할 경우에 WL가 트랜지스터(3)를 드라이브하면서 캐패시터에 저장돼 있던 전하들이 비트라인을 통해 출력된다.

DRAM이 SRAM과 가지는 가장 근본적인 차이 중에 하나는 메모리 셀의 데이터를 여러 번 참조할 경우 캐패시터의 전하들이 방전되면서 데이터가 사라지는 Destructive Read 문제가 발생한다는 것이다. 따라서 DRAM에서는 다른 메모리들고 다르게 Memory Refresh라는 작용이 필요하다. 일정 이상의 Read가 이루어지고 나서는 DRAM은 Memory Refresh를 통해 데이터 손실을 방지한다. 문제는 Memory Refresh가 Read/Write 작용과 동시에 이루어질 수 없다는 것이다. 메모리의 용량이 증가하고 데이터 처리 속도가 빨라질수록 Memory Refresh는 더 큰 병목으로 작용하고 있다. 64Gb 메모리를 사용할 때쯤이면 Refresh에만 64%의 시간을 소모하게 될 것이라고 한다[[3]](#footnote-3). Z80과 같은 초기의 프로세서들의 경우에는 CPU 에서 Memory Refresh 기능을 제공했었다고 한다.

ROM메모리의 경우에는 여러 종류들의 존재하는데 크게 PROM, EPROM, EEPROM 이다. 이 중에서 PROM은 현대에 거의 사용되지 않으므로 EPROM, EEPROM에 관해서만 정리하였다.

1. EPROM

Figure

EPROM은 UV Erasable Programmable Read Only Memory를 말한다. <Figure 6>에서 First-Level Polysilicon이 EPROM을 프로그래밍할 수 있게 만들어주는 곳이다. EPROM을 프로그래밍하기 위해서는 먼저 20분간 적외선에 셀을 노출시킨다. 이를 통해 EPROM의 모든 데이터들이 지워진다. 그 후에 지정된 셀들에 한해서 Hot electron injection이라는 작용을 통해 고압의 전류가 가해진다. 다량의 전하들이 게이트를 관통하면서 First-Level Polysilicon에 저장된다. 프로그래밍이 완료된 후에는 실수로 UV에 칩이 노출돼서 데이터들이 초기화되는 일을 막기 위해 EEPROM의 Quartz 창을 검은 물질로 가린다.

Figure

우측은 프로그래밍된 상태와 프로그래밍되지 않은 상태, 또는 게이트에 전하들이 충전돼 있는 상태와 그렇지 않은 상태에서 특성을 나타내는 표이다. 프로그래밍된 상태에서는 더 낮은 전압에서 Drain으로 전류가 흐르는 것을 확인 할 수 있다. 따라서 게이트에 전하들이 충전돼 있을 때가 논리적 0을 표현하고, 전하들이 충전돼 있지 않을 때가 논리적 1이 된다.

EPROM은 면적이 굉장히 작고 Field Programmable 하다는 점이 제일 큰 장점이다. 다만 필요로 하는 Gate Oxide의 기준이 상당히 높다는 점 때문에 생산 과정에서 유실되는 양이 상당히 많다. 또한 Quartz 크리스탈 창문이 달린 패키지를 사용해야 한다는 점이 EPROM을 상대적으로 고가로 만든다.

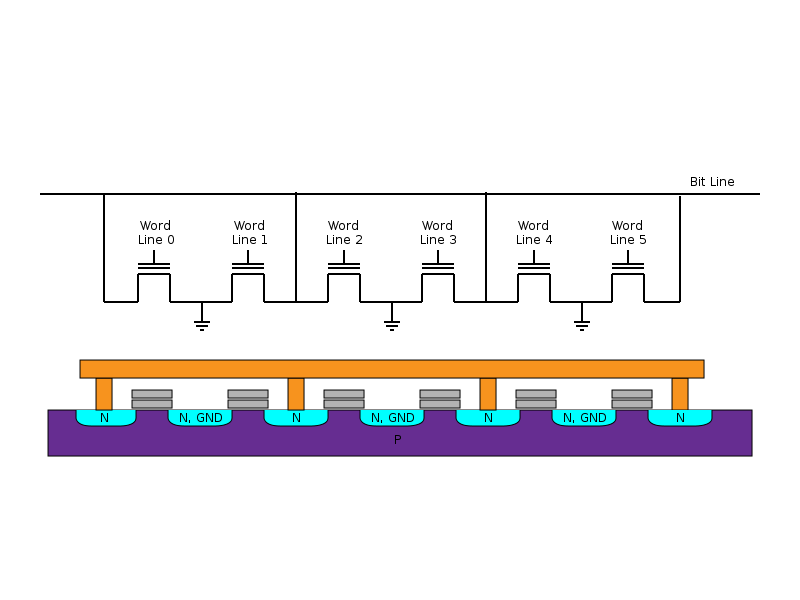
1. EEPROM

Electrically Erasable Programmable Read Only Memory, EEPROM은 EPROM 보다도 더 간편한 사용성을 가진다. 프로그래밍, 초기화 과정에서 필요한 고압의 전류가 내부적으로 생성되기 때문에 재프로그래밍이 굉장히 간편하다.

EEPROM의 경우에는 두가지가 있는데 각각 Parallel EEPROM과 Serial EEPROM이라고 부른다. 현재 시장에서는 90%가 Serial EEPROM이며 10%만이 Parallel EEPROM이다.

Parallel EEPROM의 경우에는 주소를 통한 접근이 평행적인 인터페이스를 통해서 이루어진다. 따라서 접근에 필요한 클락수가 매우 낮은 대신 메모리의 용량이 증가함에 따라 핀 수가 증가해야 하며 이것은 패키지 크기의 증가와 가격의 증가로 이어진다. 이에 비해 Serial EEPROM은 전력 소모량, 구동 전압이 낮으며 앞서 말했듯이 필요한 핀수가 적으며 크기가 작다. 이로 인해 Parallel EEPROM은 성능이 매우 중요한 어플리케이션에서는 사용되며 그 외 대부분의 경우에는 Serial EEPROM을 사용한다.

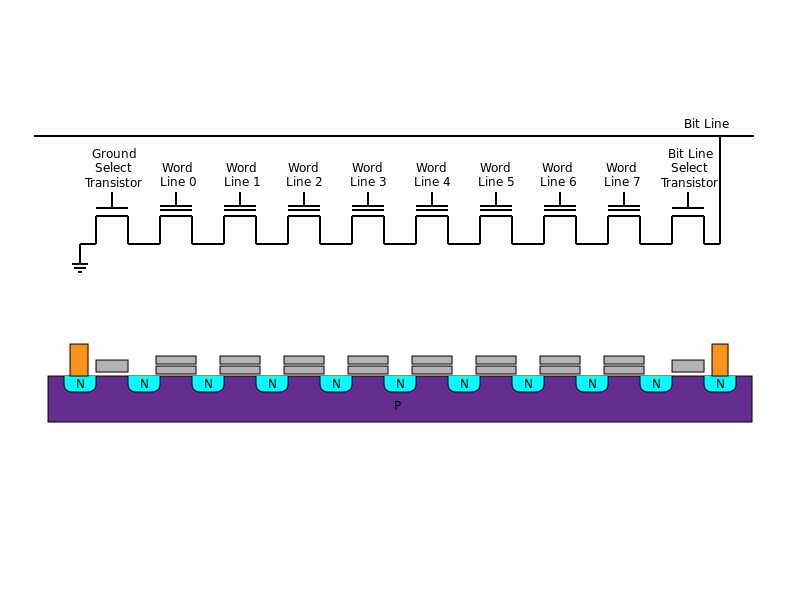
1. Flash EEPROM

Flash ROM 메모리는 EEPROM 기술을 기반으로 Toshiba가 개발하여 1984년에 출시하였다. 이로 인해 Flash ROM은 EEPROM과 똑 같은 원리에 비슷한 구조를 갖고 있으나 구분을 위해 Non-flash EEPROM 메모리만 EEPROM 이라고 부르고 있다. EEPROM은 비트 단위의 Read/Write를 제공하는 대신 Flash ROM은 여러 비트위 묶음 단위로 Read/Write를 제공한다.

Figure

Flash ROM 메모리에는 2가지 종류가 있는데 각각 NAND Flash랑 NOR Flash이다. NOR 메모리의 경우 <Figure >에서와 같이 두개의 Floating Gate MOSFET의 아웃풋이 그라운드로 커플링돼서 NOR게이트의 형태를 띈다. Word Line 에 신호가 가해지면 해당하는 셀이 신호를 그라운드로 당겨버려서 Bit Line의 신호가 낮아진다. 이러한 현상은 NOR 게이트와 흡사하다. NOR Flash는 Read/Write 딜레이가 적어서 속도가 중요한 곳에서 주로 볼 수 있다. 적은 오류로 인해서 정확성과 속도 모두 중요한 BIOS ROM에는 주로 NOR Flash가 사용된다.

NAND Flash는 NAND게이트처럼 여러 개의 트랜지스터들이 직렬로 연결돼 있어서 NAND와 비슷한 작용을 한다. 모든 직렬로 연결된 트랜지스터들이 ON 돼야만 Bit Line의 신호 레벨이 낮아진다. NAND Flash의 경우에는 이러한 트랜지스터들의 배치로 인해 추가적인 주소 연산 단계들이 필요하나, 라우팅이 훨씬 단순하고 덜 필요하기 때문에 더 밀도 높은 메모리를 생산할 수 있다.



Figure

1. Real Life Applications

DRAM의 가장 큰 어플리케이션은 컴퓨터 주메모리이다. 칩당 2~4GB 크기로 많이 생산되며 최근에는 4세대 DRAM인 DDR4로 기술이 발전해 있다. 2016년에 삼성은 16~19nm 수준의 DDR4 RAM생산이 가능하다고 발표했다. DDR4 DRAM은 정확하게 SDRAM 이라고 부른다. 이는 Synchronous Dynamic Random-Access Memory를 뜻한다. 일반적인 DRAM과의 차이는 외부의 클락 신호를 이용해서 Memory Controller가 Finite-State-Machine을 구현하여 메모리를 작동시킨다는 것이다. 이 방식을 통해 메모리 연산을 Pipelining해서 성능을 높일 수 있다. Memory Controller의 종류에는 Static과 Dynamic이 있으며 Dynamic은 런타임에 메모리 연산의 스케줄을 생성해낸다는 차이가 있다.

SRAM은 컴퓨터의 CPU에서 Cache를 구현하는데 이용된다. DRAM에 비해서 면적대비 메모리 용량이나 가격은 불리하나 메모리 연산 속도는 DRAM에 비해서 절대적으로 빠르며 높은 Clock rate에서도 정상적으로 구동하다. 이로 인해 속도가 절대적으로 중요한 CPU Cache에서 주로 사용된다. 오늘날 프로세서들은 3단계의 Cache 구조를 갖고 있다. 각각 L1, L2, L3 캐시이며, L1 캐시는 속도를 기준으로, L2, L3 캐시는 용량을 기준으로 최적화된 SRAM을 사용한다. 멀티코어 패러다임에서는 코어간의 메모리 공유 자원 이슈로 인해 캐시를 여러 코어가 공유한다. L1 캐시는 각 코어마다 하나씩, L2 캐시는 코어 2개마다 하나씩, L3 캐시는 모든 코어들이 공유하는 것이 일반적이다.

ROM 메모리는 그래픽 카드, 하드디스크, DVD 플레이어, 키보드 등의 기기에서 Firmware를 내부적으로 포함하고 있어야 하는 경우에 사용된다. 예전에는 컴퓨터의 BIOS 또는 부트 섹션도 EEPROM, EPROM을 사용하였으나 최근에는 플래시 메모리가 그 자리를 대체하고 있다. 플래시 메모리의 등장으로 인해 오늘날에는 보조 메모리에 ROM을 사용하는 것이 점점 보기 쉬워지고 있다.

1. 참조한 2003년에 작성된 것이라 현대에는 달려졌을 수 있다.

   “Current Sense Amplifiers for Embedded SRAM in High-Performance System-on-a-Chip Designs” (2003); Wicht, Bernhard; Springer-Verlag Berlin Heidelberg; [↑](#footnote-ref-1)
2. “High-Performance and Low-Voltage Sense-Amplifier Techniques for sub-90nm SRAM”; Manoj Sinha et al; Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, USA\* Microprocessor Research Labs, Intel Corporation, Hillsboro, OR 97124, USA; [↑](#footnote-ref-2)
3. Liu et al., “RAIDR: Retention-Aware Intelligent DRAM Refresh,” ISCA 2012. [↑](#footnote-ref-3)