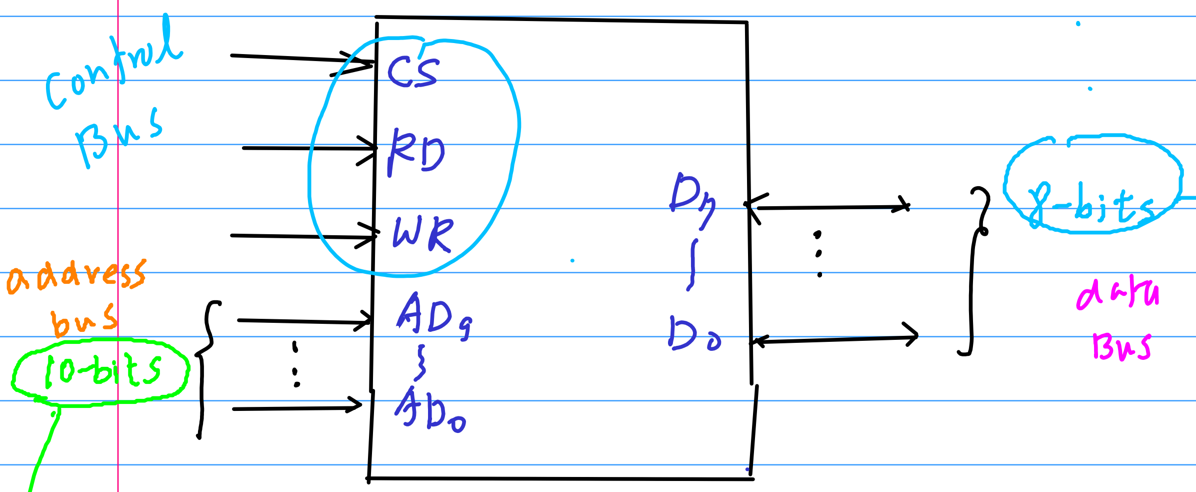
DRAM (H.1)

2006.06.06

www.wikiversity.org

The necessities in Computer Organization

RAM (Random Access Memory)



Chip Select

CS	RD	WR	
0	X	X	선택되지 않는
1	1	0	읽기 동작
1	0	1	쓰기 동작

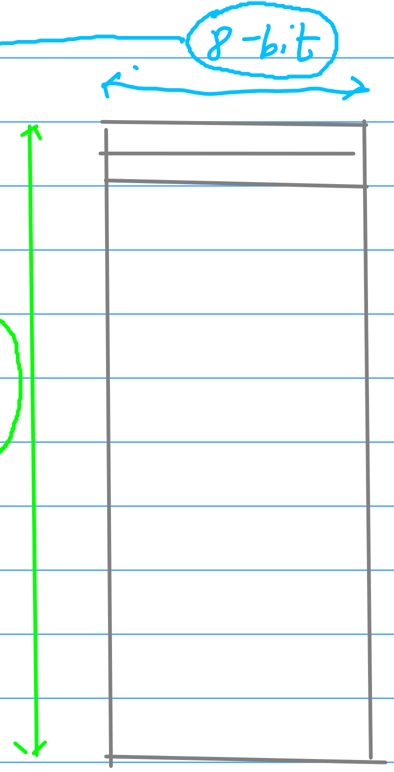
memory size $1K \times 8$ bit

$AD_9 \sim AD_0$ $D_7 \sim D_0$

$2^{10} \times 8$ bit

1K

$2^{10} = 1024$



DRAM ---- capacitor ← 전하가 있으면 '1'
없으면 '0'

시간이 흐르면 전하가 사라진다 (leak)

Dynamic → refresh (주기적 재충전)

SRAM --- Flip-flop

Static (refresh X)

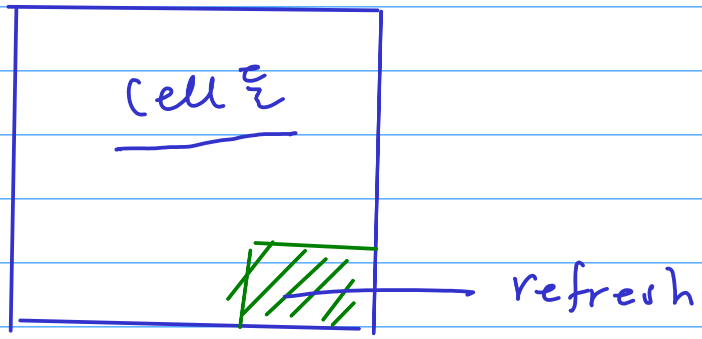
Cell ---- memory에서 1-bit을 저장하기 위한 기본구조

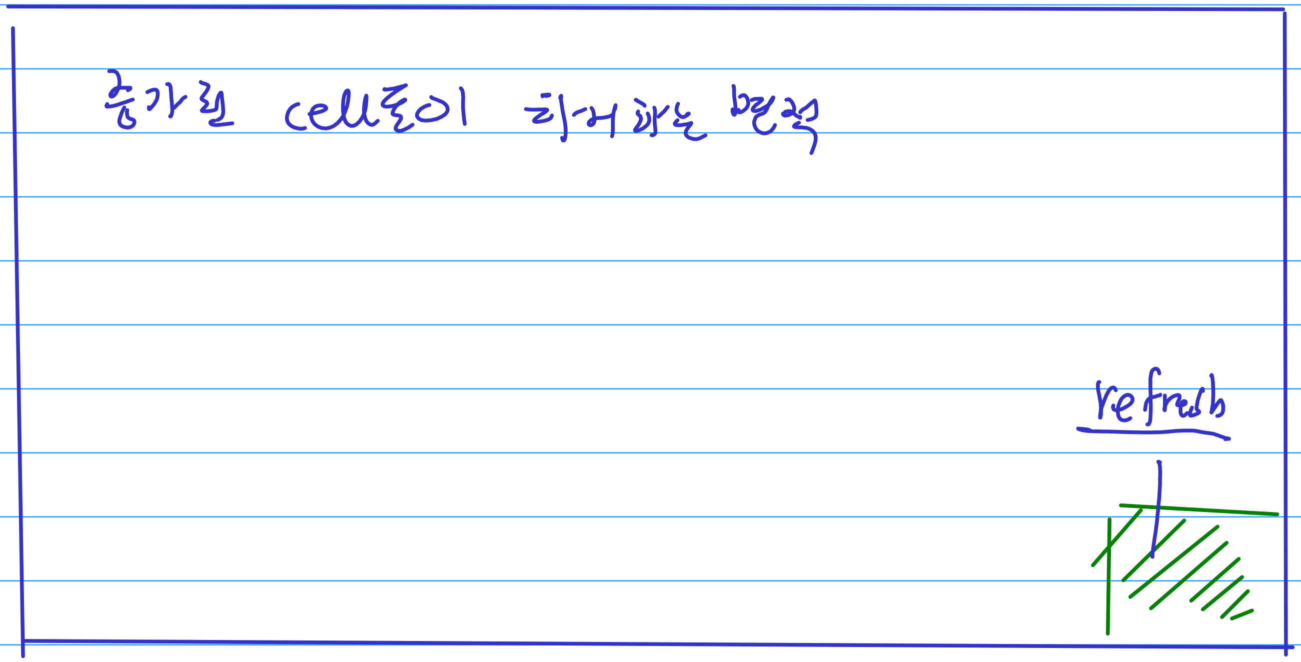
DRAM cell은 SRAM cell에 비하여 간단하므로
(적은 수의 transistor로 구현 가능)
반도체 chip 면적을 적게 차지한다.
단위 면적 당 많은 cell을 집적할 수 있어서
low cost integrate
↳ 대용량 main memory로 사용된다.

SRAM — fast - 비싸다 — Cache memory
로 사용된

→ refresh 회로는 chip에서 차지하는 면적이
DRAM의 용량이 커질수록
상대적으로 줄어들기 때문에 가격에 영향이 없다

→ refresh 회로는 chip에서 차지하는 면적이
DRAM의 용량이 커질수록
상대적으로 줄어들기 때문에 가격에 영향이 없다





64 bits

① 8 x $\boxed{8 \text{ bit}}$ organization   address: 3-bits

② 16 x $\boxed{4 \text{ bit}}$ organization   address: 4-bits

③ 64 x $\boxed{1\text{-bit}}$ organization.   address: 6 bits

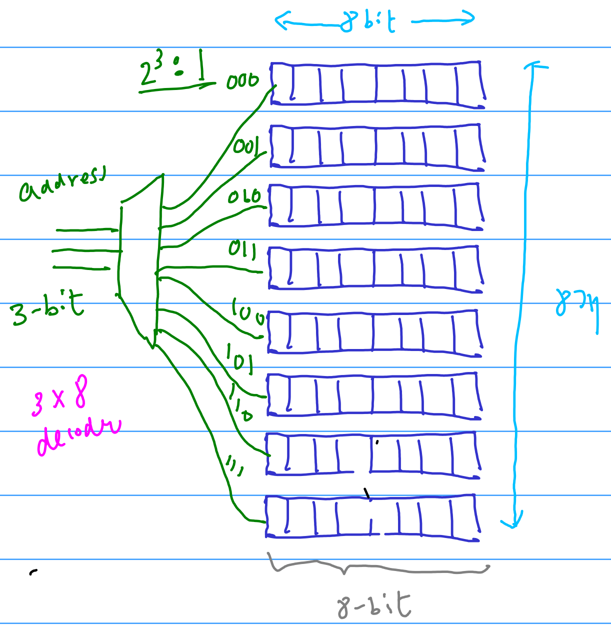
row addr: 3-bits
col addr: 3-bits

data: 1-bit

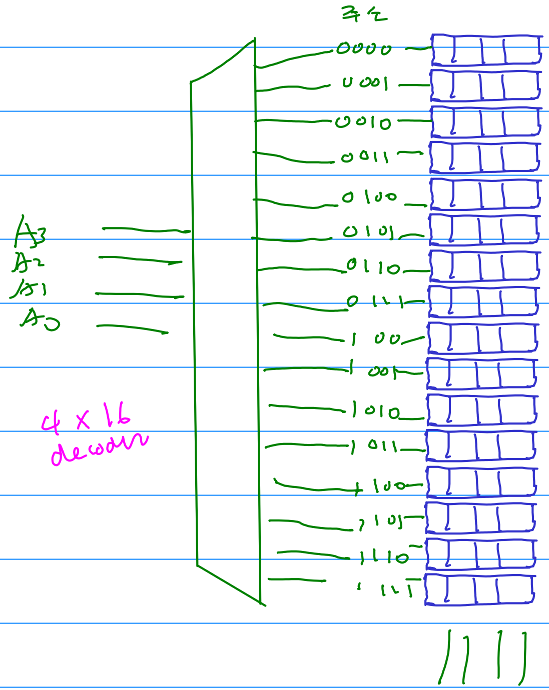
④ 16M x $\boxed{4 \text{ bit}}$ data: 4-bit

64-bit Memory Configuration

① 8 x 8 bit organization



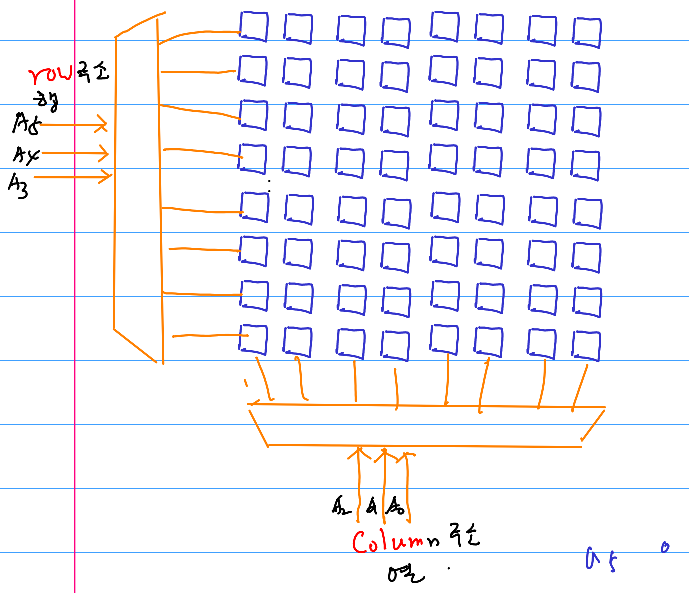
② 16 x 4 bit organization



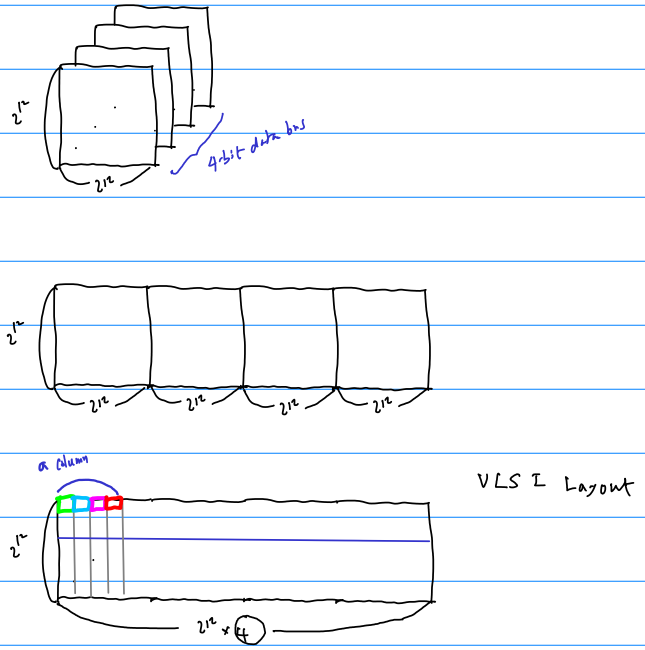
③ 64 x 1-bit organization.

2^6

address : 6-bit　데이타 : 1-비트

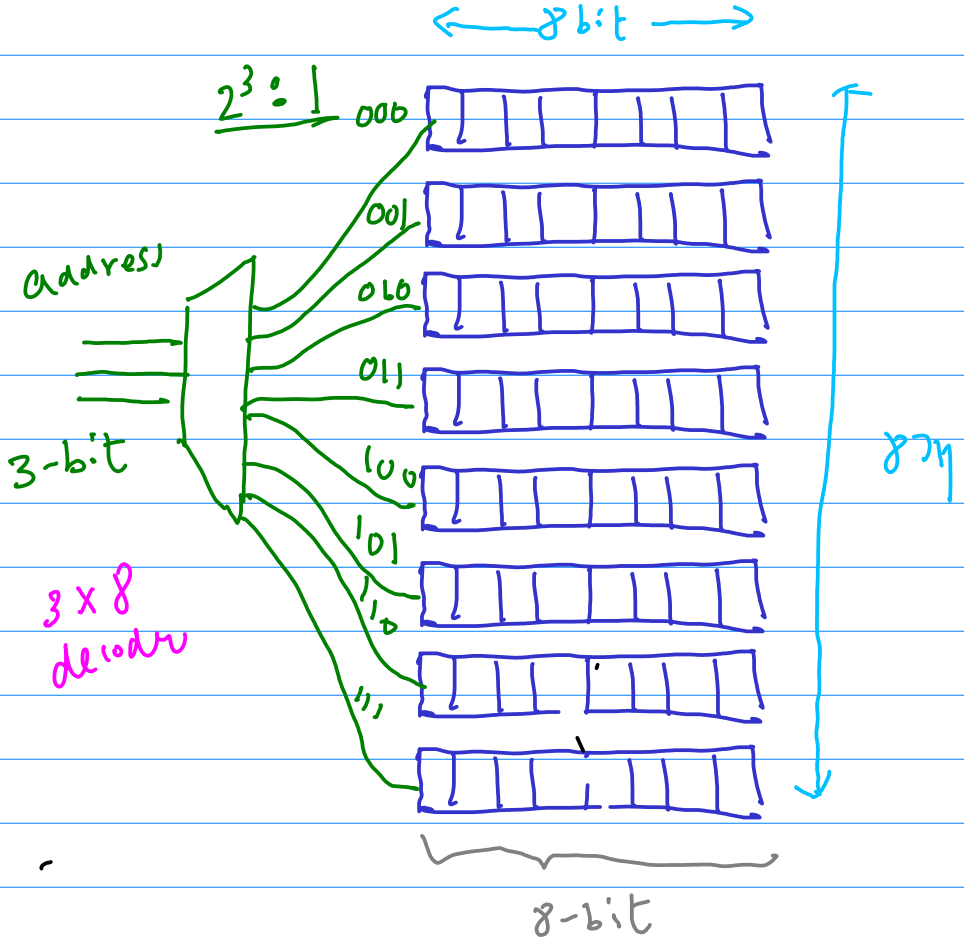


④ 16M x 4 bit



64

① 8 x 8 bit · organization



$1K = 2^{10}$

$1M = 2^{20}$

만약 용량이 8kbit → $\underbrace{\overset{2^{10}}{1K}}_{\text{address 10-bit}} \times \underbrace{\boxed{8\text{-bit}}}_{\text{data 8-bit}}$

1Mbit → $\underbrace{2^7 K}\times \boxed{8\text{-bit}}$

$$2^{20} = 2^{17} \cdot 2^3$$
$$= 2^7 \cdot 2^{10} \cdot 8$$
$$= 2^7 K \cdot 8$$

$\underbrace{128K}_{\text{address 17-bit}} \times \underbrace{\boxed{8\text{-bit}}}_{\text{data 8-bit.}}$

용량이 1Mbit $= 2^{20}$ bits

$= \underline{2^7} \cdot 2^{10} \cdot 2^3$ bits

$= 127 \cdot K \cdot 8$

$\underline{127K} \times \underline{8 \text{ bits}}$

2^{17} | 1

address 17-bit | data 8-bit

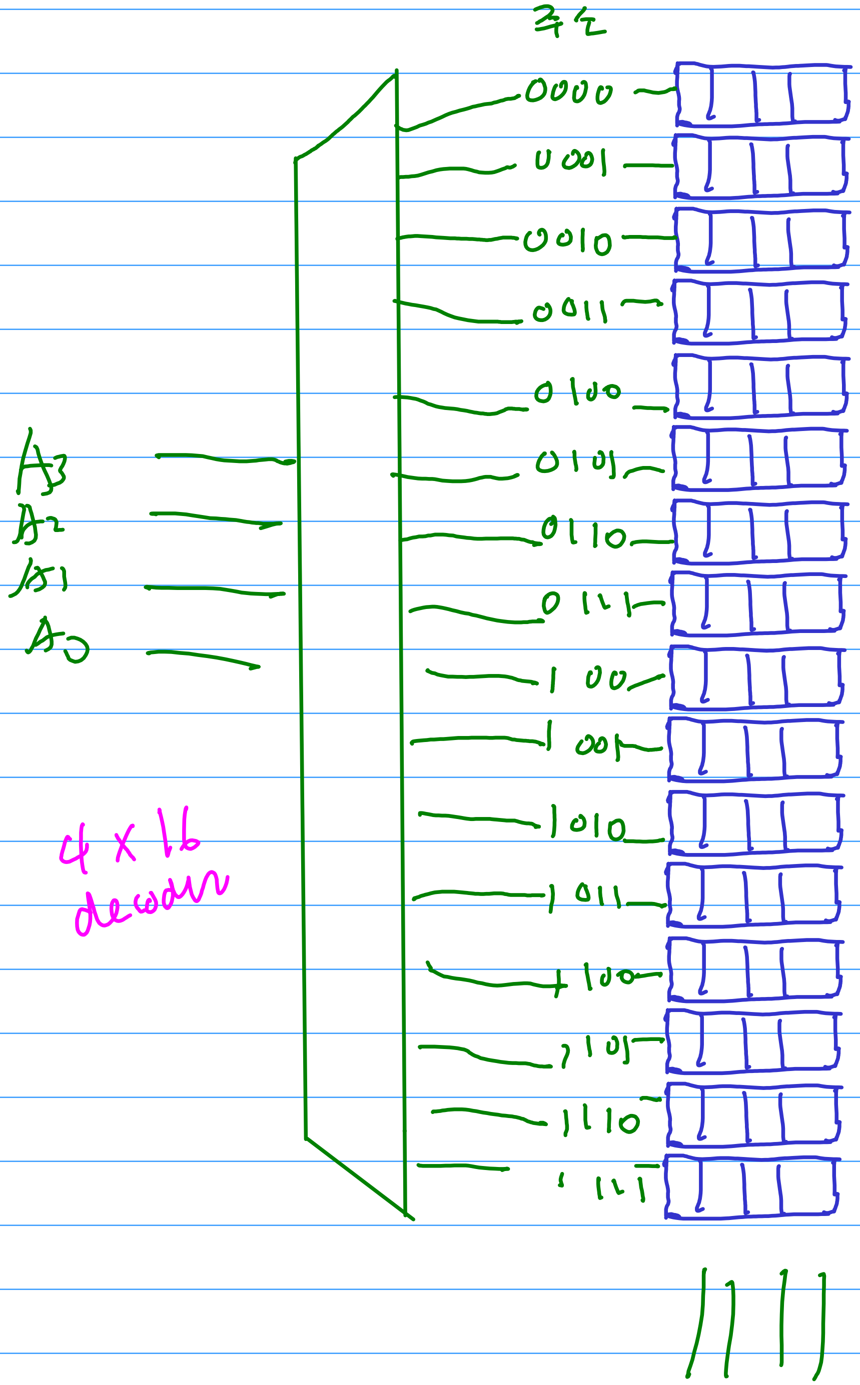
용량이 1Gbit $= 2^{30}$ bit

$= 2^7 \cdot 2^{20} \cdot 2^3$

$\underbrace{128M}_{\text{address 27-bit}} \times \underbrace{8\text{-bits}}_{\text{data 8-bit}}$

64

② 16 x 4 bit organization



총 용량이 8K bit

$2^3 \cdot 2^{10}$ — $2^{11} \cdot 2^2$

$\underbrace{2K} \times \underline{4 bit}$

address : 11 bit   data = 4bit

총 용량이 1 Mbit

2^{20} bit $2^8 \cdot 2^{10} \cdot 2^2$

$\underline{256K} \times \underline{4 bits}$

add: 18   data 4bits

총 용량이 1 Gbit

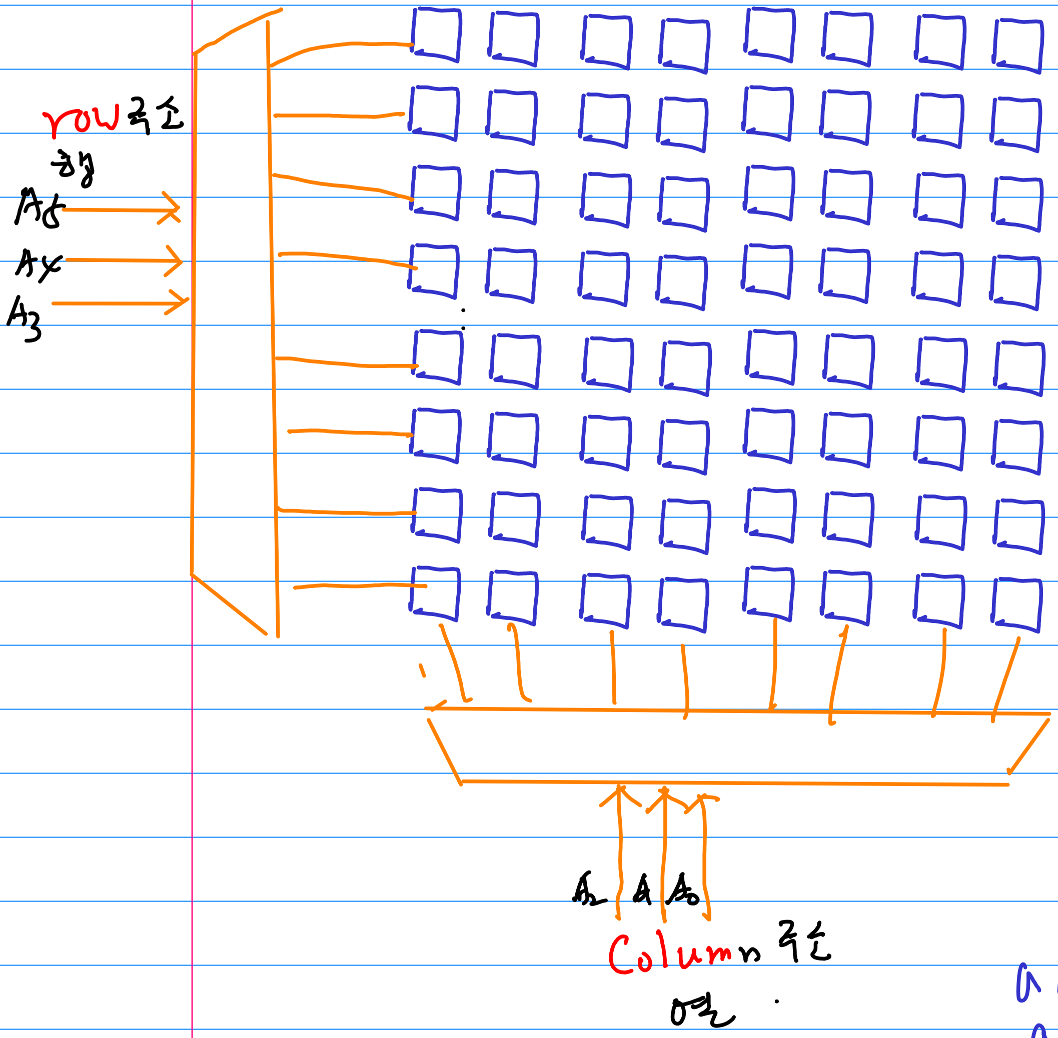
2^{30} $2^8 \cdot 2^{20} \cdot 2^2$

$\underline{257 M} \times 4 bits$

address 28 $\overline{\text{data 4bit}}$

③ $\underbrace{64}_{2^6}$ x 1-bit organization.

address : 6-bit data : 1-bit

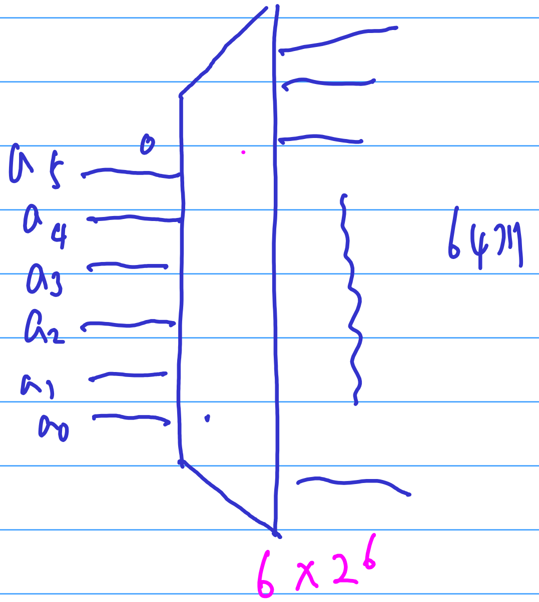


$\underline{64}$ x $\underline{1\text{-bit}}$

$\underline{2^6}$ data : 1-bit

address : 6-bit

decoder : $\underline{6 \times 64}$



6×2^6

a_5 a_4 a_3 a_2 a_1 a_0

6 x 64
decoder

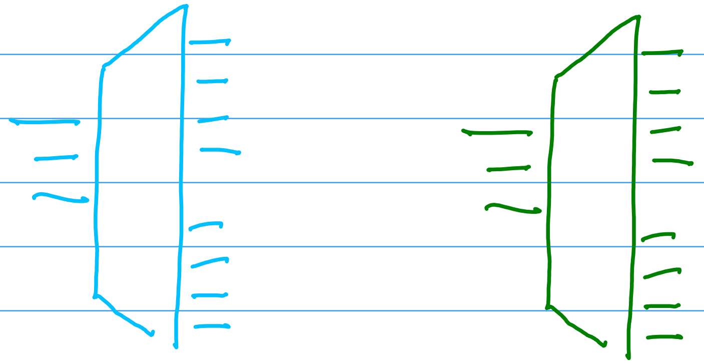
a_5 a_4 a_3

row address

a_2 a_1 a_0

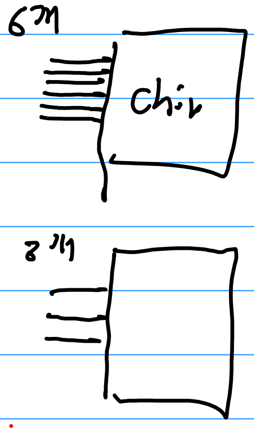
colum address

6x64 decoder



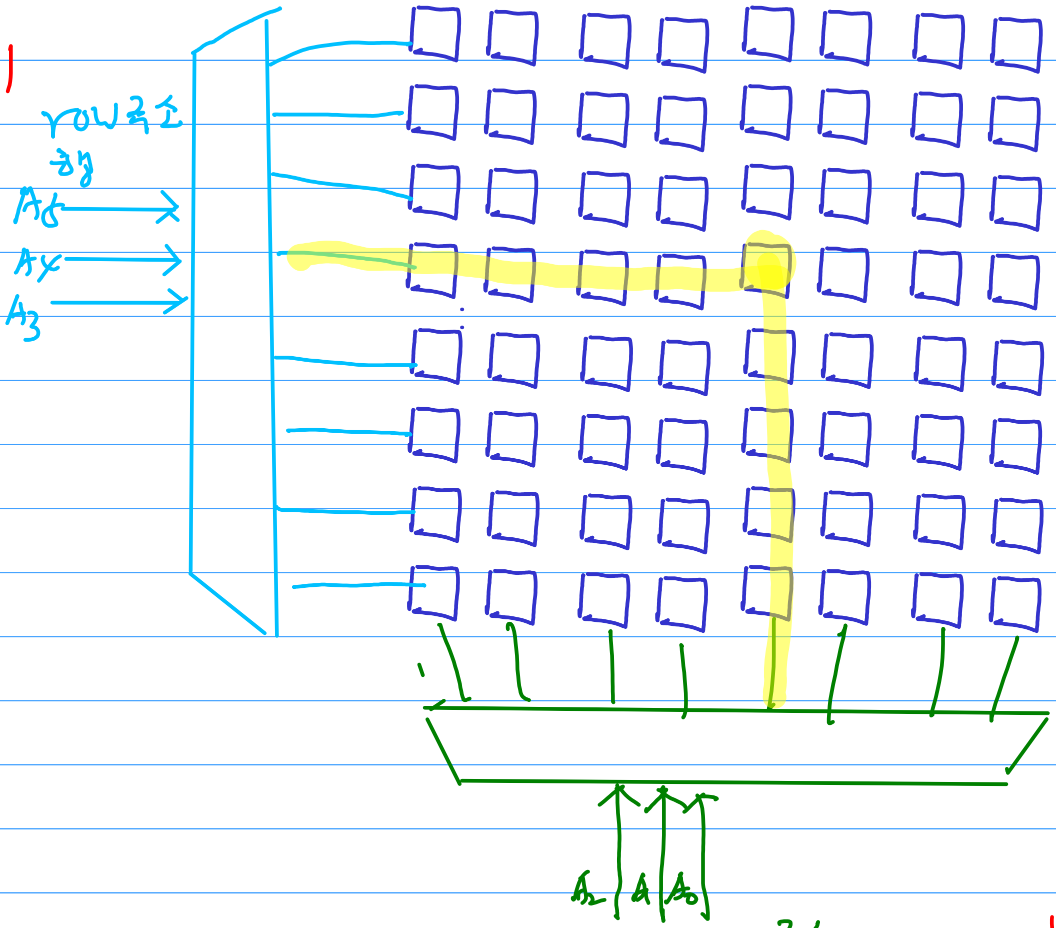
3x8 decoder

3x8 decoder



package

row address



Column주소 열.

col address

총 용량이 8K bit

$\underline{8K} \times 1\text{-bit}$

$2^3 \cdot 2^{10}$

2^{13}

address: 13-bit data = 1-bit

용량이 1 Mbit

$\underline{1M} \times 1\text{-bit}$

2^{20}

address: 20 data : 1-bit

용량이 1 Gbit

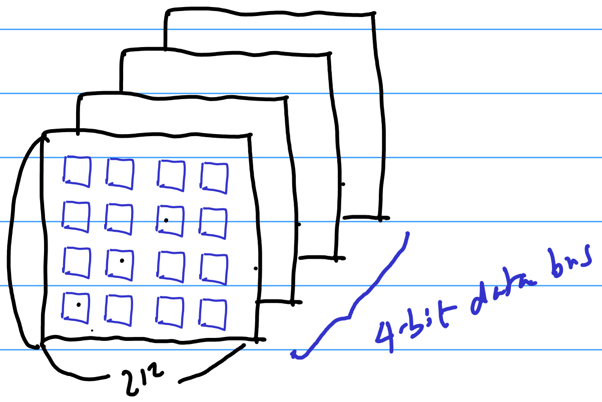
$1G \times 1\text{-bit}$

2^{30}

address: 30-bit data: 1-bit

④ 16M × 4 bit

$16M = 2^4 \cdot 2^{20} = 2^{24} = (2^{12})^2$



Conceptual

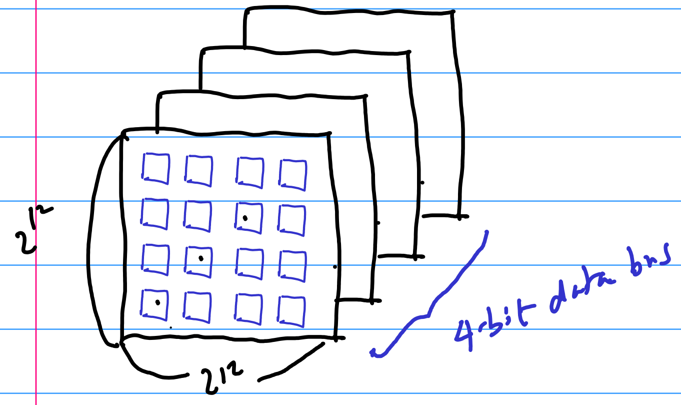
총용량 64Mbit

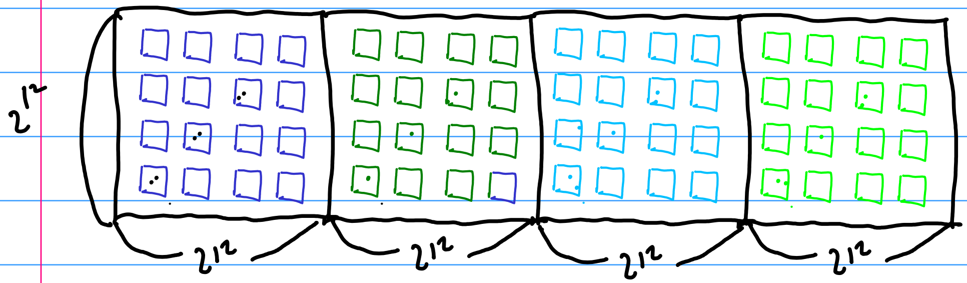
$2^6 \cdot 2^{20} = 2^{26}$

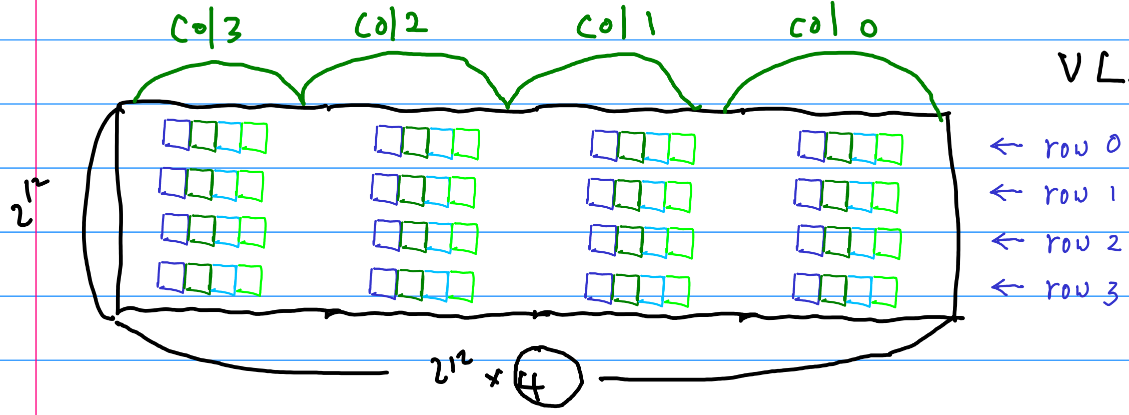
$2^{12} \cdot 2^{12} \cdot 2^2$

$2^{12} = 2^2 \cdot 2^{10} = 4 \times 1024 = 4096$

$$\underbrace{\underbrace{4096}_{\text{12-bit row address}} \times \underbrace{4096}_{\text{12-bit col address}}}_{\text{24-bit address}} \times \text{4-bit}$$

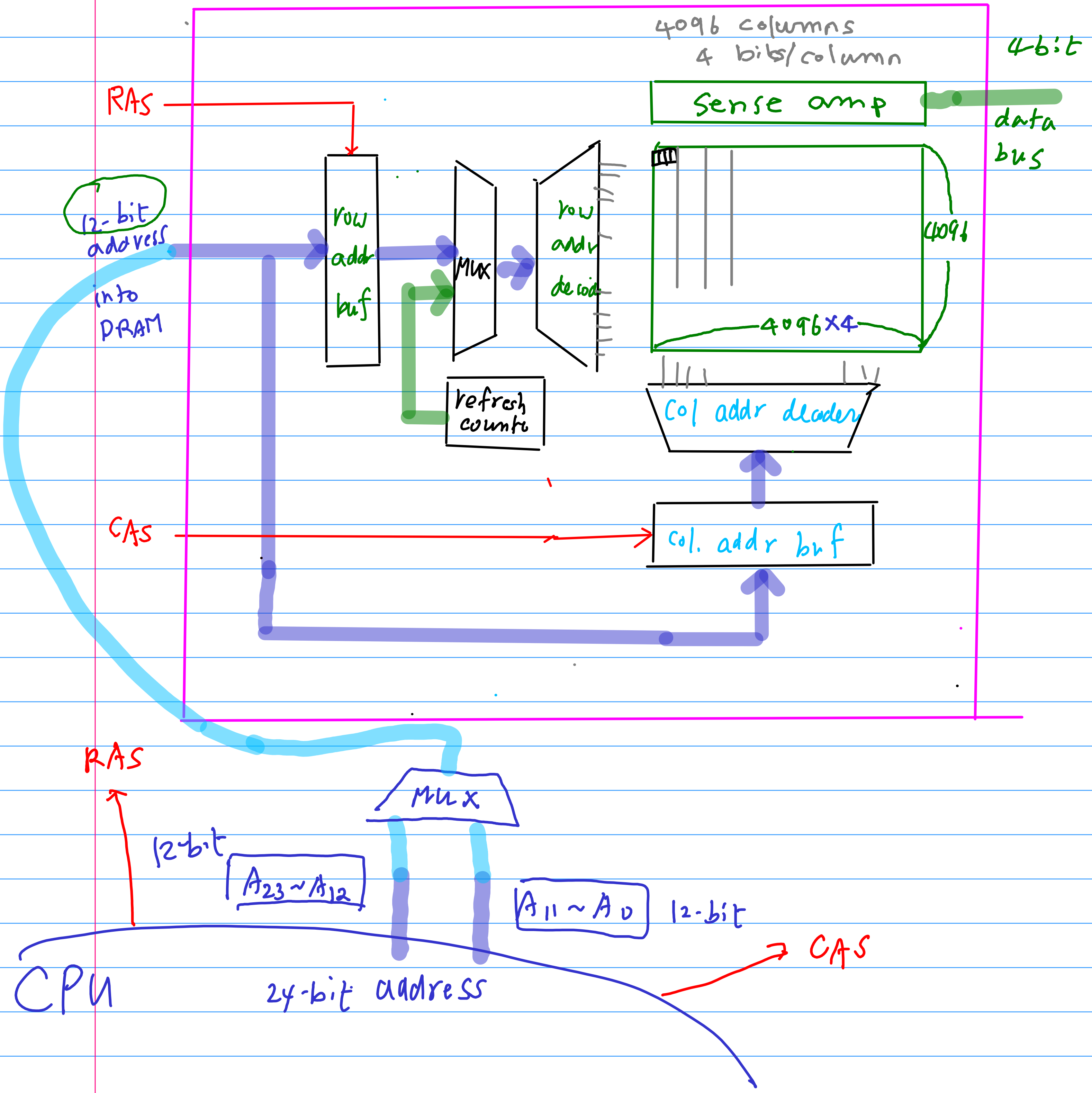

2^12
2^12
4bit data bus

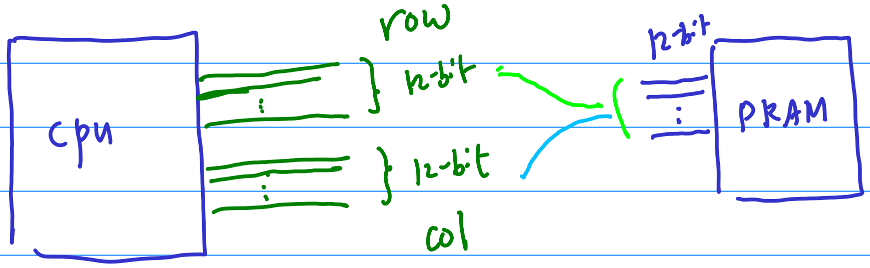

2^12
2^12
2^12
2^12
2^12


col 3
col 2
col 1
col 0
← row 0
← row 1
← row 2
← row 3
2^12
2^12 x 4

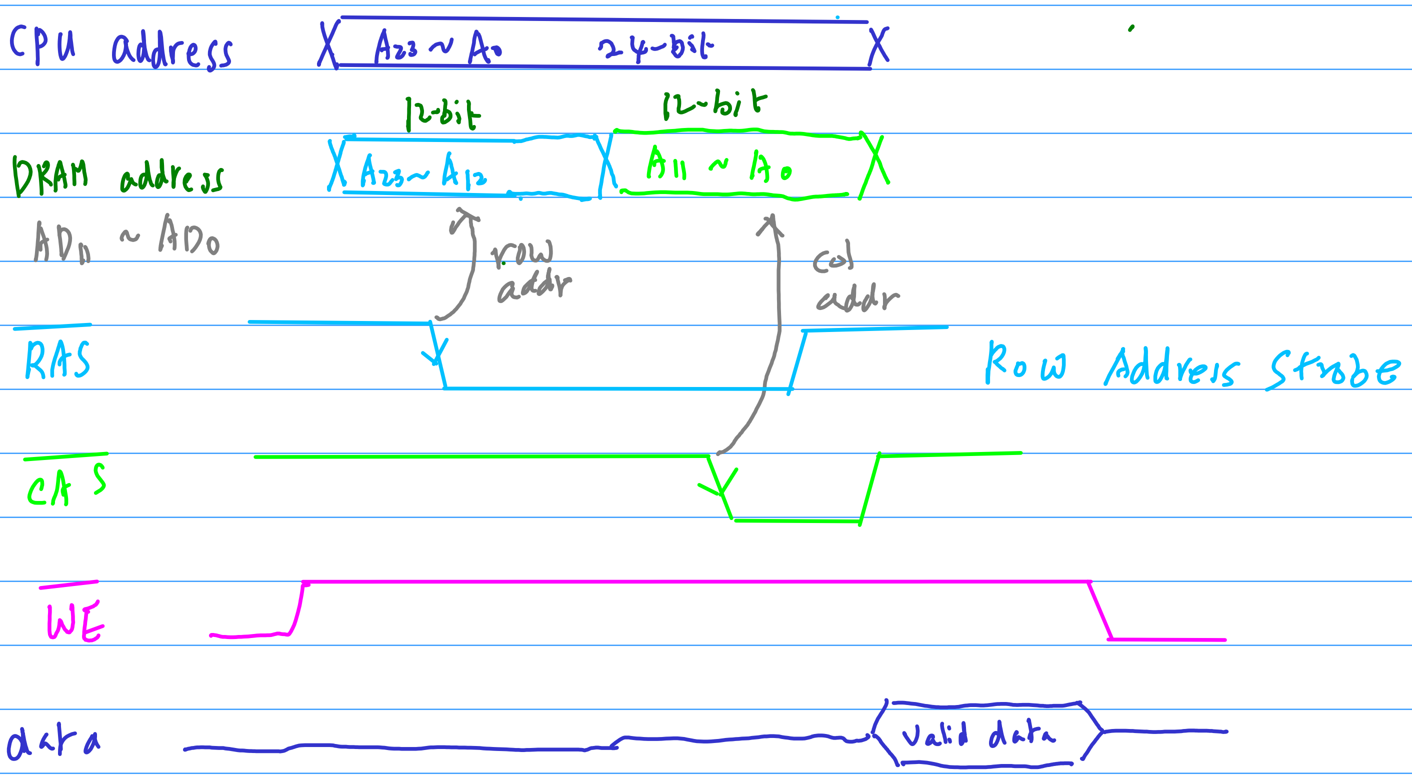
VLSI Layout

64Mbit (16M×4bit) DRAM의 구조





timing diagram

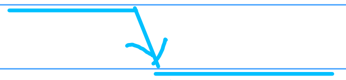


CPU address는 CPU가 내보내는 (출력)하는 총 24-bit address를 의미한다

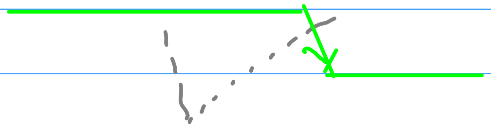
$$\underbrace{A_{23} \cdots\cdots A_{12}}_{\text{Row Address}} \; \underbrace{A_{11} \cdots\cdots A_{0}}_{\text{Col address}}$$

시간적으로 먼저 선택이 되어서 DRAM chip의 address로 연결된다

RAS (Row Address strobe)

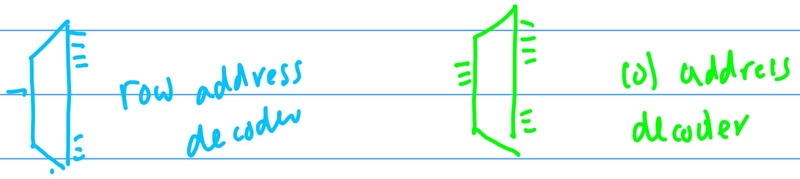


다음에 하위 12-bit의 선택이 되어서 DRAM chip의 address로 연결된다

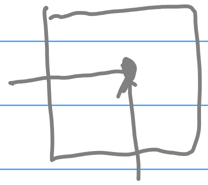


DRAM의 내부에 Address Buffer Register에 latch

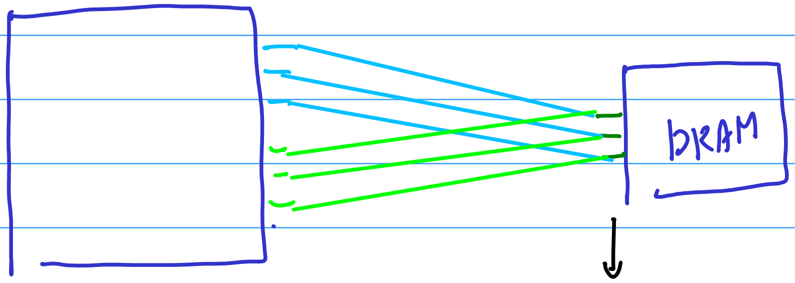
Add Buffer Reg에 latch된 주소들은 각각



에 의해서 해당 기억장소를 선택한다



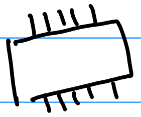
시간적으로 multiplexing



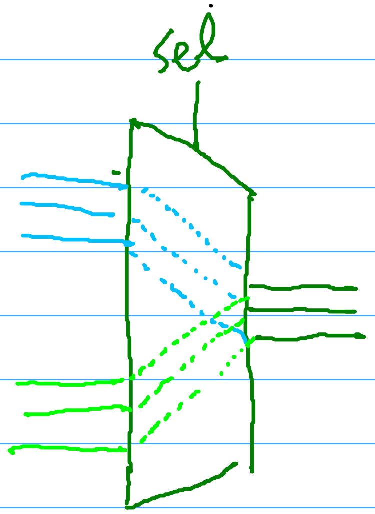
↓

24 bit → 12-bit

chip의 pin 수를 줄임

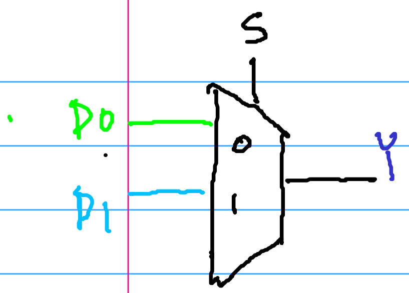


RAS or CAS



Memory Controller

Multiplexor



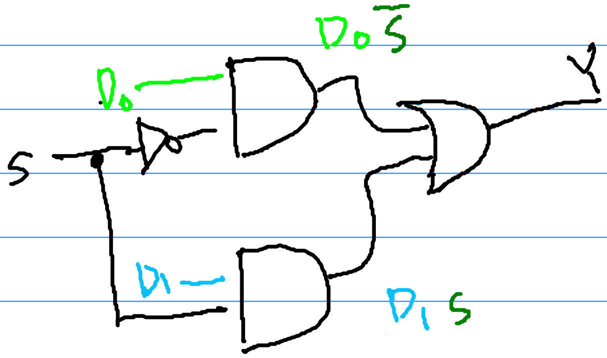
S	D_1	D_0	Y	
0	0	0	0	D_0
0	0	1	1	D_0
0	1	0	0	D_0
0	1	1	1	D_0
1	0	0	0	D_1
1	0	1	0	D_1
1	1	0	1	D_1
1	1	1	1	D_1

S	Y
0	D_0
1	D_1

$$Y = D_0\bar{S} + D_1 S$$

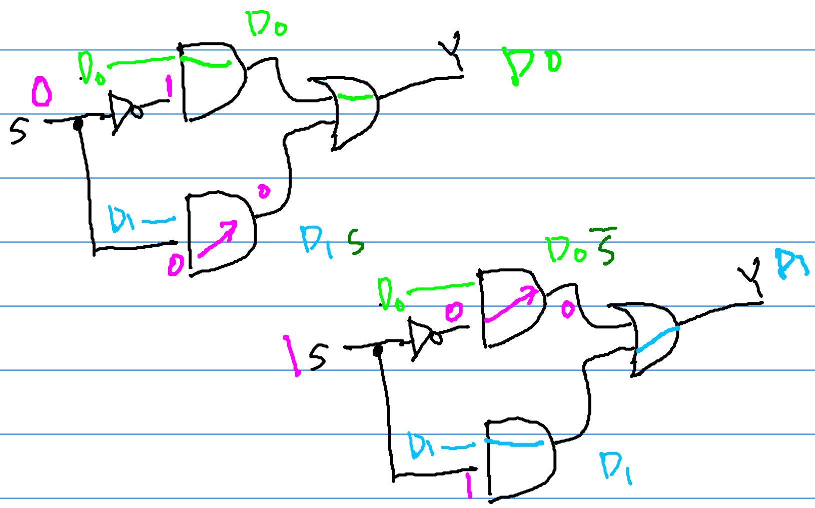
$\bar{S}=1$ 일때 D_0 이거나

$S=1$ 일때 D_1

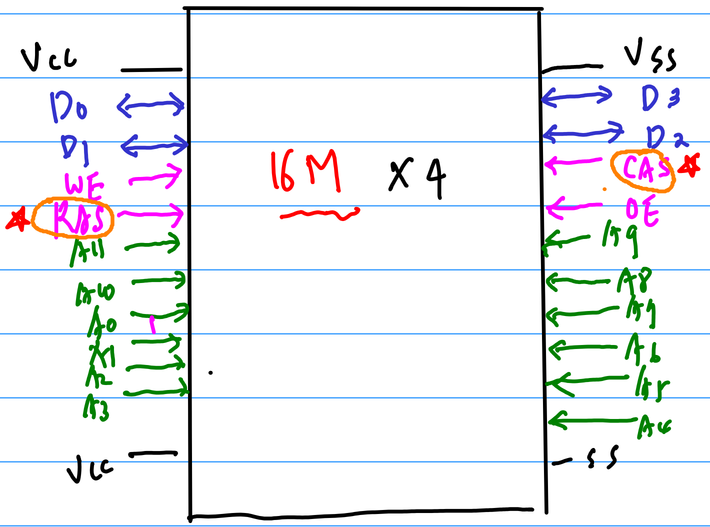


$S=0$ 일때 D_0 이거나

$S=1$ 일때 D_1



DRAM Chip



$$(2^{12})(2^{12})$$
$$= 2^{24}$$
$$= 2^{4} \cdot 2^{20}$$
$$= 16 \cdot M$$

data bus	D3 ~ D0	4-bit
addr bus	A11 ~ A0	12-bit
control bus	WE, OE, CAS, RAS	

DRAM

* clock X

Async DRAM w/o banks

SDRAM

- Dynamic
- Synchronous 동기식
  - Clock 신호에 맞추어서 data를 전송한다

원래 DRAM은
필요한 data 와 control signal 들어오는 즉시
access 동작 (RD / WR) 수행하여 응답한다

쓰기동작 : 전송된 data를 즉시 해당 기억장소에 저장
읽기동작 : 인출동작을 수행한후 즉시 data 버스로 전송

DRAM에서는 이런 동작들이 연속적으로 발생
→ cpu가 wait
→ Bus는 다른 장치 가 사용 X
→ 성능이 안좋다.

SDRAM 액세스동작이 clock에 동기되어서 수행된다.

① CPU가 1 clock cycle 동안
address 와 읽기 제어 신호를 BUS로 보낸다.
CPU는 결과를 기다리지 않고 다른 연산을 수행하수 있고

⋮

② SDRAM이 주소와 읽기 신호를 받은 즉시
access 동작을 수행한다
SDRAM System Bus 사용권리 획득한후
1 clock cycle 동안 data를 Bus로 전송한다

③ CPU가 Bus에 있는 data를 읽는다.

SDRAM

Parallel access (pipeline access) ← 여러개의 $\boxed{\text{bank}}$ 사용

bank마다 다른 주소에 대한 access가 동시에 수행됨

512M bit 용량 SDRAM

$\boxed{\text{4개의 Bank}}$ 사용

512 M

$2^9\ 2^{20}$

$\underline{2^{27}} \times \underbrace{2^2}_{\text{bank}}$

$\underline{2^{24}} \times 2^3$

$2^4 \times 2^{20} \times 2^3$

$\boxed{16M \times 8\text{-bit}}$ 1 bank의 구성.

data bus의 width

$\boxed{16M \times 8\text{-bit}} \times \underline{4\ \text{bank}}$

$= 16M \times \underline{4\ \text{bank}} \times 8\text{-bit} = \underline{64M} \times 8\text{-bit}$

$= 2^4 \cdot 2^{20} \cdot 2^2 \times 8\text{-bit} = 2^6 \cdot 2^{20} \times 8\text{-bit}$

$\underline{2^{26}} \times 8\text{-bit}$

즉 address $\underline{26\text{-bit}}$

256 Mbit SDRAM — Banks

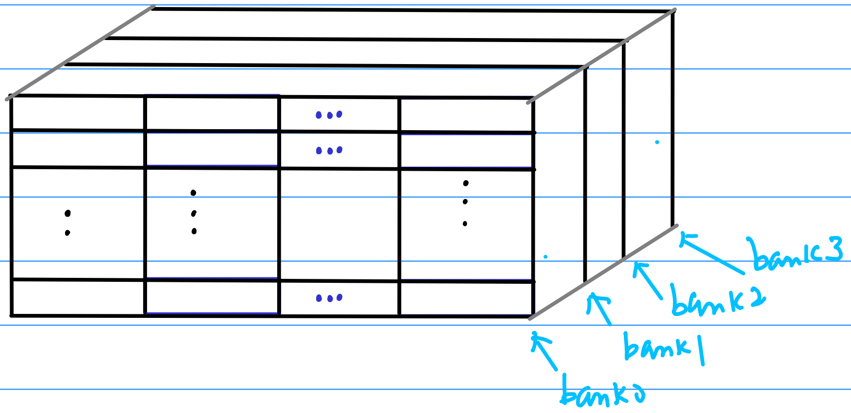
64M x4-bit $\quad 2^6 \cdot 2^{20} \cdot 2^2 = 2^{28}$

32M x 8-bit $\quad 2^5 \cdot 2^{20} \cdot 2^3 = 2^{28}$

16 M x 16-bit $\quad 2^4 \cdot 2^{20} \cdot 2^3 = 2^{28}$

2^8 M bit = 256 Mbit memory

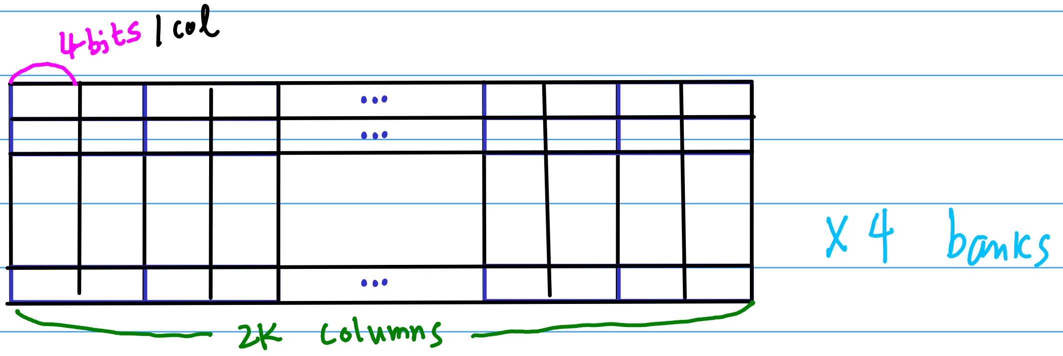
	64M x4-bit	32M x 8-bit	16 M x 16-bit
configuration	16M x 4 x 4 bnk (4+20+2) bit	8 M x 8 x 4 bnk (3+20+2) bit	4 M x 16 x 4 bnk (2+20+2) bit
row addr	8K (13-bit)	8K (13-bit)	8K (13-bit)
bnk addr	4 (2-bit)	4 (2-bit)	4 (2-bit)
col addr	2K (11-bit)	1K (10-bit)	512 (9-bit)
	26-bit	25-bit	24-bit

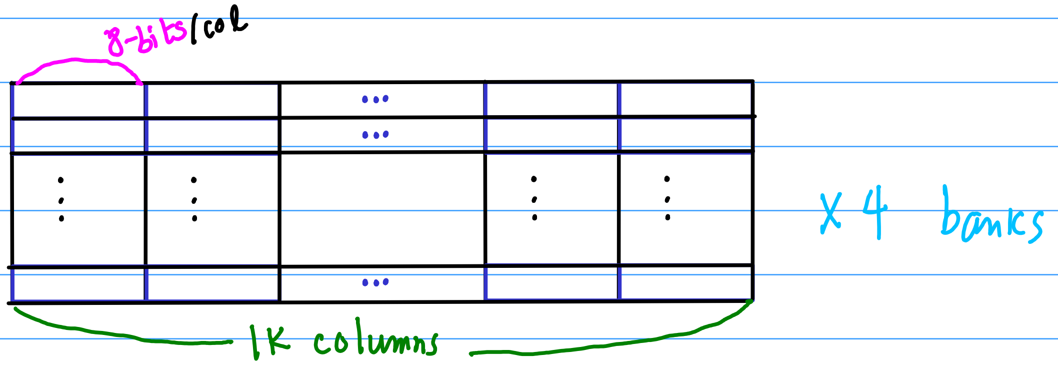


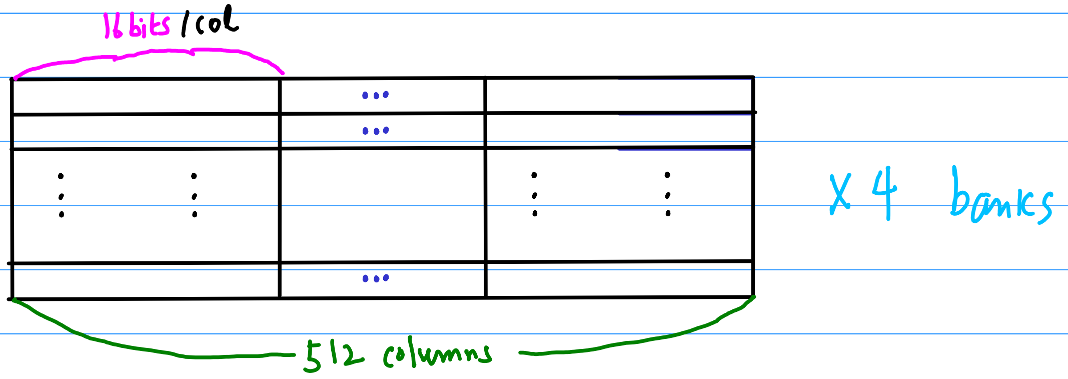
4 banks

256 Mbit SDRAM – Data Width (4, 8, 16-bit)

	64M x 4-bit	32M x 8-bit	16M x 16-bit
configuration	16M x 4 x 4 bnk (4+20+2) bit	8M x 8 x 4 bnk (3+20+2) bit	4M x 16 x 4 bnk (2+20+2) bit
row addr	8K (13-bit)	8K (13-bit)	8K (13-bit)
bnk addr	4 (2-bit)	4 (2-bit)	4 (2-bit)
col addr	2K (11-bit)	1K (10-bit)	512 (9-bit)
	26-bit	25-bit	24-bit







256 Mbit SDRAM – SIMM, DIMM

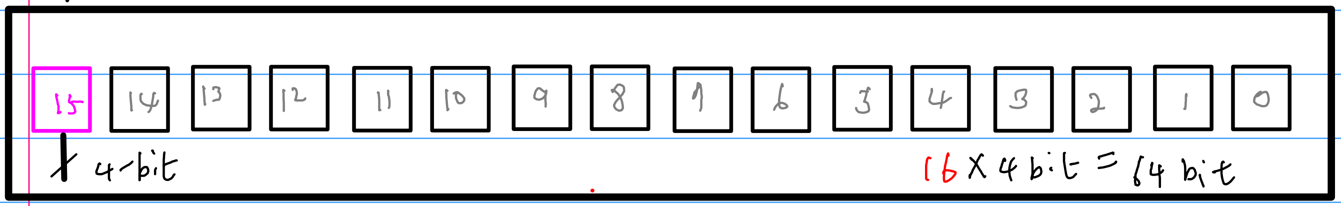
SIMM (Single In-Line Memory Module) single side

DIMM (Double In-Line Memory Module) double side

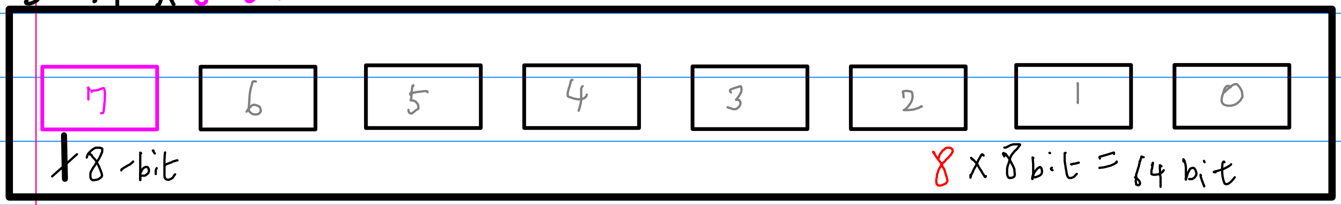
a unit of 64-bit data width

System Bus Data width

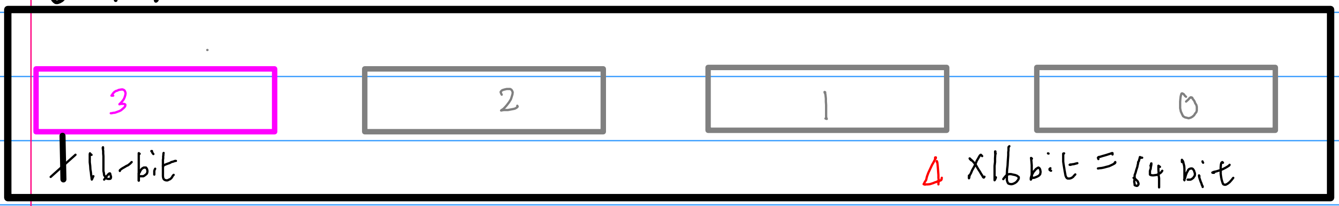
64M x 4-bit



32M x 8-bit



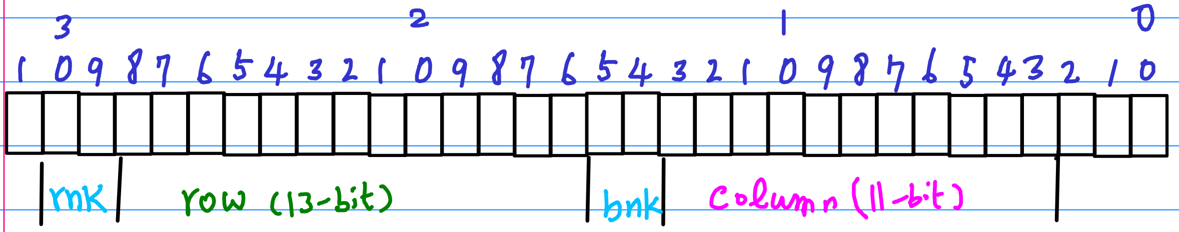
16M x 16-bit



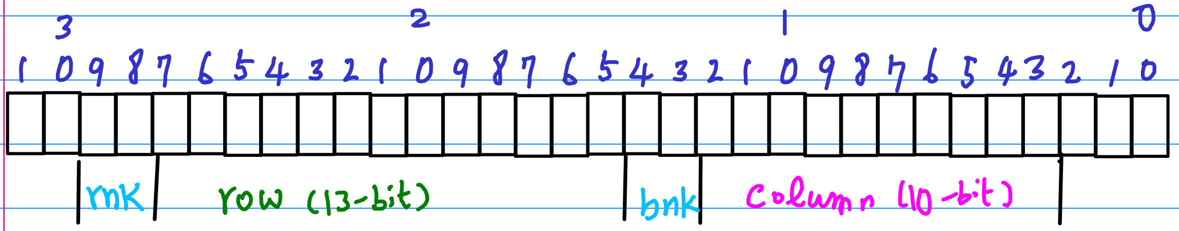
256 Mbit SDRAM – Bank Interleaving

	64M x4-bit	32 M x 8-bit	16 M x 16-bit
configuration	16M x 4 x 4 bnk (4+20+2) bit	8 M x 8 x 4 bnk (3+20+2) bit	4 M x16x 4 bnk (2+20+2) bit
row addr	8K (13-bit)	8K (13-bit)	8K (13-bit)
bnk addr	4 (2-bit)	4 (2-bit)	4 (2-bit)
col addr	2K (11-bit)	1K (10-bit)	512 (9-bit)

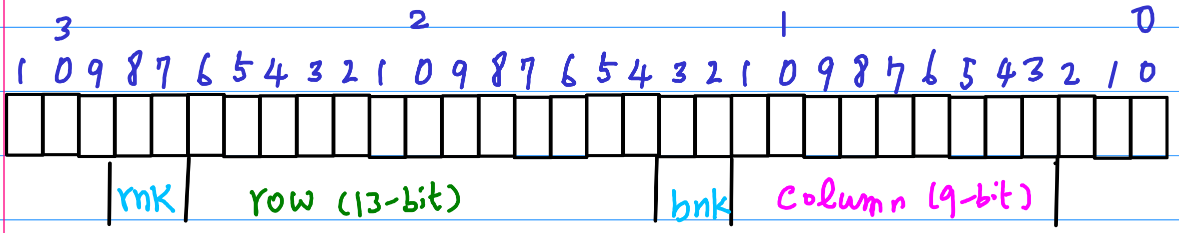
26-bit 25-bit 24-bit



26-bit



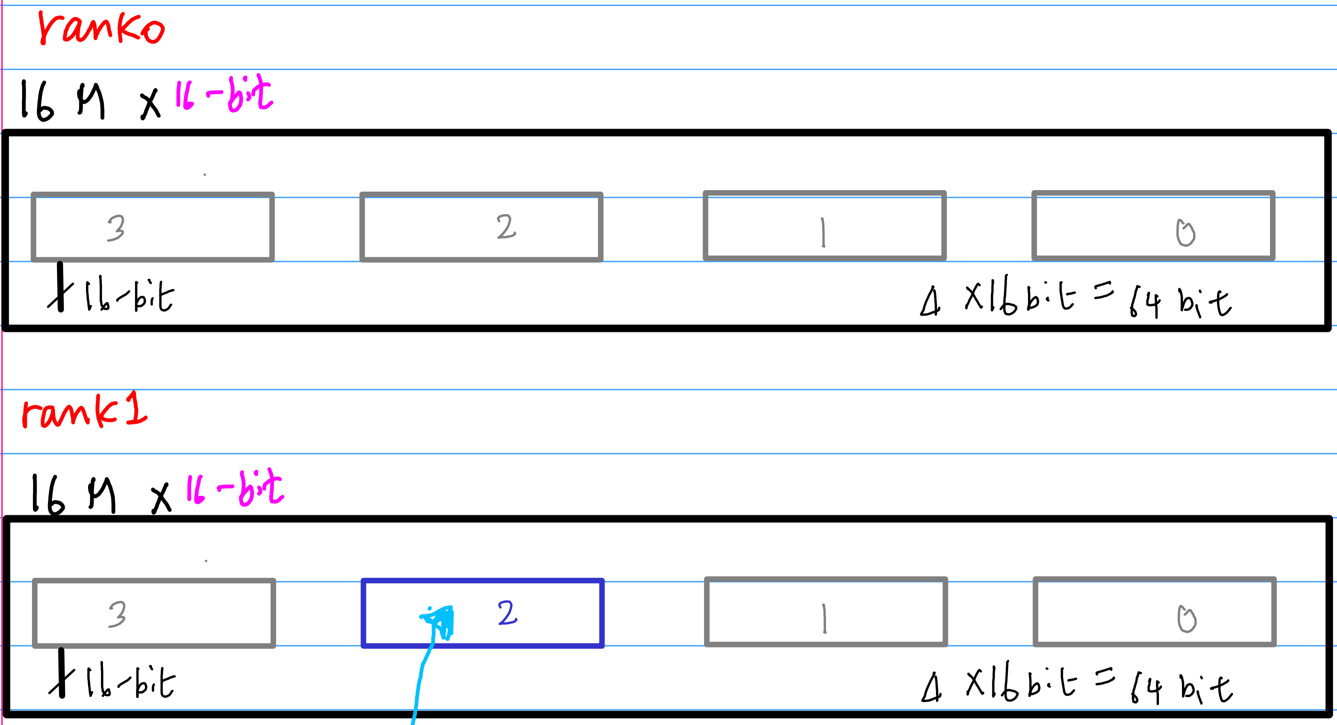
25-bit

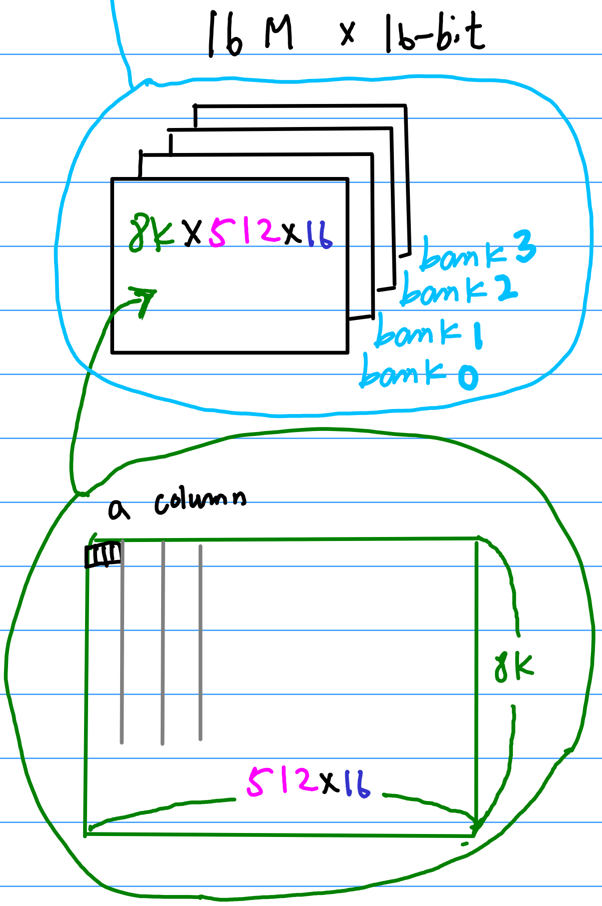


24-bit

64-bit = $\underbrace{\text{8 bytes}}_{\text{3-bit address}}$

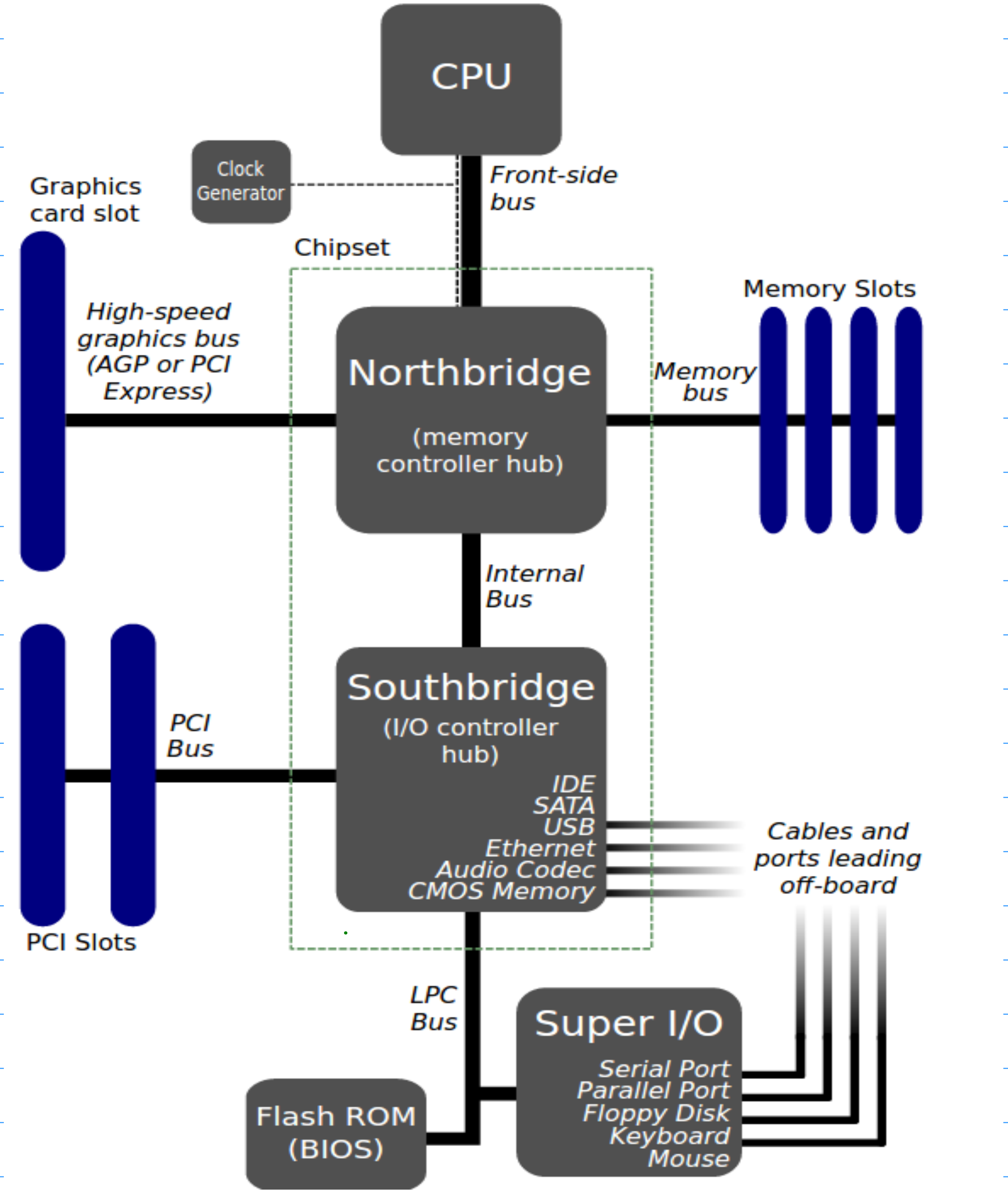
256 Mbit SDRAM - Rank





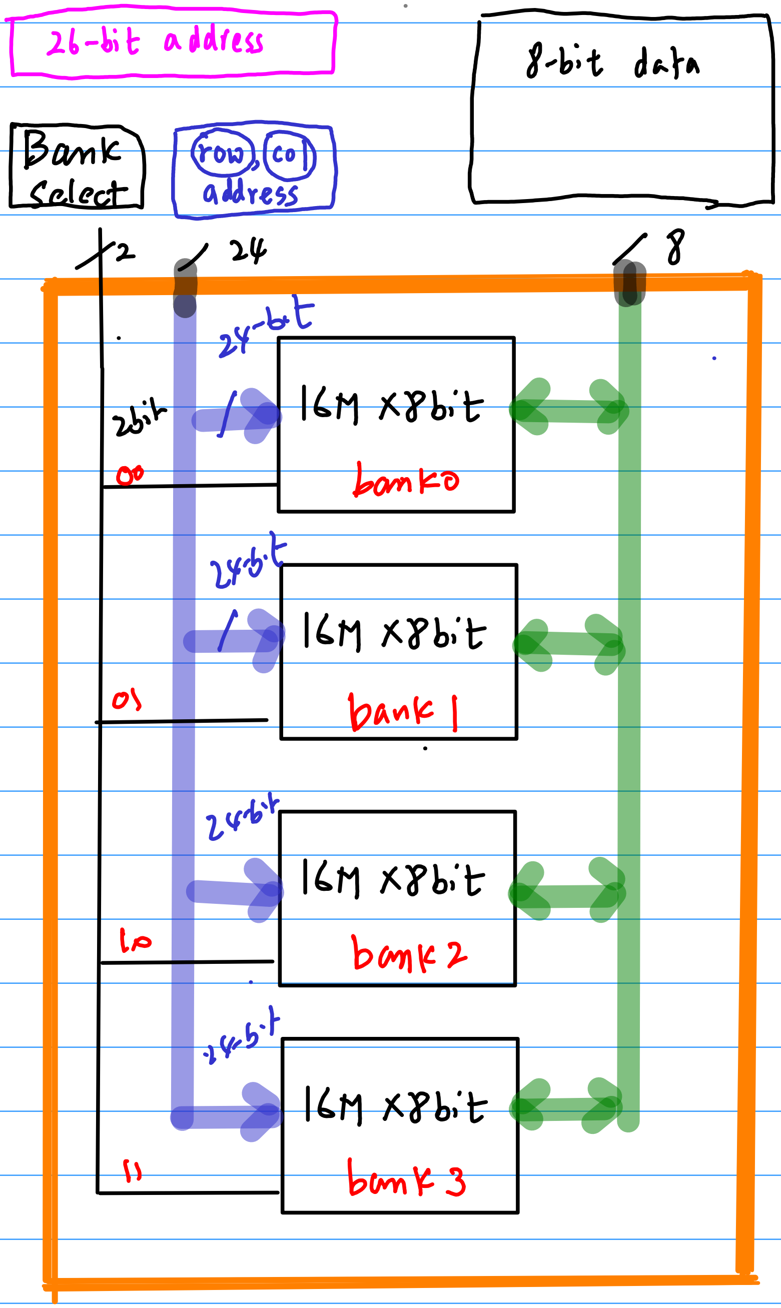
16 M x 16-bit
4 M x 16 x 4 bnk
(2+20+2) bit
8K (13-bit)
4 (2-bit)
512 (9-bit)
24-bit

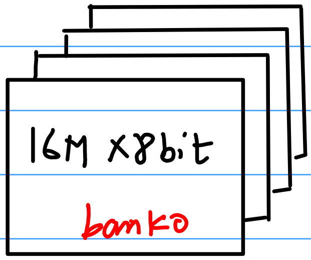
Intel Memory Controller



https://upload.wikimedia.org/wikipedia/commons/b/bd/Motherboard_diagram.svg

SDRAM



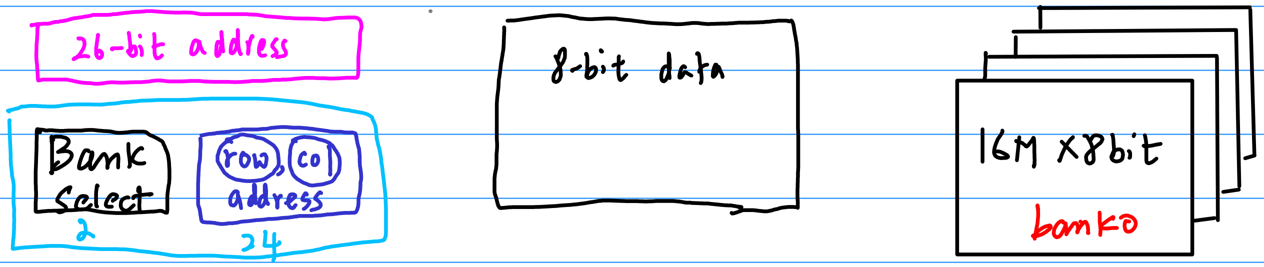


24-bit address는
동시에 4개의
bank에
병렬로 입력된다

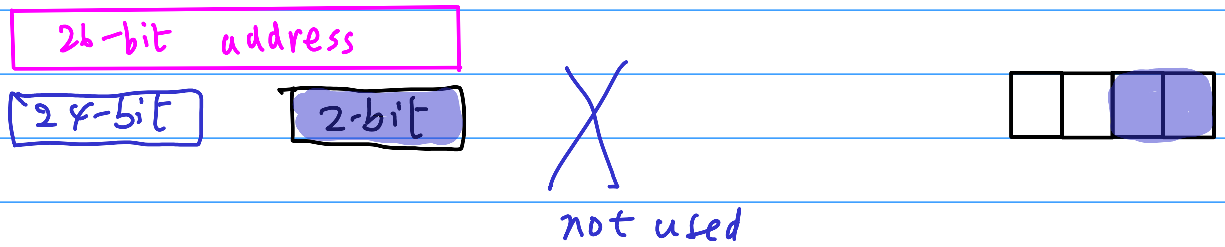
4개의 bank들이
1개의 data bus를
시간적으로 공유한다.

한번에 1개의 bank만
data bus를 사용가능

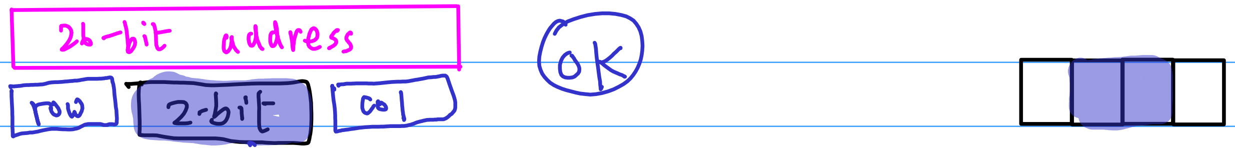
bank address positions



①

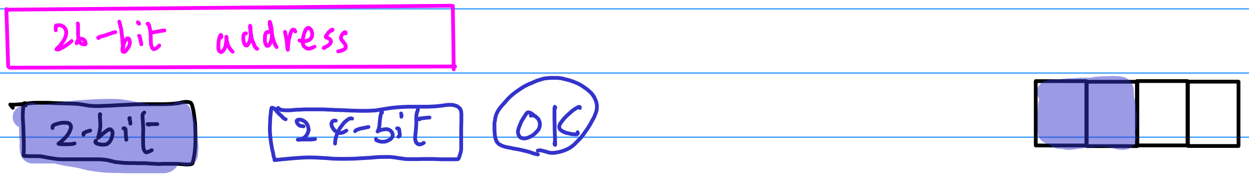


②

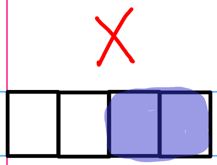


Bank Interleaving

③



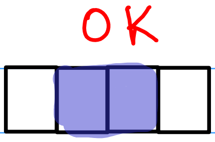
Bank Sequential



address	bank
0	b0
1	b1
2	b2
3	b3
4	b0
5	b1
6	b2
7	b3
8	b0
9	b1
10	b2
11	b3
12	b0
13	b1
14	b2
15	b3

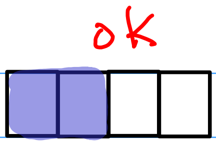
b0	b1	b2	b3
0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

fragmentation



address	bank
0	b0
1	b0
2	b1
3	b1
4	b2
5	b2
6	b3
7	b3
8	b0
9	b0
10	b1
11	b1
12	b2
13	b2
14	b3
15	b3

b0	b1	b2	b3
0	2	4	6
1	3	5	7
8	10	12	14
9	11	13	15



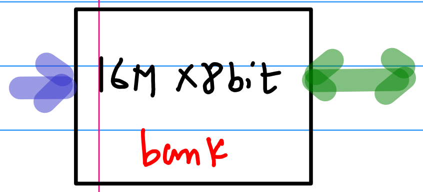
address	bank
0	b0
1	b0
2	b0
3	b0
4	b1
5	b1
6	b1
7	b1
8	b2
9	b2
10	b2
11	b2
12	b3
13	b3
14	b3
15	b3

b0	b1	b2	b3
0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

bank interleaving

Bank Architecture

24-bit address



$16M = 2^4 \cdot 2^{20} = 2^{24}$

$\boxed{2^{24} = 2^{13} \cdot 2^{11}}$

row　　col

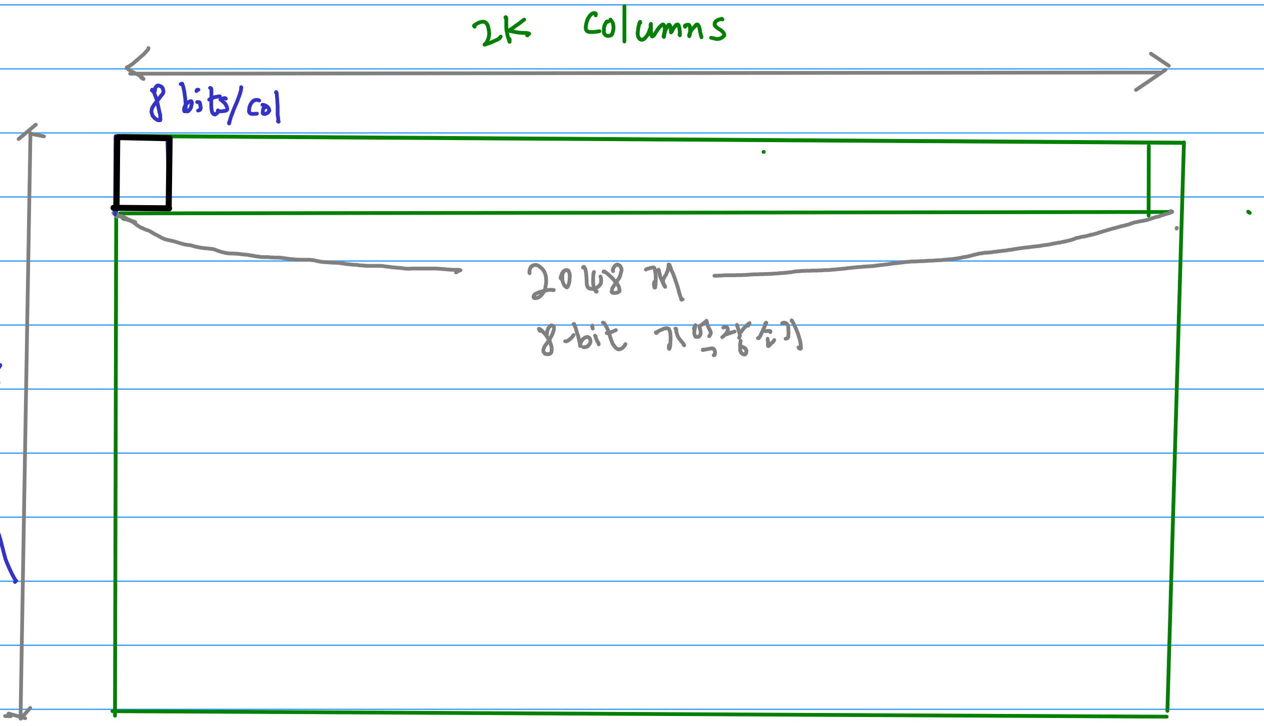
13-bit　$\boxed{2^3 \cdot 2^{10} = 8K}$　　$\boxed{2^1 \cdot 2^{10} = 2K}$　11-bit

13-bit
11-bit
24-bit
+ 2
26-bit

16Kbit = 2K x 8-bit linear memory array

$\frac{1024}{8}$

[illegible]



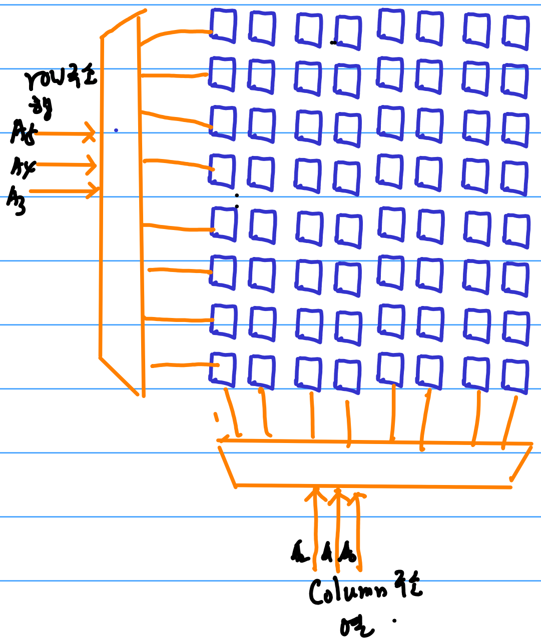
row
13-bit

8K rows
$= 2^{13}$
$= 8192$개

SDRAM

DRAM과 같이 Row & Col Address Decoder 사용

Row & Col address 사용



→ 1-bit data access 가능

8-bit data access를 위해서는 8개 copy가 필요

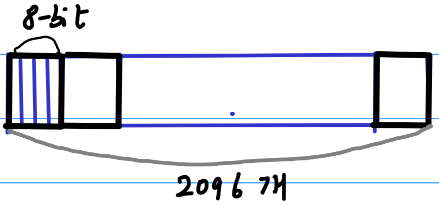
$$\boxed{2^{24} = 2^{13} \cdot 2^{11}}$$

row　　col

13-bit　$2^3 \cdot 2^{10} = 8K$　　$2^1 \cdot 2^{10} = 2K$　11-bit

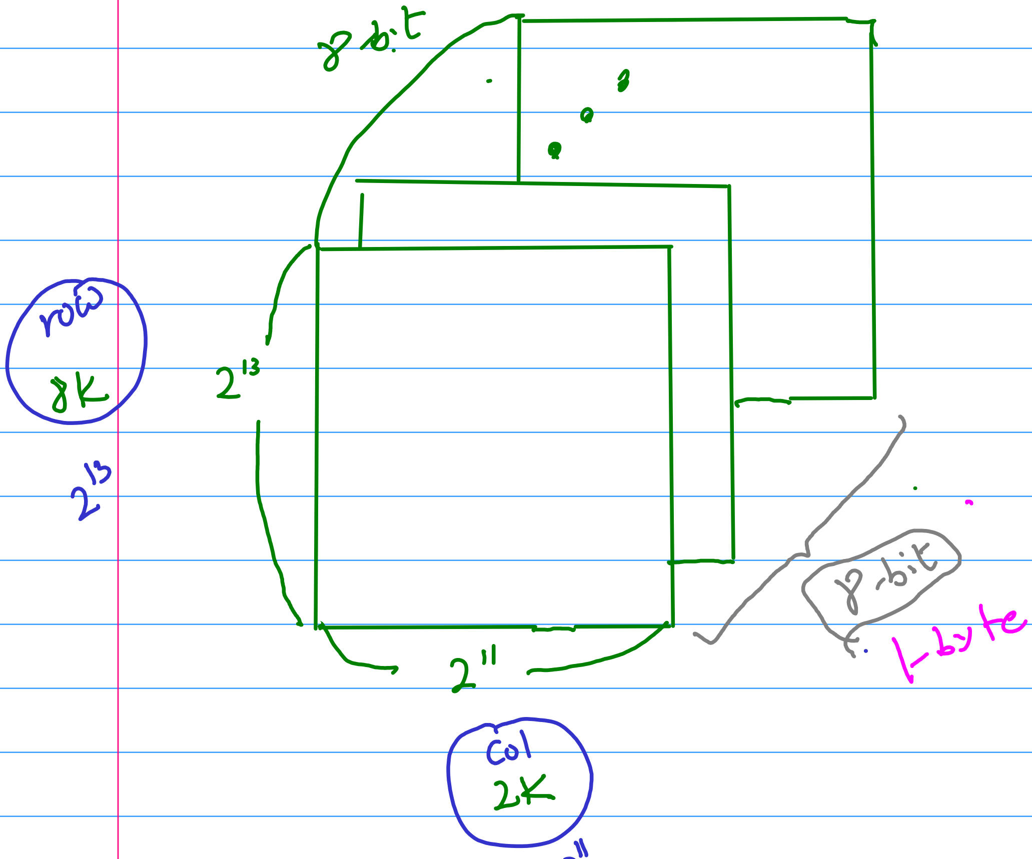
- 총 8K개의 row가 있다.
- 각 row에는 16K bit가 저장된다.

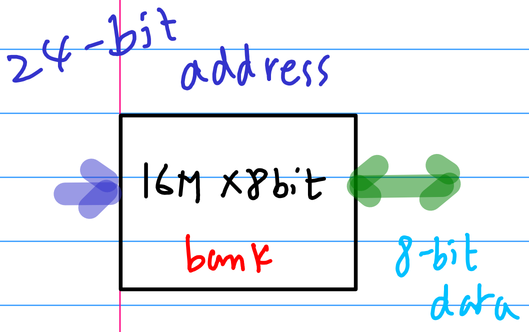
16Kbit = $\boxed{2K \times 8\text{-bit}}$



data bus가 8-bit 이므로
16Kbit / 8-bit = 2K개가 있다.

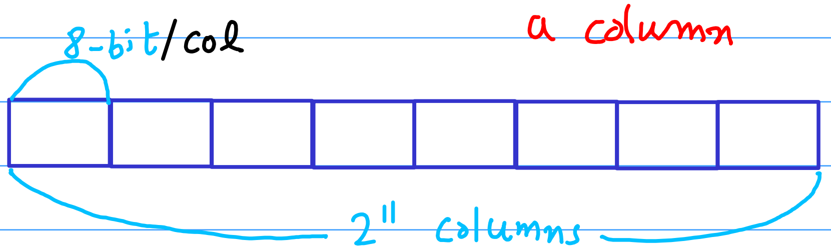
개념적인 그림



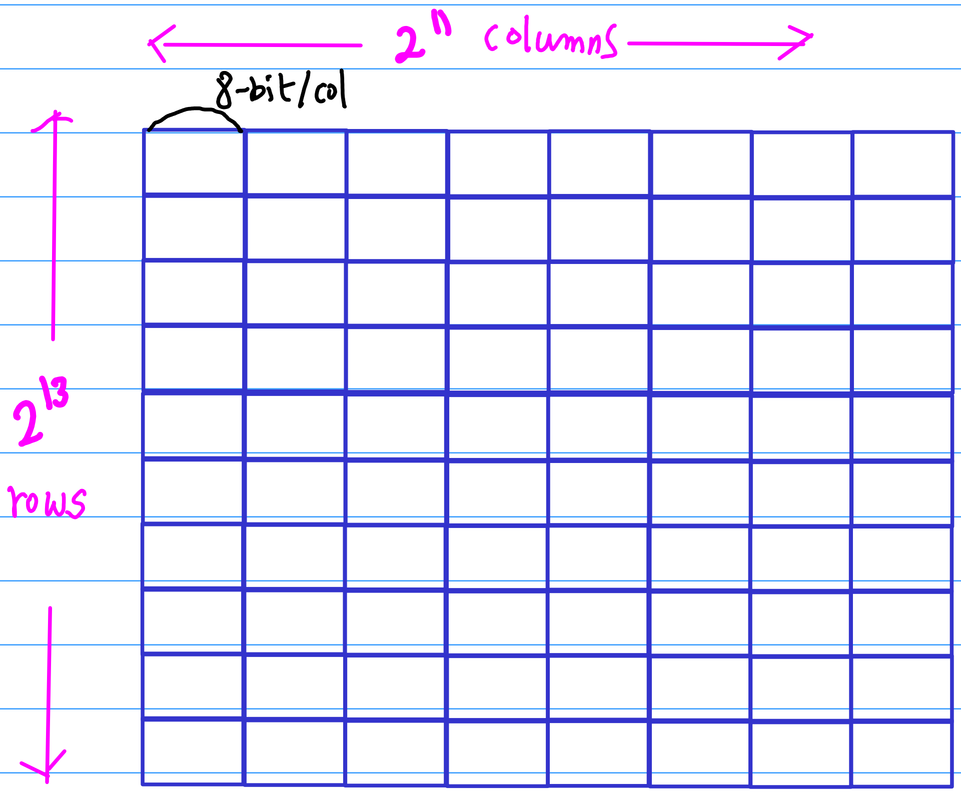


$$\boxed{2^{24} = \underbrace{2^{13}}_{row} \cdot \underbrace{2^{11}}_{col}}$$

$$\boxed{2^3 \cdot 2^{10} = 8K} \qquad \boxed{2^1 \cdot 2^{10} = 2K}$$



2^{11} x 8-bit



① Activate row

② RD/WR col

③ Precharge

$\underbrace{16M}_{2^4 \cdot 2^{20}} \times 8\text{-bit}$

$= 2^{24} = 2^{13} \cdot 2^{11}$

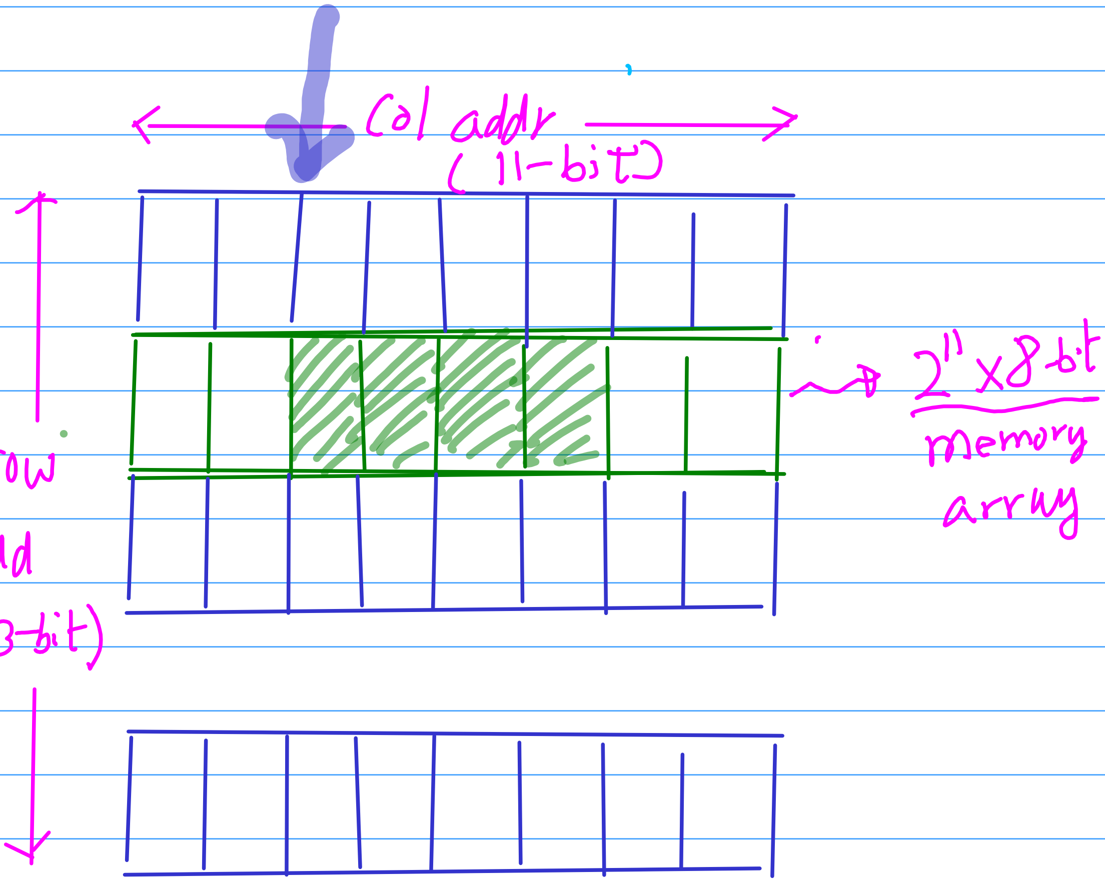
② col Address를 선택한다

③ burst (sequential address)

① 먼저 access할 Row를 선택 (Row Addr)

"ACTIVATE"

time delay

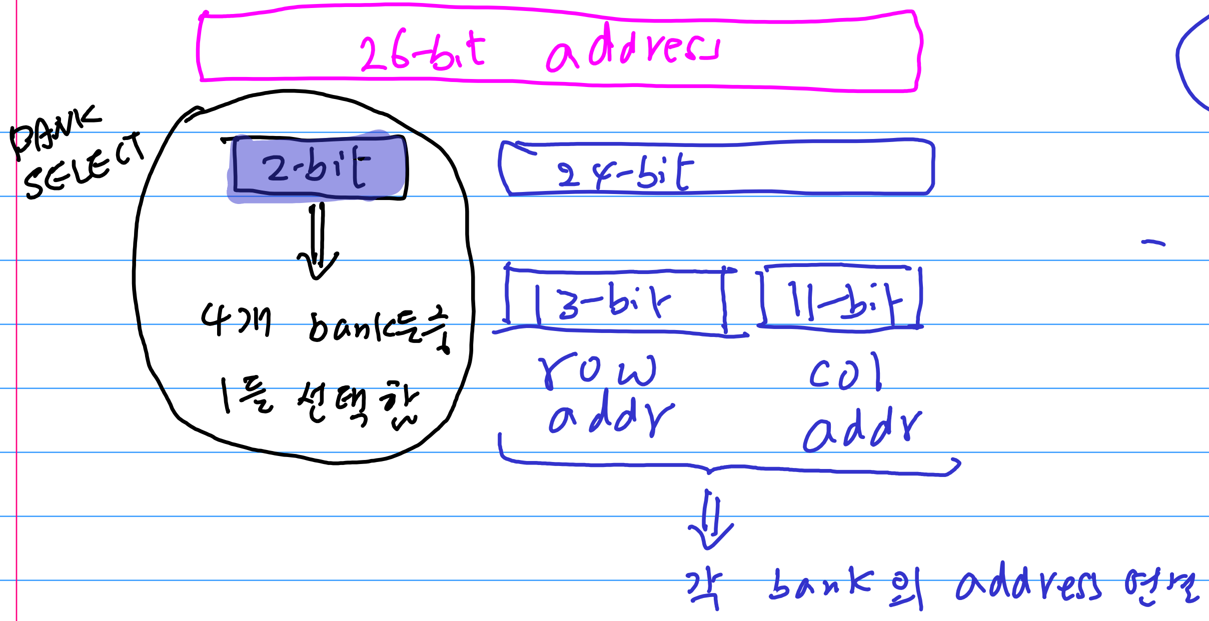


Bank Interleaving

allows operations to 2 ~ 4 banks simultaneously

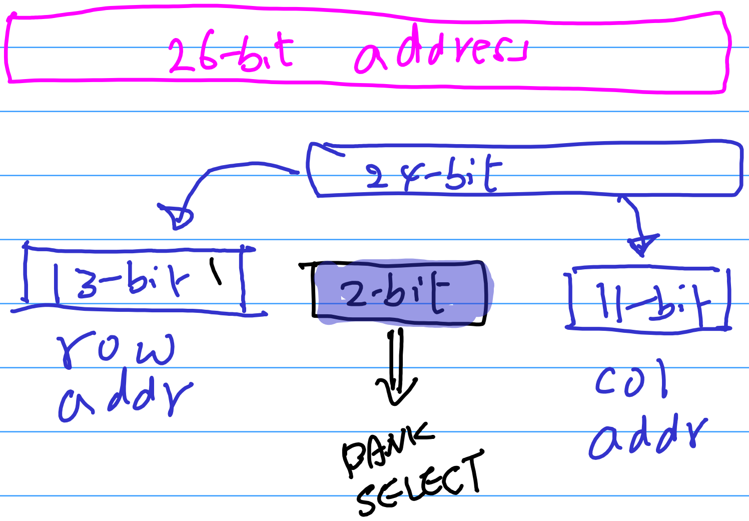
independent memory arrays
inside of a DRAM chip

①



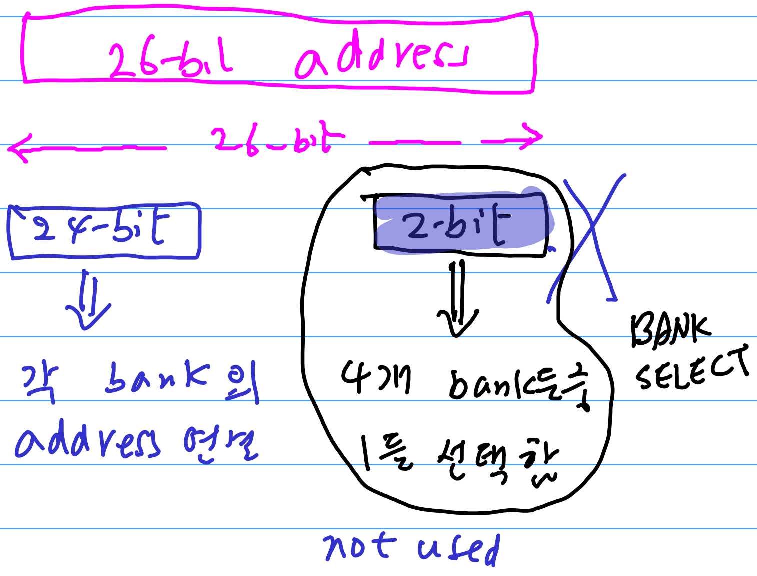
OK

②



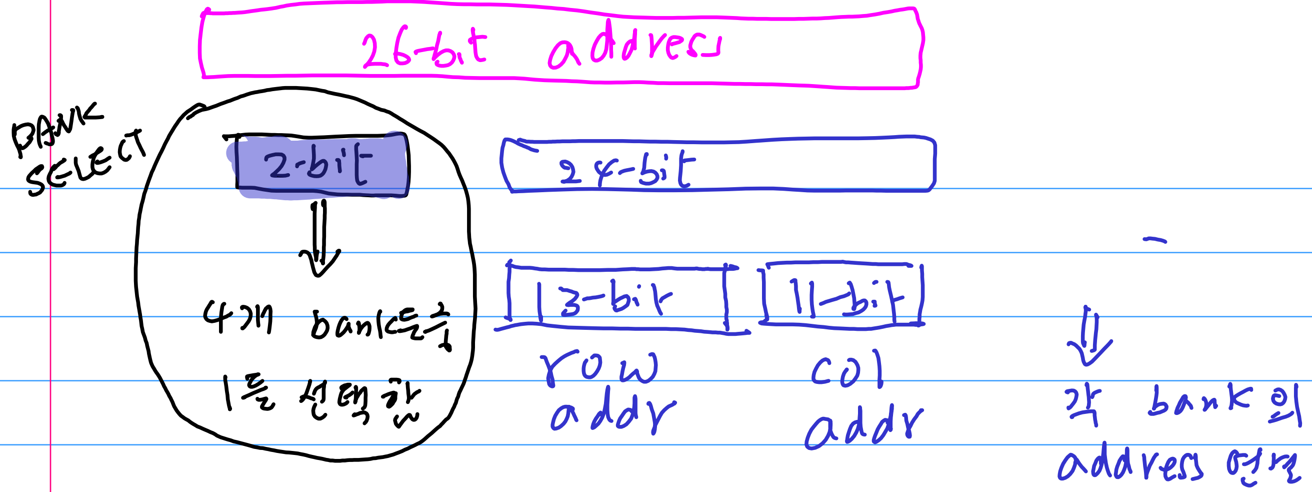
OK

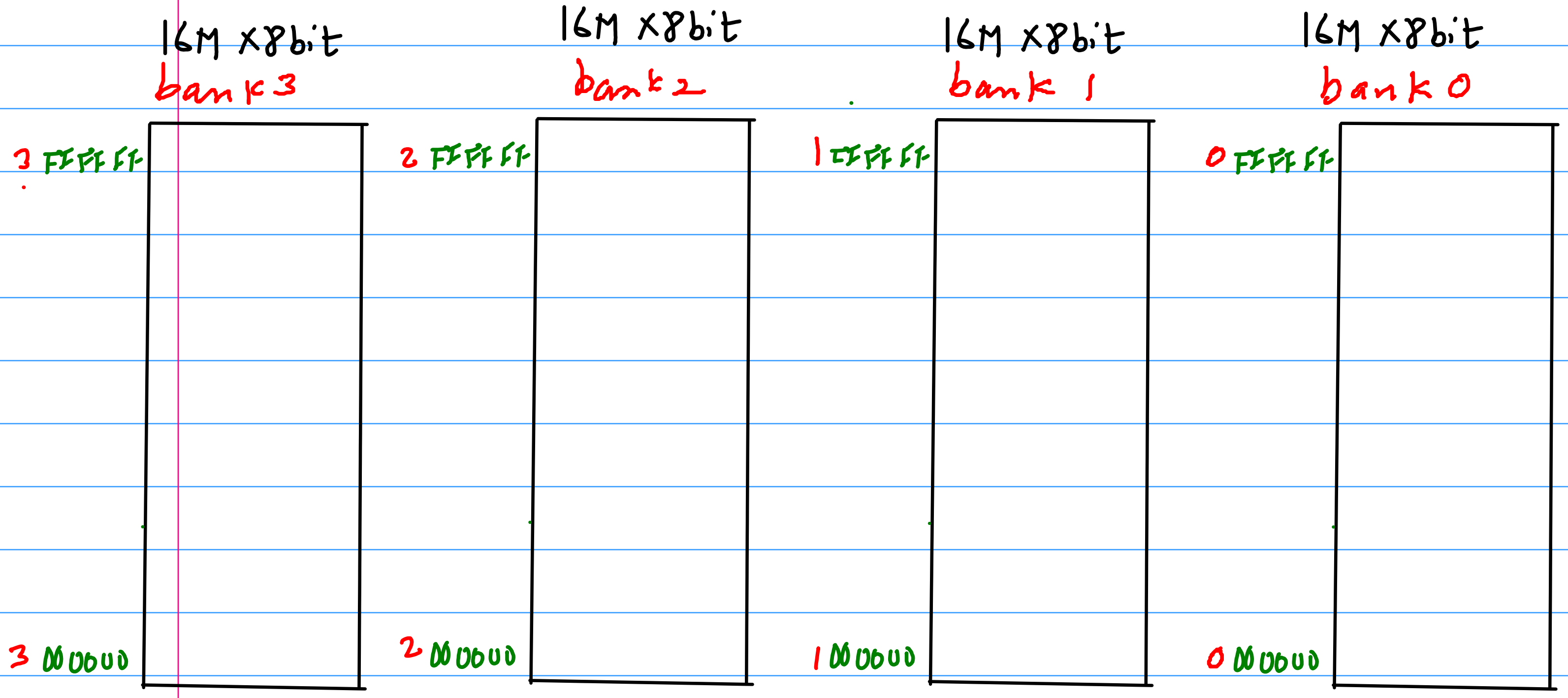
③

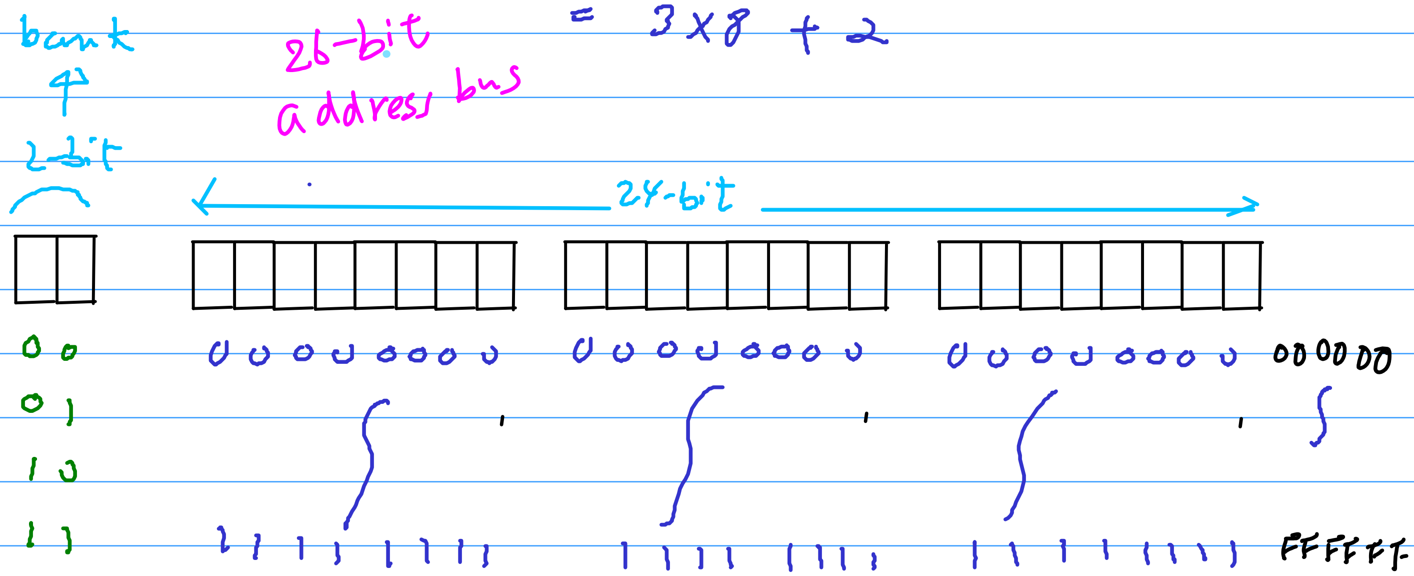


X not used

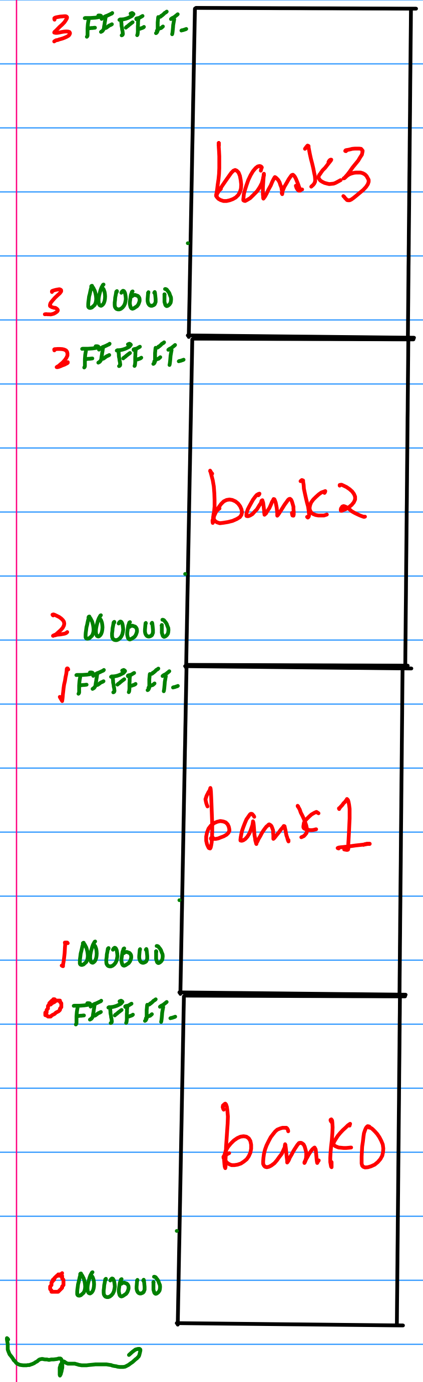
①

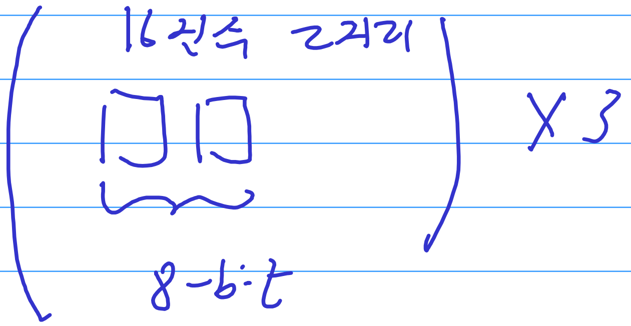

26-bit address
BANK SELECT
2-bit
4개 bank들중 1를 선택함
24-bit
13-bit
11-bit
row addr
col addr
⇓
각 bank의 address 연결


16M x8bit
bank 3
16M x8bit
bank 2
16M x8bit
bank 1
16M x8bit
bank 0
3 FFFFFF
3 000000
2 FFFFFF
2 000000
1 FFFFFF
1 000000
0 FFFFFF
0 000000


bank
2-bit
26-bit
address bus
= 3x8 + 2
24-bit
0 0
0 1
1 0
1 1
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
000000
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1
FFFFFF

$2^{26} = 2^{6} \cdot 2^{20} = 16M$



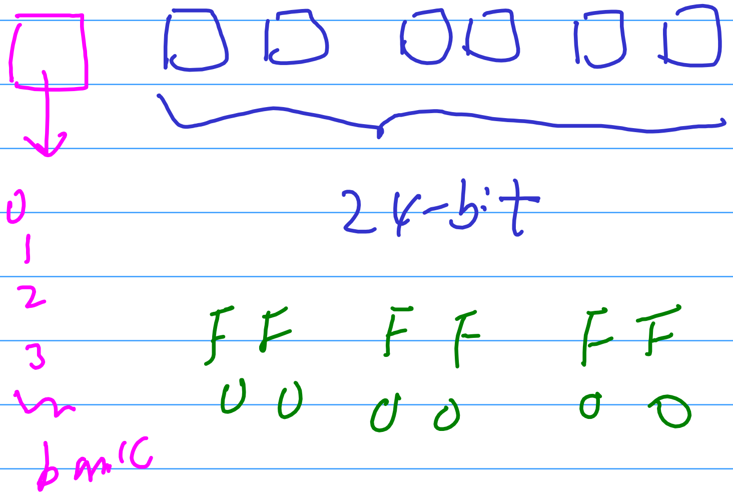


26-bit

$\underline{2} + \underline{24}$

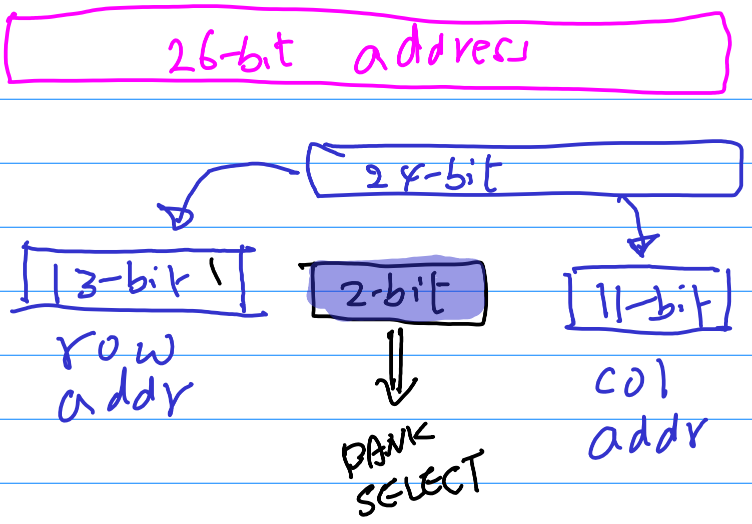
bank row + col

(13) · (11)

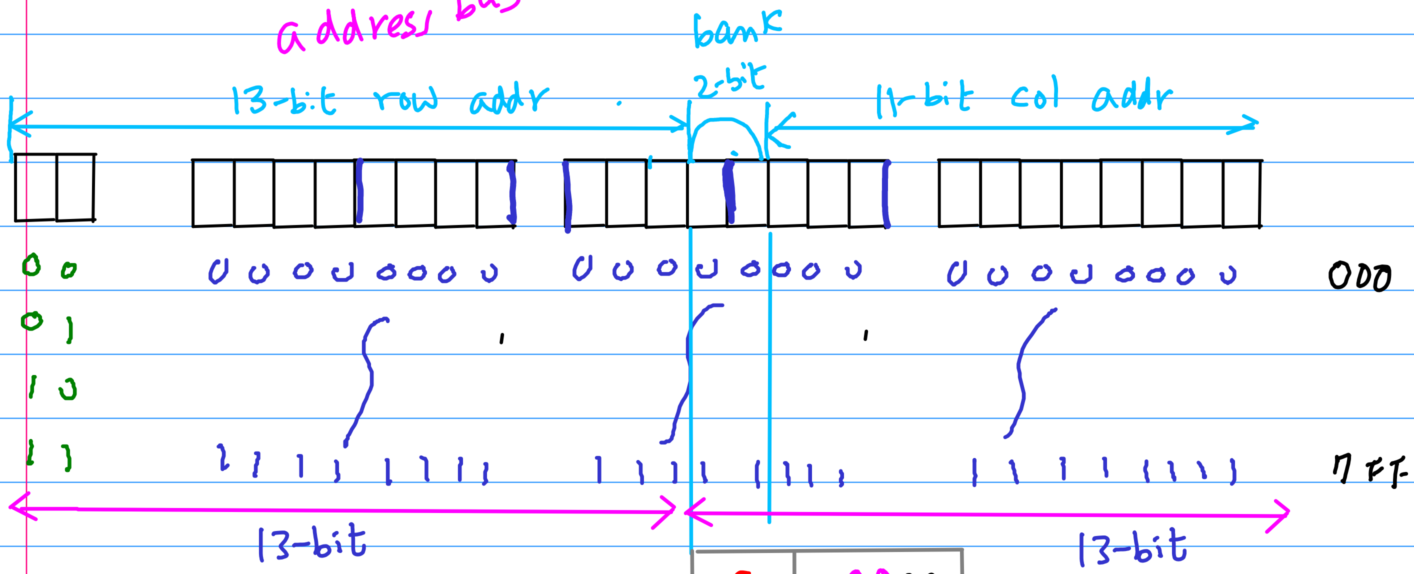


2

OK



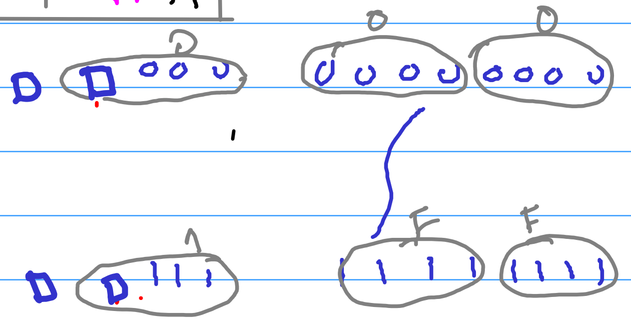
26-bit address bus = 3x8 + 2



0	0000
	07FF
1	0800
	0FFF
2	1000
	17FF
3	1800
	1FFF

bank

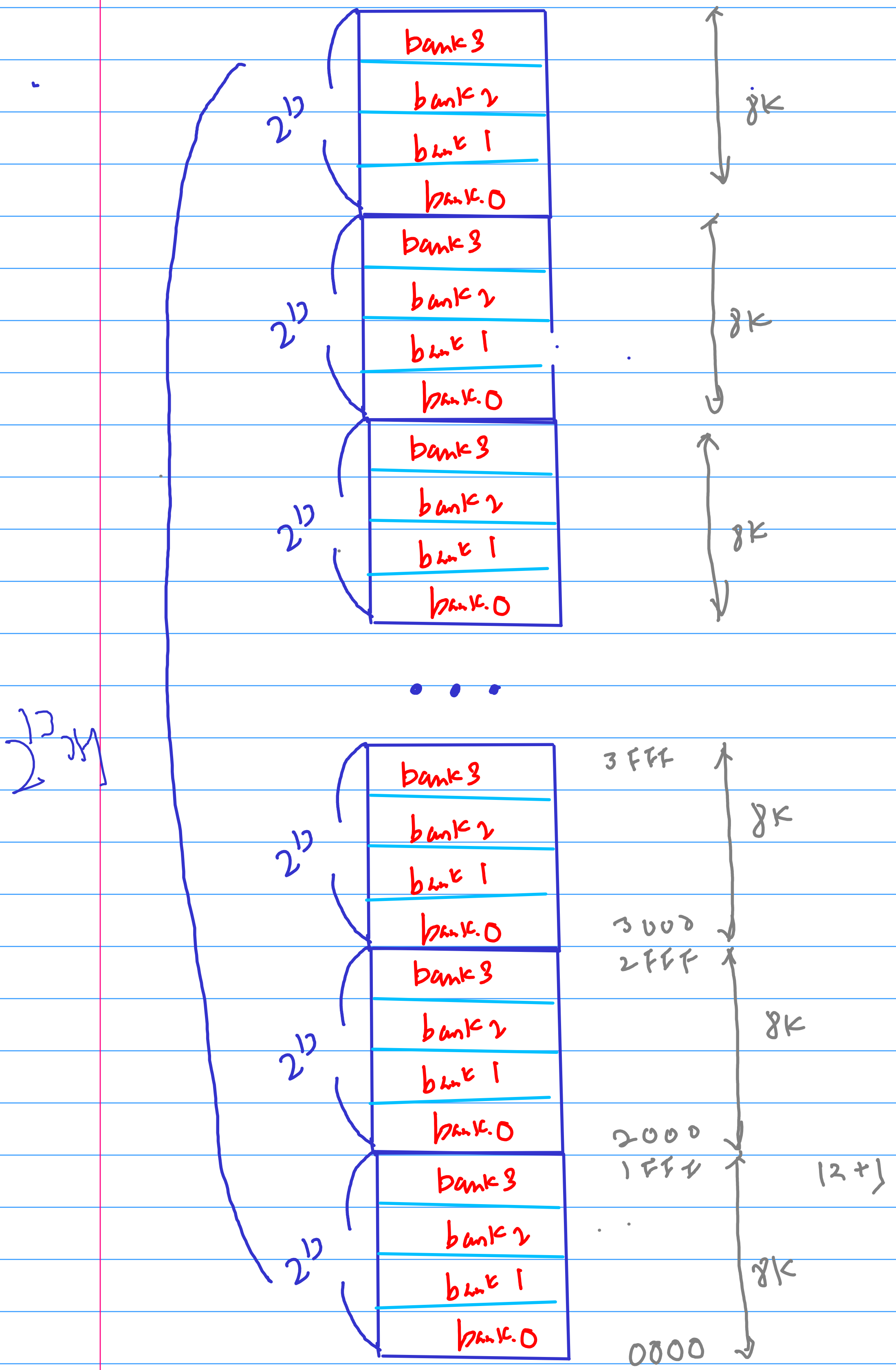
0	00
1	01
2	10
3	11



$\boxed{\text{26-bit address}}$

$2^{26} = 2^6 \cdot 2^{20} = 64M$

$2^{13} = 2^3 \cdot 2^{10} = 8K$



$\underline{2^{13}} \cdot \underline{2^{13}} = 2^{26}$

2^{13+13}

$8K \cdot 8K$
$= 64M$
$= 2^6 2^{20}$

③

26-bit address

X not used

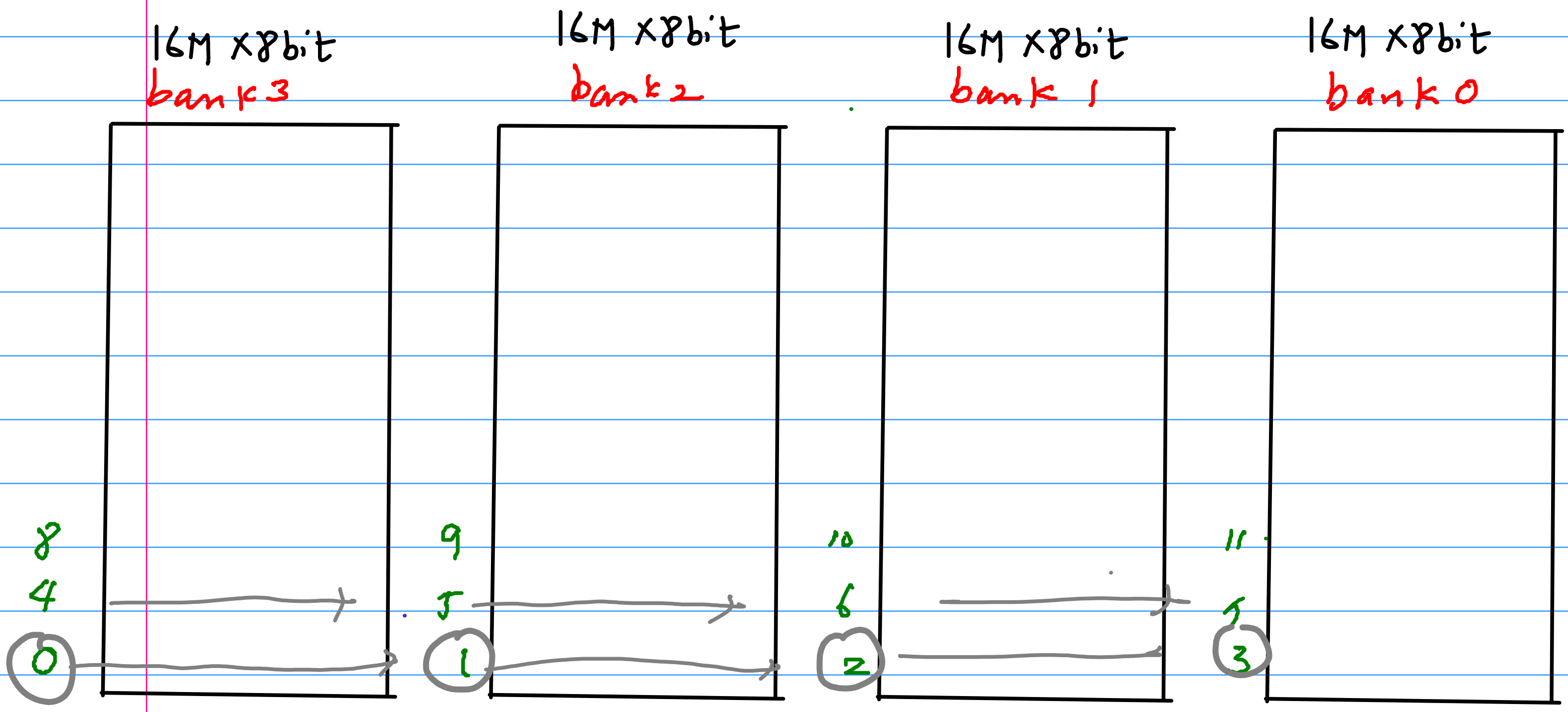
← 26-bit →

24-bit
⇓
각 bank의
address 연결

2-bit
⇓
4개 bank들중
1를 선택함

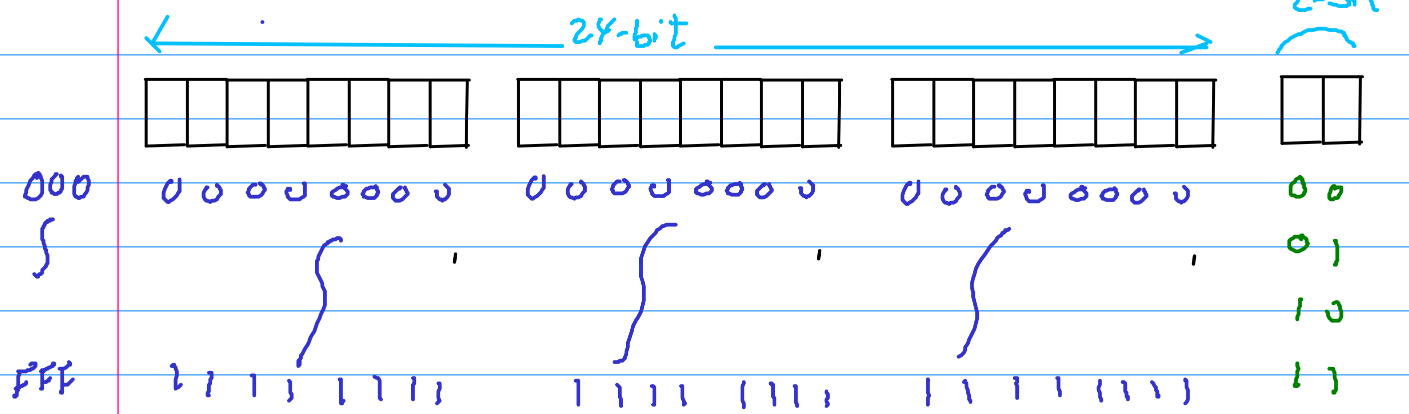
BANK
SELECT

not used



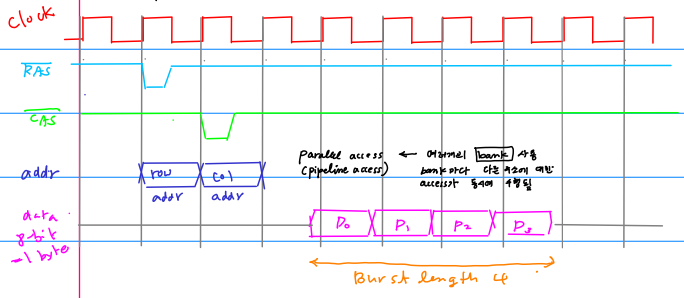
26-bit
address bus = 3x8 + 2

bank
↑
2-bit

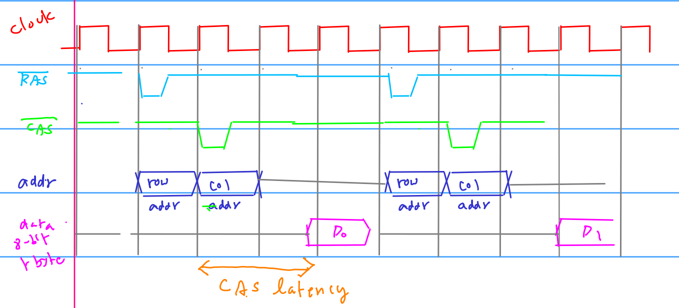


.

Burst Read Mode

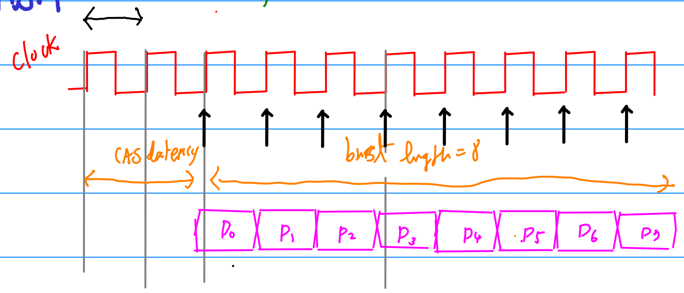


Parallel access (pipeline access) ← 여러개의 bank 사용
bank마다 다른 주소에 대한 access가 동시에 수행됨



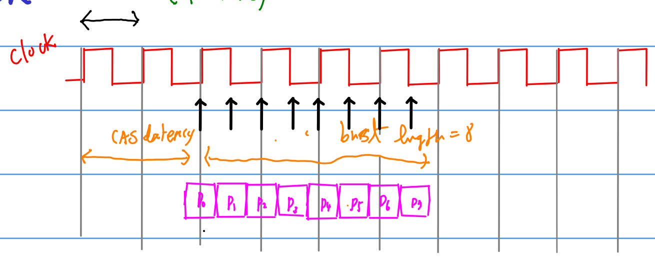
SDRAM

7.5ns. (133MHz)



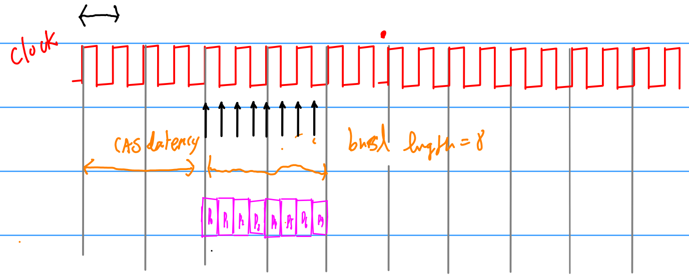
DDR

7.5ns. (133MHz)

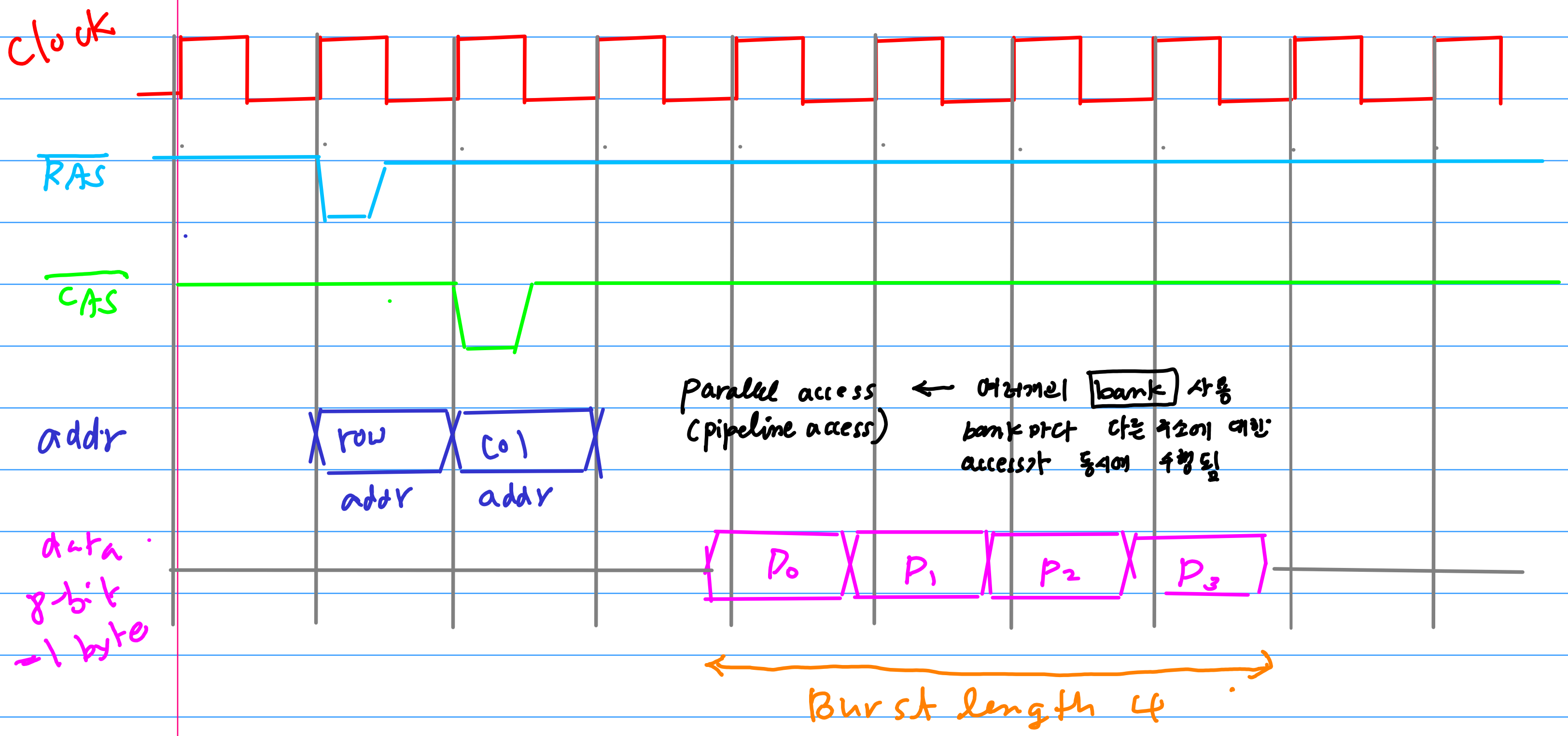


DDR2

7.5ns/2 = 3.75ns (266MHz)

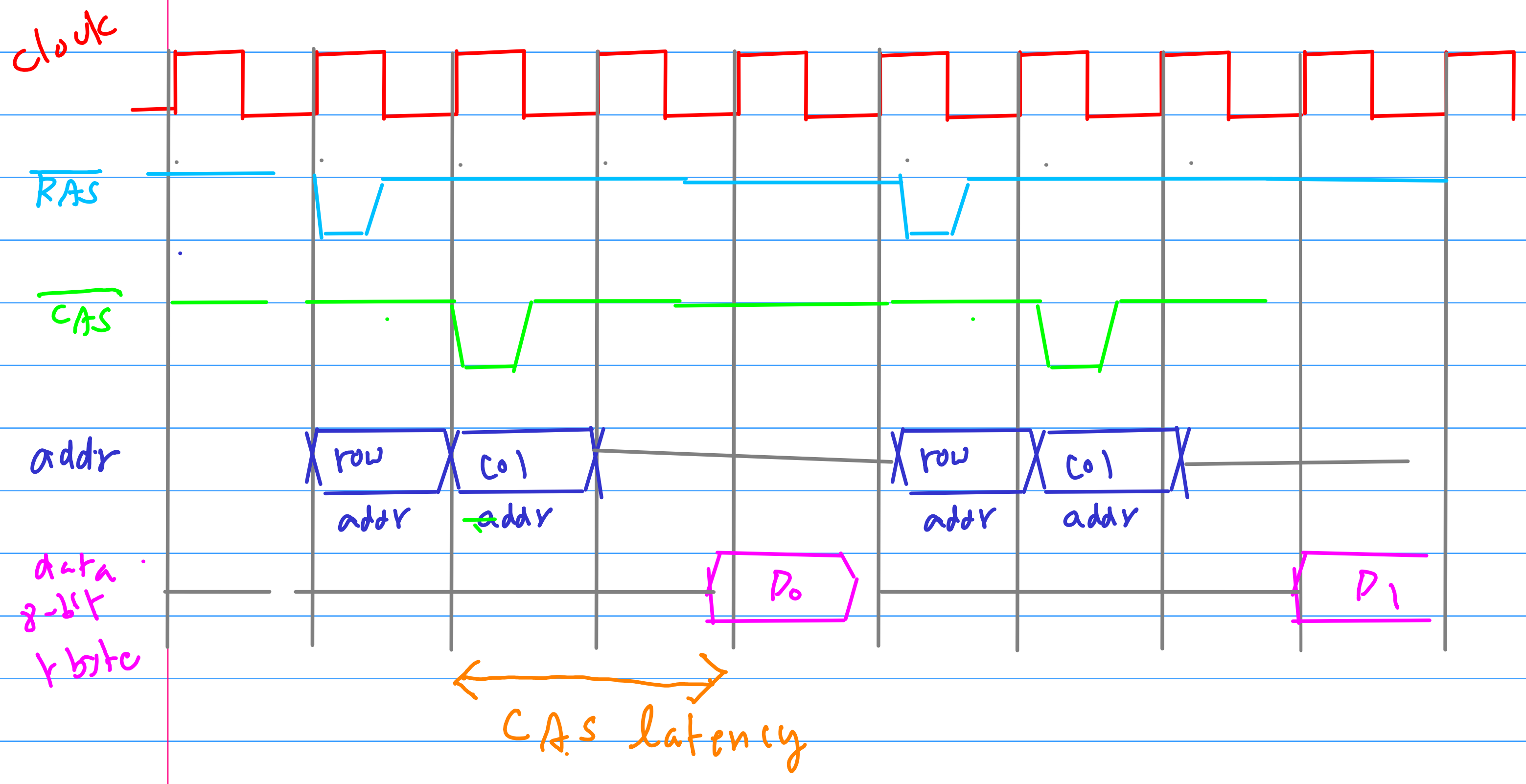


Burst Read Mode



Parallel access ← 여러개의 bank 사용
(pipeline access)

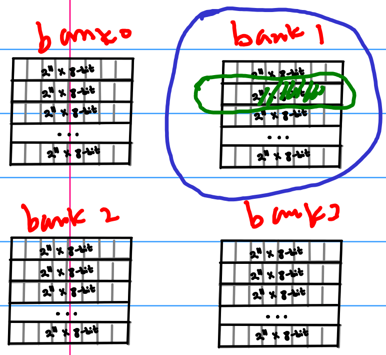
bank마다 다른 주소에 대한
access가 동시에 수행됨



각 Bank에서 병렬로 col addr, row addr
access하므로 CAS latency hide

Burst Mode : 연속 전송동작

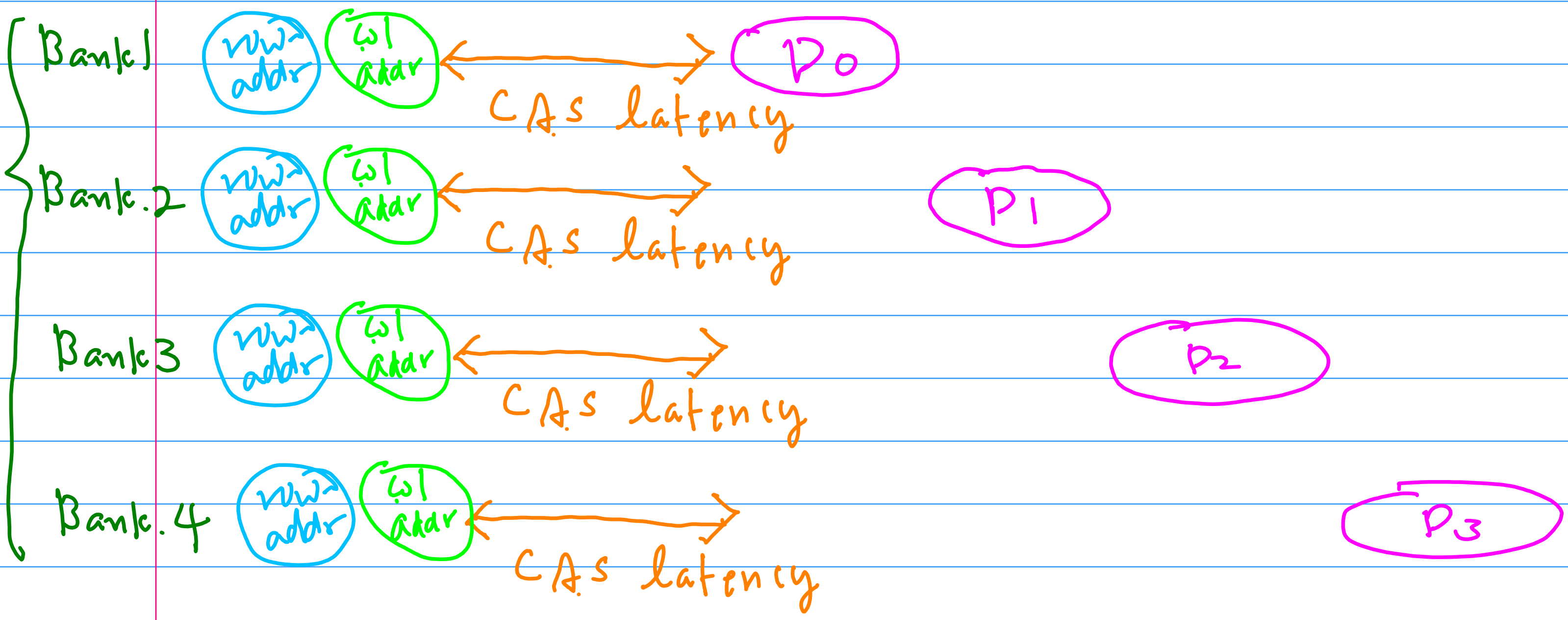
같은 bank 내의 같은 row에 있는



한번 읽을 때 여러개의 Byte 들을 연속적으로 전송할 수 있다

$$\underbrace{P_0,\ P_1,\ P_2,\ P_3}_{\text{burst length} = 4}$$

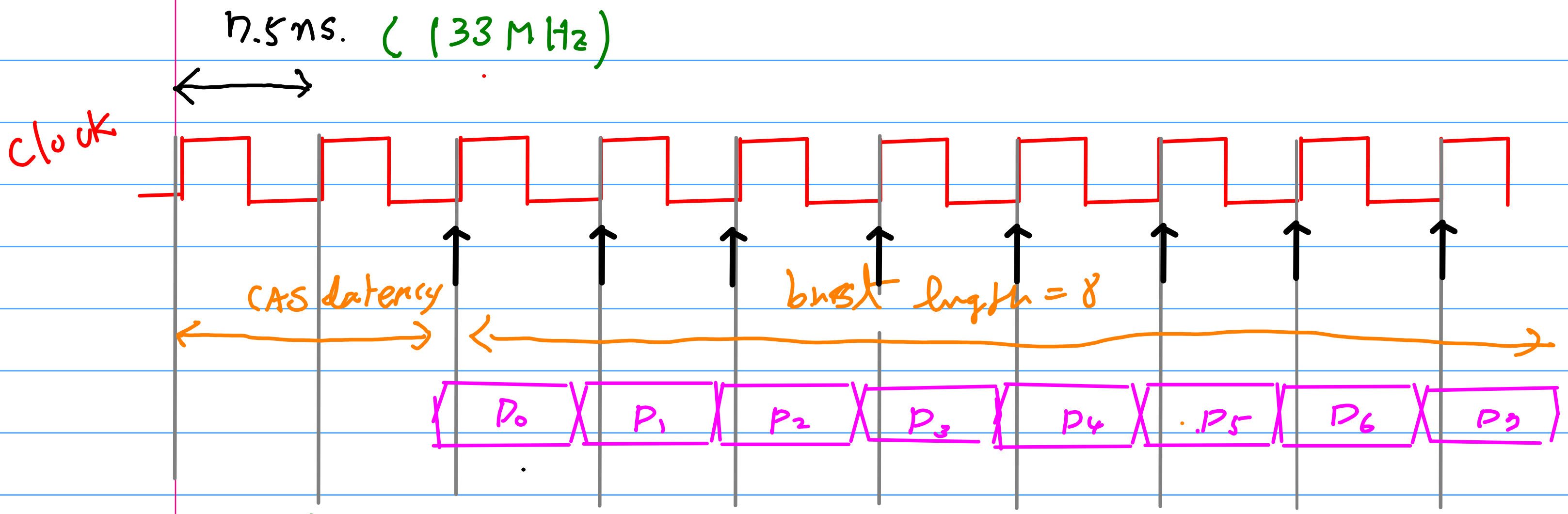
각 Bank에서 병렬로 col addr, row addr access하므로 CAS latency hide



DDR (Double Data Rate)

SDRAM PC133 ~ 133 MHz

$$\frac{1}{133 \times 10^6} \text{ sec} = \frac{1}{133} \mu\text{sec} = 0.0075 \ \mu\text{sec} = 7.5 \text{ nsec}$$



8 bit x8
=64bit

64-bit data/cycle

7.5 ns 주기 clock rising edge 마다 64-bit data가 전송

burst Length = 8 byte

8 byte · 전송 끝 마치려면

8×7.5 = 60ns 필요

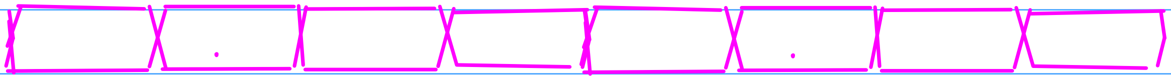
CAS latency 2×7.5 = 15 ns

75 ns.

SDRAM (Single Data Rate)

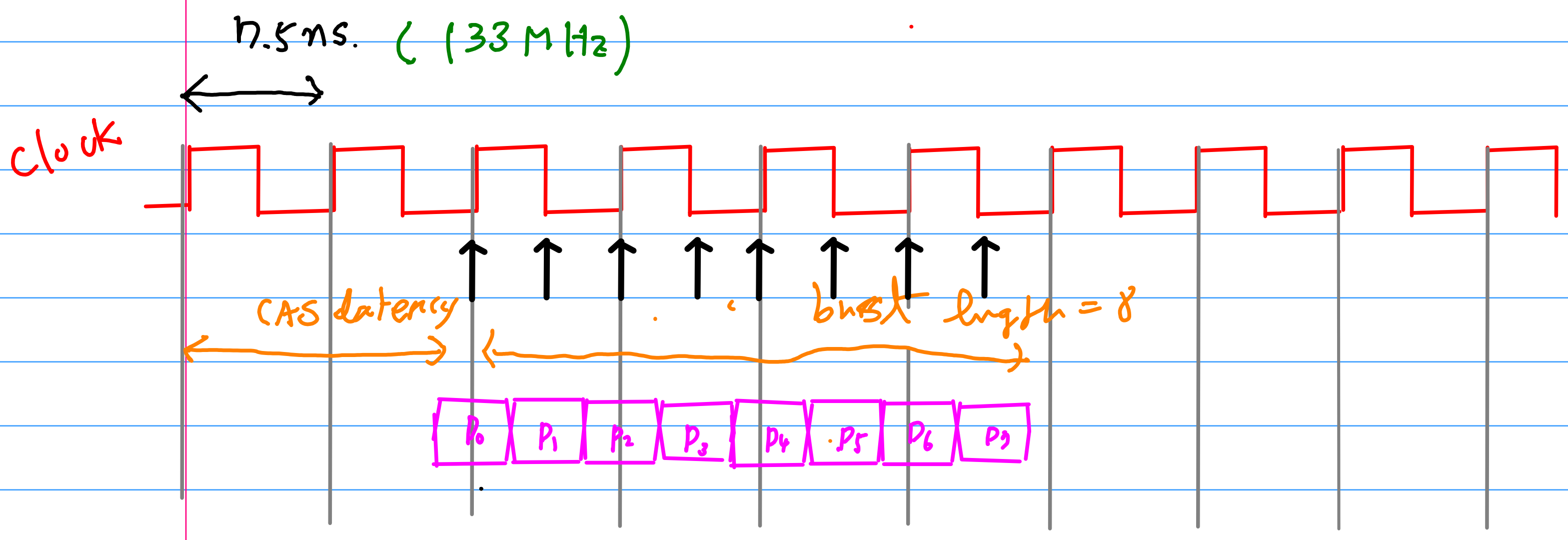
L synchronous

Single



DDR (Double Data Rate)

DDR-266 Clock는 133MHz x 2



{ rising edge
falling edge

CAS latency 2×7.5

data transf $60/2 = 30$

$\overline{}$ 45ns

DDR-266 실제 clock는 133MHz

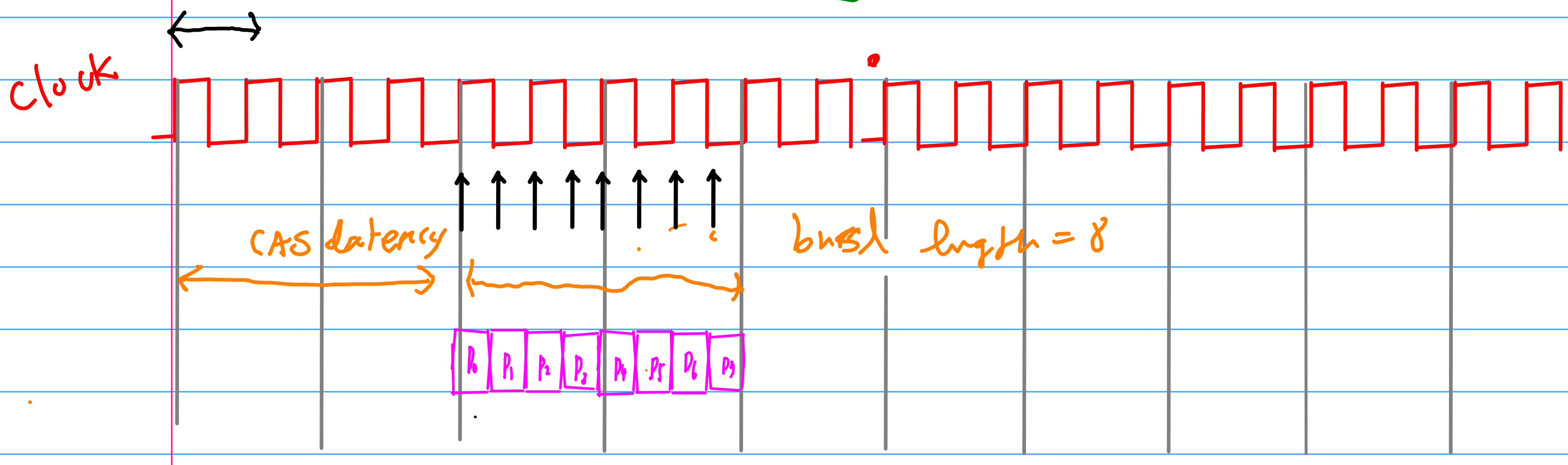
DDR-333 실제 clock는 166MHz

DDR-400 실제 clock는 200MHz

DDR 2

2배 빠른 클럭 freq를 사용하는
DDR SDRAM.

7.5ns/2 = 3.75ns (266MHz)



CAS latency   4 × 3.75 = 15

data transfer   30/2 = 15

30 ns

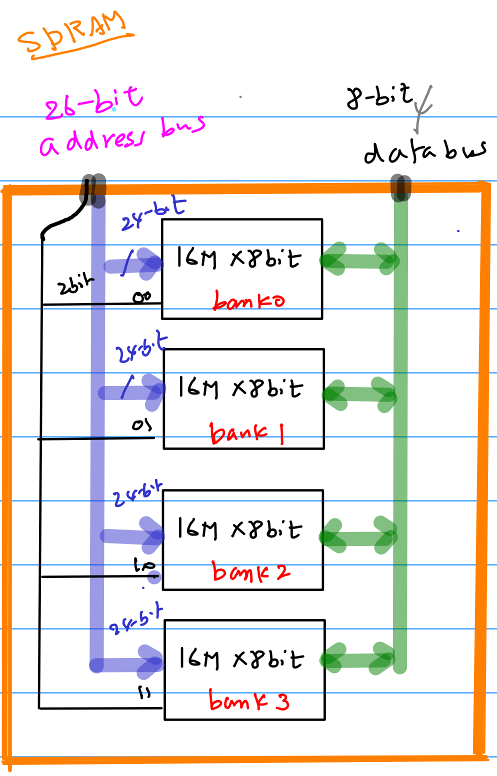
google

Computer Organization in plain view

wikiversity.org

(The necessities in Computer Organization

wikiversity.org)

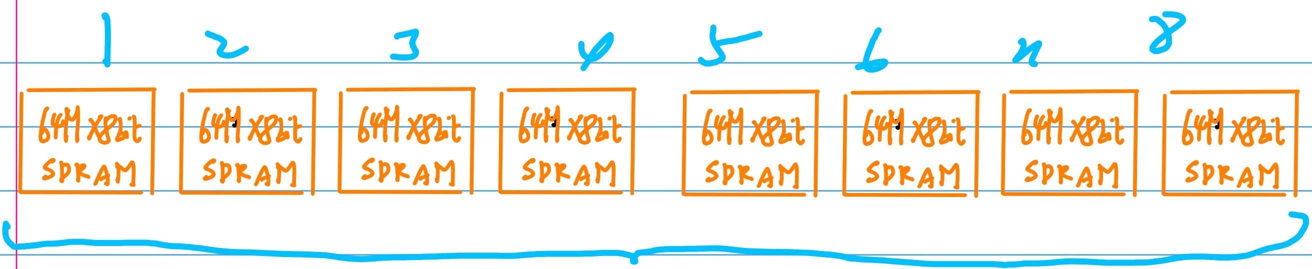


$2^{24} = 2^6 \cdot 2^{20}$

64M x8bit
SDRAM

칩당 8-bit 입출력

64-bit PC



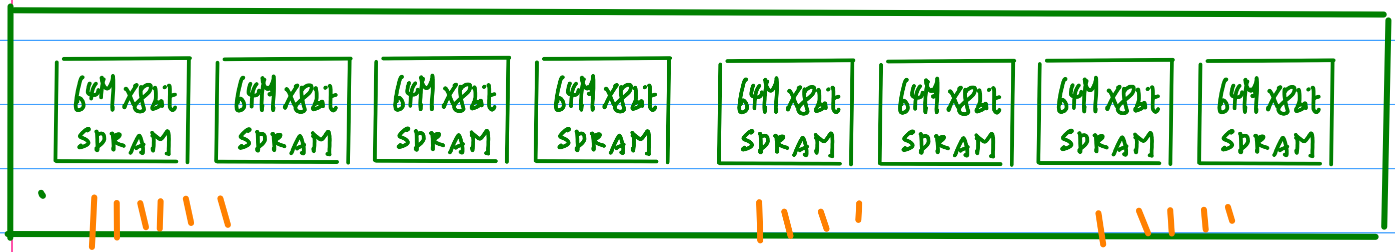
64-bit 읽고 쓰고

$$8 \times 64M \times 8bit$$

$$\underbrace{2^3 \cdot 2^{16}}_{2^{19}} \cdot M \times 8bit$$

$$M \times 8bit$$

512M x8-bit



SIMM (Single In-line Memory Module)

한면

DIMM (Dual In-line Memory Module)

양면

SDRAM : 2Gbit = 256M x 8bit

$$2\cdot2^{30} = 2^{8}\cdot2^{20}\cdot2^{3}$$

SIMM = 8 x SDRAM = 8 x 256M x 8 bit

= 8 x (2G Bit)

= 16 G Bit

= 2 G Byte

memory rank

일반적으로 CPU — Memory 64-bit data 전송

데이터 입출력 width가 64-bit 단위가 되도록
구성된 module을 memory rank

JEDEC 정의 데이터 입출력 폭이) 64-bit가
여러개의 메모리칩을 사용하여
구성한 기억장치 module

각 rank에서 address 마다
64-bit씩 read / write 한다

SDRAM · (□ x 4bit) 16개

= □ x 64bit

SDRAM · (□ x 8bit) 8개

= □ x 64bit

single rank module – 64-bit

rank가 한개

dual rank module – 64-bit X 2.

rank가 두개