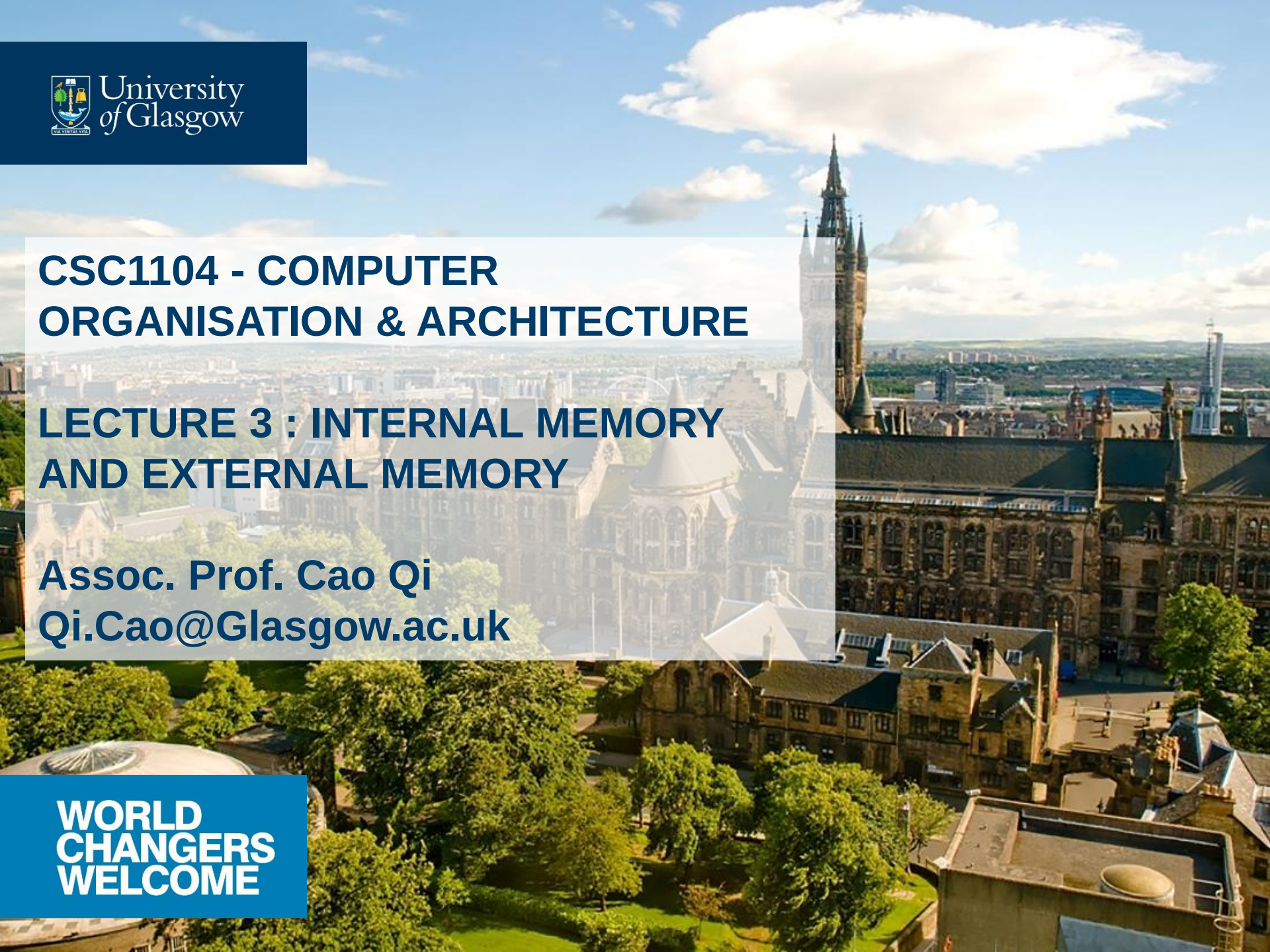


University of Glasgow

CSC1104 - COMPUTER ORGANISATION & ARCHITECTURE

LECTURE 3 : INTERNAL MEMORY AND EXTERNAL MEMORY

Assoc. Prof. Cao Qi
Qi.Cao@Glasgow.ac.uk

WORLD CHANGERS WELCOME

University of Glasgow

School of Computing Science

Acknowledgement

- Main contents of CSC1104 - Computer Organisation and Architecture are derived from:
  - **Computer organization and architecture, Designing for performance.** Author: William Stallings. Publisher: Pearson.

    Acknowledgement to: Author and Publisher.
  - **Computer organization and Design, The hardware/software interface.** Authors: D. Patterson and J. Hennessy. Publisher: Morgan Kaufmann.

    Acknowledgement to: Authors and Publisher.

Lecture Contents

- Internal Memory:
  - **Semiconductor Main Memory**
  - **Memory Organization**
  - **Error Correction Code (ECC)**
- External Memory:
  - **Solid State Drives (SSD)**
  - **Hard Disk Drive (HDD)**

Internal Memory

University of Glasgow

School of Computing Science

Semiconductor Memory Types

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level		
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		

- Most common memory is random-access memory (RAM):
  - Supporting both read data from and write new data into rapidly.
  - Volatile. Data is lost if no power supply.

Read-Only Memory (ROM) Programmable ROM (PROM)

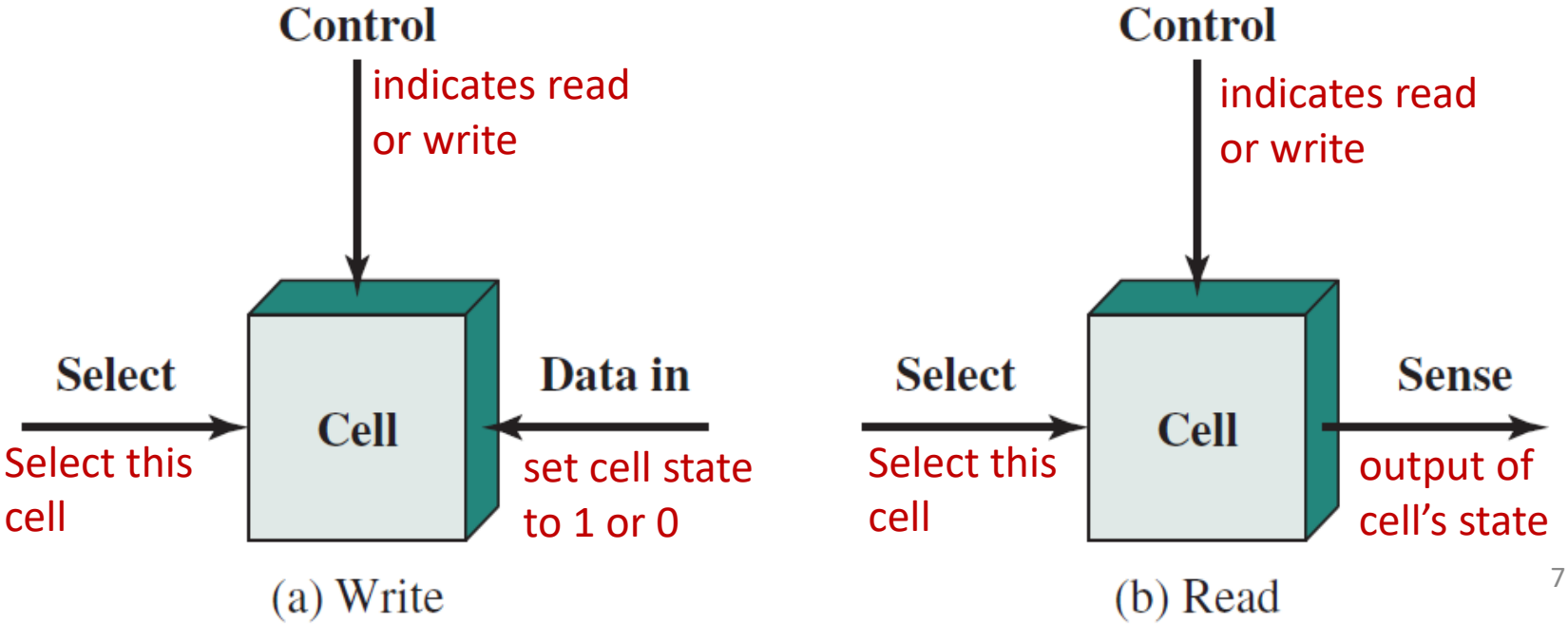
- ROM: Can read data, not change or write new data.
- Non-volatile: No power required to maintain bit values in memory.
- PROM: a less expensive alternative of ROM.
  - Erasable programmable read-only memory (EPROM), Electrically erasable programmable read-only memory (EEPROM), Flash Memory.
- **Flash Memory**: Erase electrically. Can erase blocks of memory, but no byte-level erasure. Achieves high density.

University of Glasgow

School of Computing Science

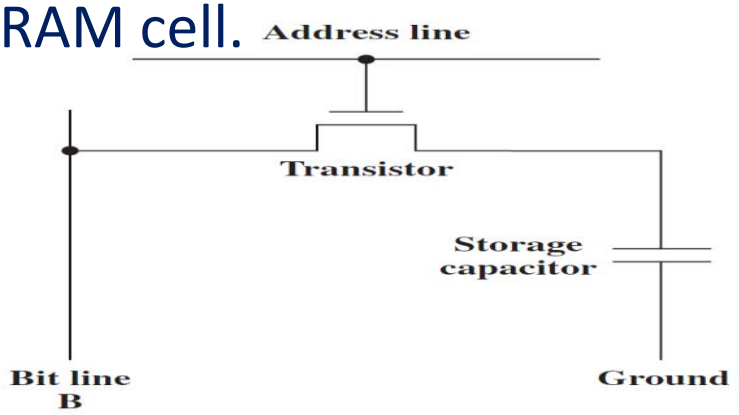
Semiconductor Main Memory

- Common properties of semiconductor memory:
  - **Exhibit two stable (or semi-stable) states, represent binary 1 and 0.**
  - **Capable of being written into, to set the state.**
  - **Capable of being read to sense the cell's state.**
- 3 functional terminals of memory cell (Select, Control, Data in/Sense):

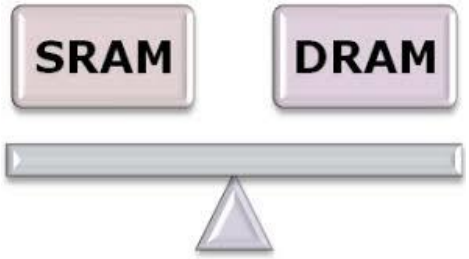


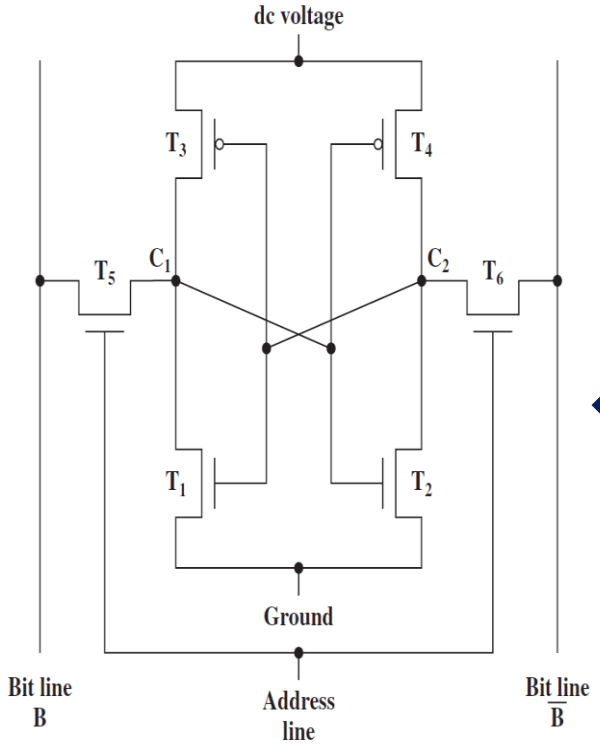
Dynamic RAM (DRAM)

- RAM divided into: *dynamic* (DRAM), *static* (SRAM).
- DRAM is made with cells to store data as charge on capacitors.
- Presence or absence of electric charges in a capacitor is interpreted as: binary '1' or '0'.
- **Refresh operation**: Capacitors have a natural tendency to discharge; DRAM requires periodic charges being ***refreshed*** to maintain data storage.
- That is why it is called ***dynamic***, as opposed to the static storage in a SRAM cell.



SRAM versus DRAM





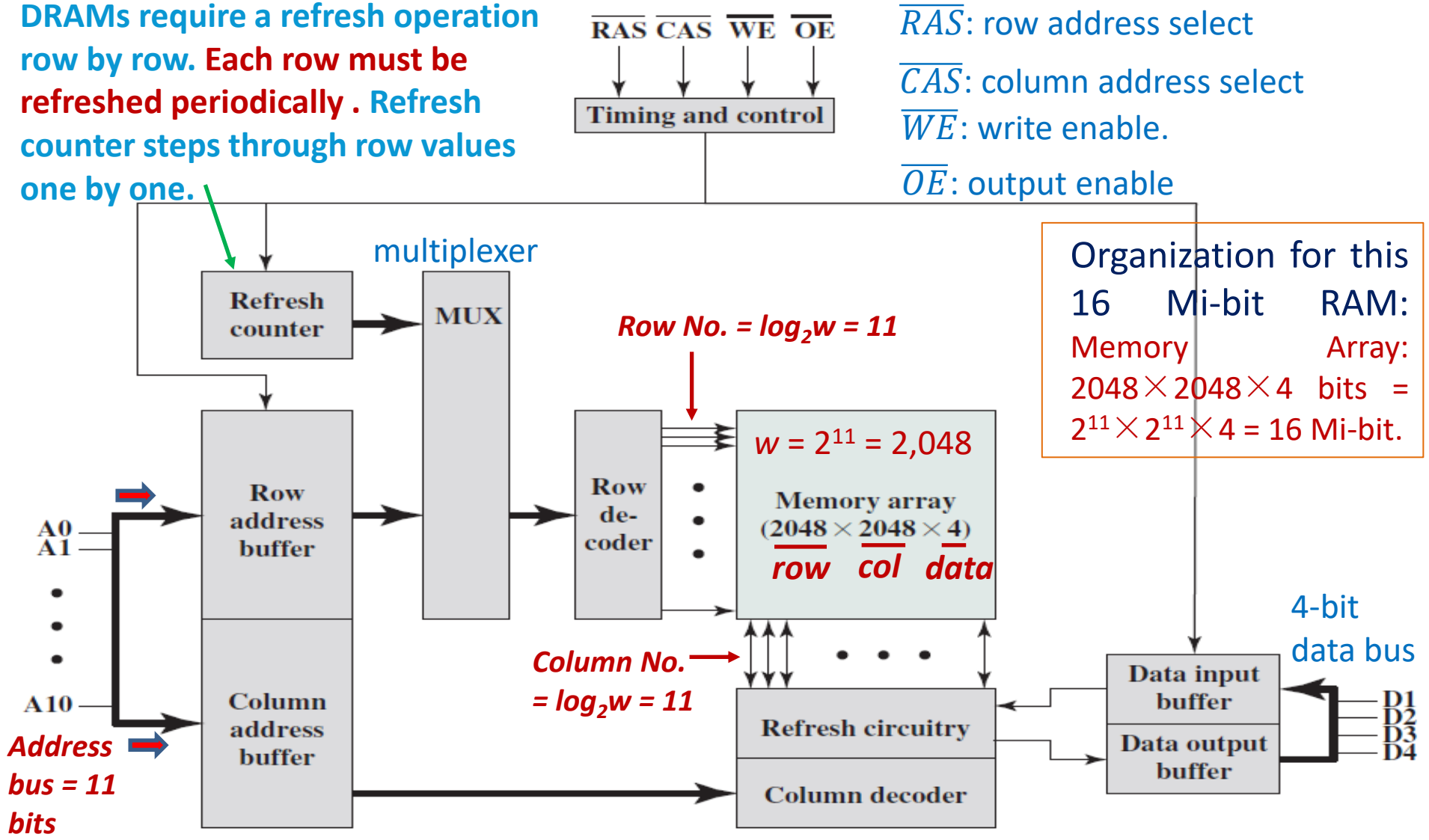
- Both are volatile: Power must be continuously supplied to preserve the bit values.
- DRAM (dynamic cell):
  - Simpler to build, smaller.
  - Denser (smaller cells = more cells per unit area)
  - Less expensive (only 1 transistor)
  - Requires refresh circuitry.
  - Used for main memory.
- SRAM (static cell):
  - Faster.
  - More expensive. (contains 6 transistors)
  - Used for cache memory.

University of Glasgow

School of Computing Science

Typical Organization of A 16-Mibit DRAM

DRAMs require a refresh operation row by row. Each row must be refreshed periodically . Refresh counter steps through row values one by one.



$\overline{RAS}$: row address select

$\overline{CAS}$: column address select

$\overline{WE}$: write enable.

$\overline{OE}$: output enable

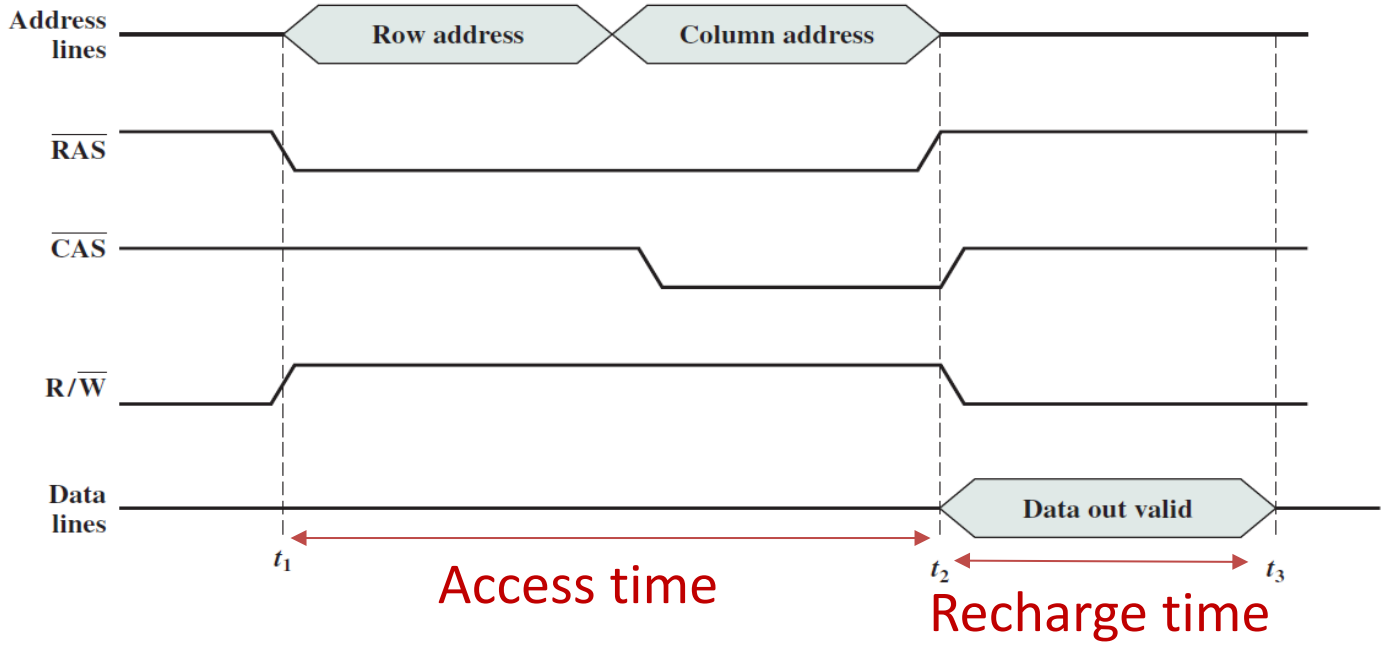
Organization for this 16 Mi-bit RAM:
Memory Array: $2048 \times 2048 \times 4$ bits = $2^{11} \times 2^{11} \times 4$ = 16 Mi-bit.

University of Glasgow

School of Computing Science

DRAM Memory Cycle Time

- **Access time**: time from $\overline{RAS}$ pull low (address is presented to DRAM), to $\overline{CAS}$ pull high (data is available for use): from t_1 to t_2
- **Recharge time**: recharge all DRAM cells before they can be accessed again: from t_2 to t_3
- Memory cycle time = access time + recharge time.



University of Glasgow

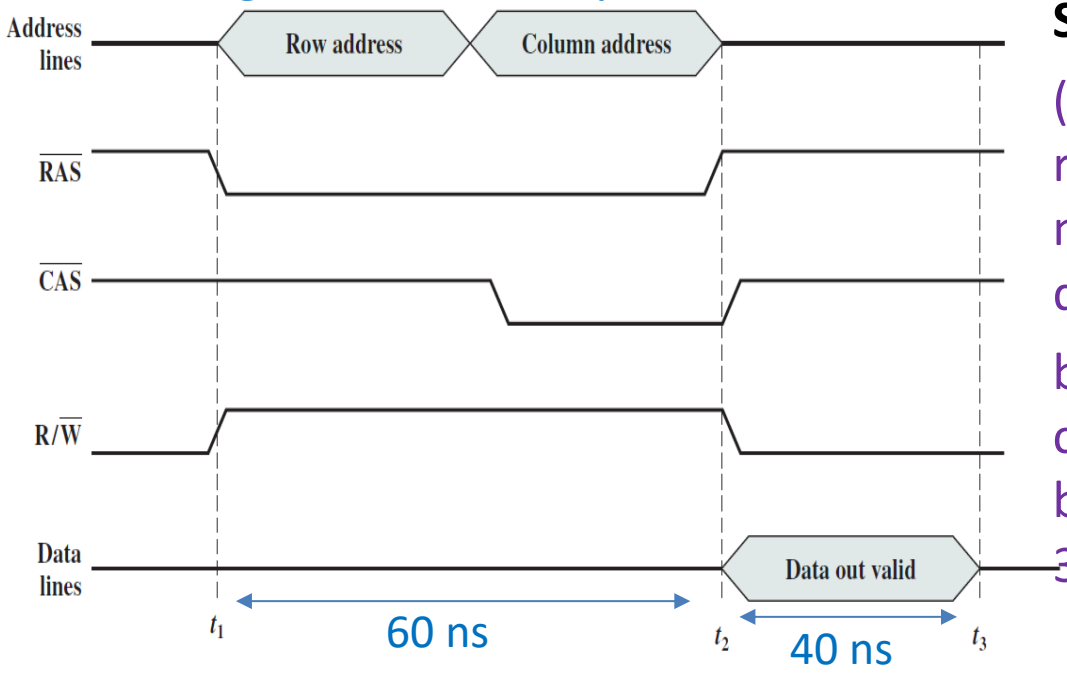
School of Computing Science

Example 3.1 – DRAM Memory Cycle Time

- A DRAM read operation waveform is shown below. The ***access time*** is from t_1 to t_2. The ***recharge time*** is from t_2 to t_3.

  a). What's the memory cycle time? What's the maximum data rate of this DRAM, assuming a 1-bit output?

  b). What is data transfer rate if constructing a 32-bit memory system using these cells in parallel?



Solution:

(a) Memory cycle time = access time + recharge time = 60 ns + 40 ns = 100 ns. 1 bit data needs 100 ns, thus max data rate = $1 / (100 \times 10^{-9})$ = 10 Mb/s.

b) 32-bit data need 32 such DRAM cells connecting in parallel. Thus 32 bits data needs 100 ns. The data rate = $32/(100 \times 10^{-9})$ = 320 Mb/s = 40 MB/s.

University of Glasgow

School of Computing Science

Example 3.2 – DRAM Refresh Time

- A memory is built from 64 Ki×1 DRAM cells. According to data sheet, DRAM cell array is organized into 256 rows. Each row must be refreshed periodically at least once every 4 ms.

  a). What is the time interval of successive refresh requests between every 2 rows?

  b). How long a refresh address counter do we need?

Solution:

a) As there are 256 rows in total, all rows need be refreshed at least once in 4 ms. For row by row refreshing, the time interval of every 2 rows refreshing is:

$$4 \text{ ms} / 256 = 4 * 10^{-3} / 256 = 15.625 * 10^{-6} \text{ second} = 15.625 \ \mu\text{s}.$$

b) The refresh counter need to address each row one by one. For 256 rows, an 8 bits refresh counter is needed to generate 256 unique addresses, as $2^8 = 256$.

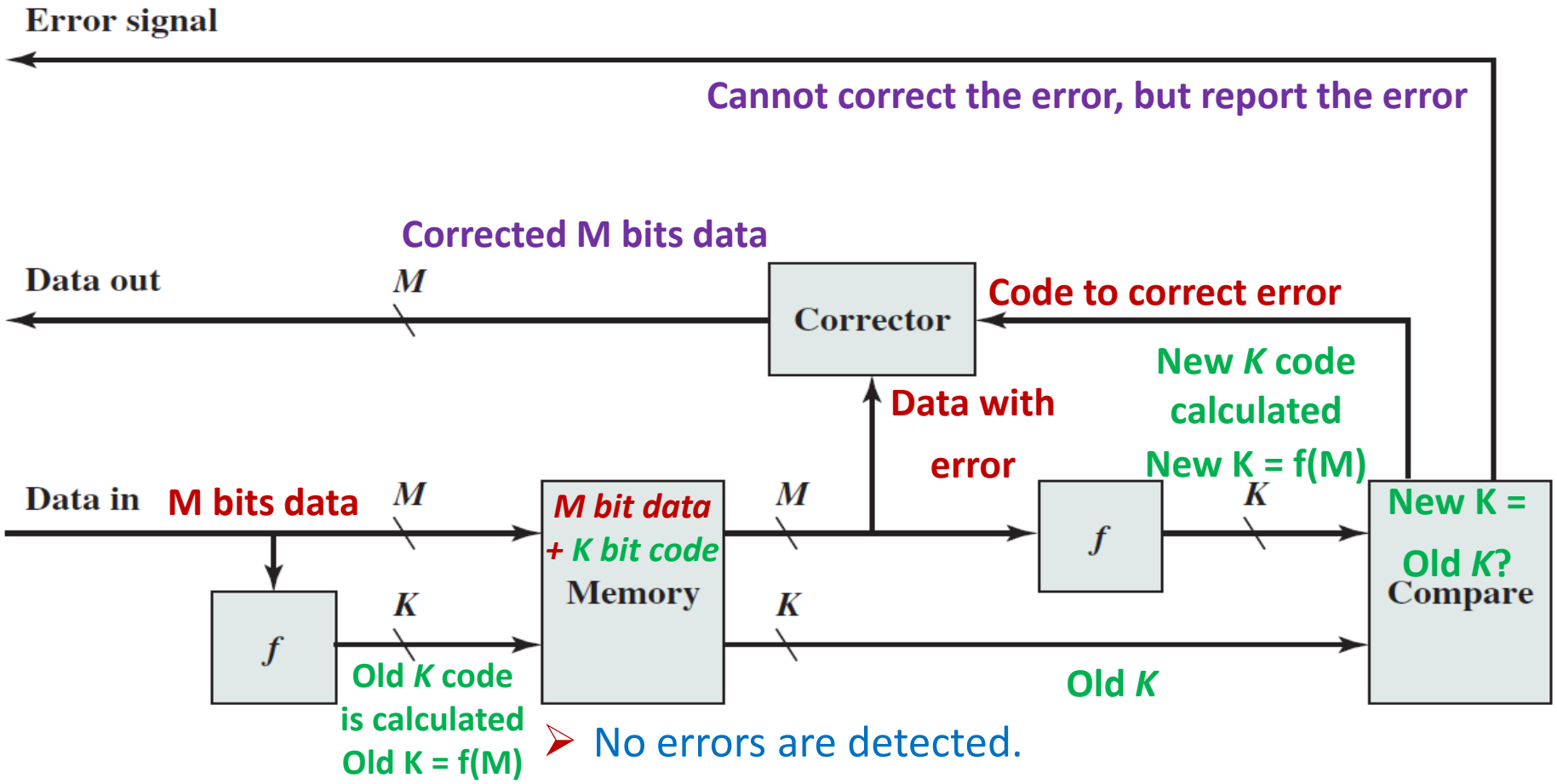
Error Correction

- Memory errors categorized as hard failures & soft errors.
- Hard Failure:
  - Permanent physical defect.
  - Memory cell or cells affected cannot reliably store data, become stuck at 0 or 1 or switch erratically between 0 and 1.
  - Harsh environmental abuse, Manufacturing defects, Wear.
- Soft Error:
  - Random, non-destructive event alters contents of one or more memory cells.
  - No permanent damage to memory.
  - Power supply problems, Alpha particles.

University of Glasgow

School of Computing Science

Error-Correcting Code (ECC) Function



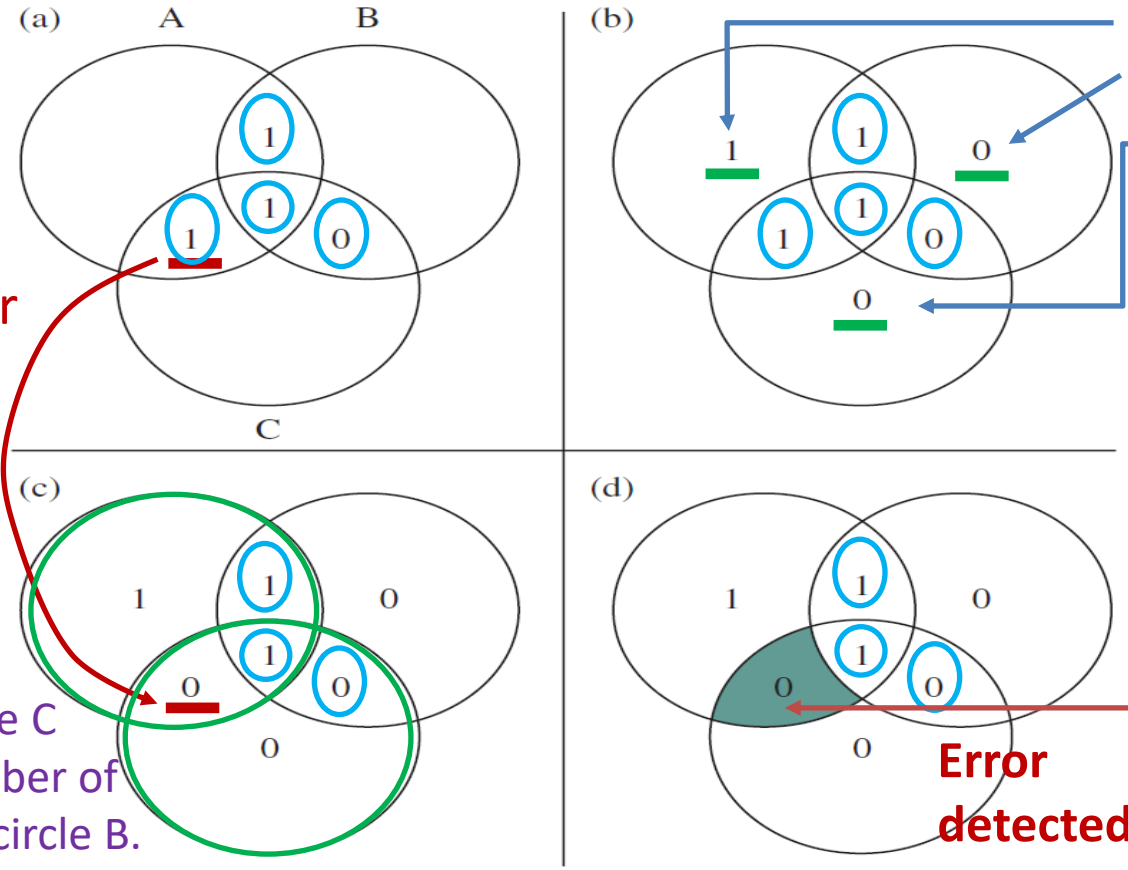
- No errors are detected.
- An error is detected, and possible to correct it.
- An error is detected. but cannot correct it. Report it.

University of Glasgow

School of Computing Science

Hamming Code: Simplest Error-Correcting Code

- Hamming code: for 4-bit words "1110" (M = 4), draw 3 intersecting circles, there are 7 compartments.
- 4 data bits are assigned to the inner compartments.



parity bits: make total number of 1s in any circle is even

If any error changes 1 data bit

circle A & circle C have odd number of 1s. but not in circle B.

Error detected

The error can be corrected by changing this bit.

University of Glasgow

School of Computing Science

Syndrome Word of Hamming Code

- For two *K* parity bits, a bit-by-bit comparison is done by taking ***exclusive-OR*** of these two. Result is called *syndrome word*. (*x* XOR $y = \bar{x} \cdot y + x \cdot \bar{y}$).
  - If a bit of syndrome word = 0, no error.
  - If a bit of syndrome word = 1, with error.
- *K*-bit syndrome word can detect which bit with error in (0, 2^K -1).
- To correct a single bit error in *M* data bits, number of *K* check bits are derived by: $2^k - 1 \geq M + K$.
  - e.g., calculate No. of check bits *K* for an 8 data bits (*M* = 8):
  - If *K* = 3: $2^k - 1 = 7$; *M + K = 11, it means* $2^k - 1 < M + K$. ✗
  - If *K* = 4: $2^k - 1 = 15$; *M + K = 12, it means* $2^k - 1 \geq M + K$. Hence need **K = 4 bits** to check an error in 8-bit data. ✓

University of Glasgow

School of Computing Science

Example 3.3 – Length of Hamming Code Check Bit

- How many check bits are needed if Hamming error correction code is used to detect single bit errors in a 512-bit and a 1024-bit data word? What are the size overhead caused by the check bits?

Solution:

- For 512-bit data ($2^9 = 512$, and $M = 512$),

  if $K = 9$, $2^k - 1 = 511$; $M + K = 512 + 9 = 521$, means $2^k - 1 < M + K$. ✗

  if $K = 10$, $2^k - 1 = 1023$; $M + K = 512 + 10 = 522$, means $2^k - 1 \geq M + K$. ✓

  Hence, $K = 10$, need 10 check bits to check errors in 512-bit data.

- For 1024-bit data ($2^{10} = 1024$, and $M = 1024$),

  if $K = 10$, $2^k - 1 = 1023$; $M + K = 1024 + 10 = 1034$, it means $2^k - 1 < M + K$. ✗

  if $K = 11$, $2^K - 1 = 2047$; $M + K = 1024 + 11 = 1035$, it means $2^k - 1 \geq M + K$. ✓

  Hence, $K = 11$, need 11 check bits to check errors in 1024-bit data.

- For 512-bit data, overhead is (512 + 10) / 512 – 100% = 1.95%.
- For 1024-bit data, overhead is (1024 + 11) / 1024 – 100% = 1.07%.

University of Glasgow

School of Computing Science

Layout of Data Bits and Check Bits

Bit position	12	11	10	9	2^3 8	7	6	5	2^2 4	3	2^1 2	2^0 1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5		D4	D3	D2		D1		
Check bit					C8				C4		C2	C1

- Check bits are at position No. as powers of 2: 2^0 (C1), 2^1 (C2), 2^2 (C4), 2^3 (C8).
- Each check bit covers on data bits whose position number contains a 1 in the same bit position as check bit.
  - C1 checks data with a 1 at the least significant bit: 0011 (3), 0101 (5), 0111 (7), 1001 (9), 1011 (11). Thus, C1 = D1⊕D2⊕D4⊕D5⊕D7.
  - C2 checks data with a 1 at the 2nd least significant bit: 0011 (3), 0110 (6), 0111 (7), 1010 (10), 1011 (11). Thus, C2 = D1⊕D3⊕D4⊕D6⊕D7.
  - C4 checks data with a 1 at the 2nd most significant bit: 0101 (5), 0110 (6), 0111 (7), 1100 (12). Thus, C4 = D2⊕D3⊕D4⊕D8.
  - C8 checks data with a 1 at the most significant bit: 1001 (9), 1010 (10), 1011 (11), 1100 (12). Thus, C8 = D5⊕D6⊕D7⊕D8.

University of Glasgow

School of Computing Science

Example 3.4 – Hamming Code Check for 8 Bit Data

- An 8-bit data is "00111001", with data bit D1 in the rightmost position. Suppose now that data bit D3 sustains an error and is changed from 0 to 1. Illustrate Hamming code able to find such error.

Solution:

- Calculate check bits based on correct data "00111001."
- C1 = D1⊕D2⊕D4⊕D5⊕D7 = 1⊕0⊕1⊕1⊕0 = 1.
- C2 = D1⊕D3⊕D4⊕D6⊕D7 = 1⊕0⊕1⊕1⊕0 = 1.
- C4 = D2⊕D3⊕D4⊕D8 = 0⊕0⊕1⊕0 = 1.
- C8 = D5⊕D6⊕D7⊕D8 = 1⊕1⊕0⊕0 = 0.
- The data received with error is "00111101". Calculate the check bits are:
- C'1 = D'1⊕D'2⊕D'4⊕D'5⊕D'7 = 1⊕0⊕1⊕1⊕0 = 1.
- C'2 = D'1⊕D'3⊕D'4⊕D'6⊕D'7 = 1⊕1⊕1⊕1⊕0 = 0.
- C'4 = D'2⊕D'3⊕D'4⊕D'8 = 0⊕1⊕1⊕0 = 0.
- C'8 = D'5⊕D'6⊕D'7⊕D'8 = 1⊕1⊕0⊕0 = 0.
- C"8 = C8⊕C'8 = 0. C"4 = C4⊕C'4 = 1. C"2 = C2⊕C'2 = 1. C"1 = C1⊕C'1 = 0.
- Thus the *syndrome word* indicates the position number as "0110" in error, meaning D3.

University of Glasgow

School of Computing Science

Illustration for Single-Error-Correcting (SEC) Code

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit	D8	D7	D6	D5	2^3	D4	D3	D2	2^2	D1	2^1	2^0
Check bit					C8				C4		C2	C1
Word stored as	0	0	1	1	0	1	0	0	1	1	1	1
Word fetched as	0	0	1	1	0	1	1	0	1	1	1	1
Position Number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Check Bit					0				0		0	1

Error

- C"8 = C8⊕C'8 = 0. C"4 = C4⊕C'4 = 1. C"2 = C2⊕C'2 = 1. C"1 = C1⊕C'1 = 0.
- Thus, the *syndrome word* indicates the position number as 0110 in error, meaning D3.

Single-Error-Correcting, Double-Error-Detecting (SEC-DED) Code

- A single-error-correcting (SEC) code can correct a single error.
- More commonly, semiconductor memory is equipped with a single-error-correcting, double-error-detecting (SEC-DED) code.
- SEC-DED codes require **an additional bit** compared with SEC codes.

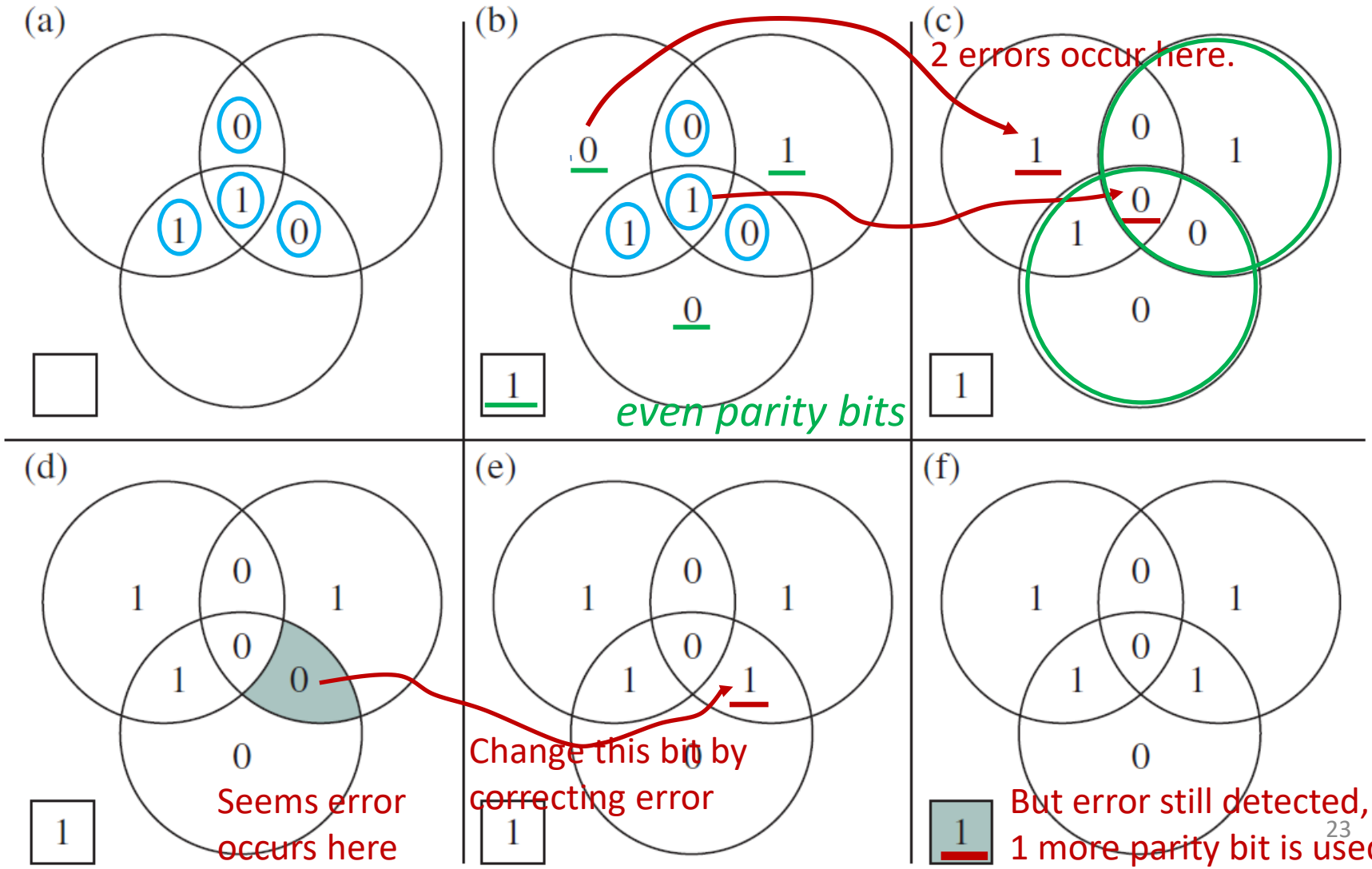
	Single-Error Correction		Single-Error Correction/ Double-Error Detection	
Data Bits	Check Bits	% Increase	Check Bits	% Increase
8	4	50.0	5	62.5
16	5	31.25	6	37.5
32	6	18.75	7	21.875
64	7	10.94	8	12.5
128	8	6.25	9	7.03
256	9	3.52	10	3.91

University of Glasgow

School of Computing Science

Hamming SEC-DED Code

- 4 data bits "0110" (*M*=4) are assigned to the inner compartments



SEC-DED Code

- An error-correcting code enhances the reliability of the memory at the cost of added complexity.
- With a 1-bit-per-chip organization, an SEC-DED code is generally considered adequate.
- e.g., using an 8-bit SEC-DED code for each 64 bits of data in main memory. Thus, the size of main memory becomes about 12% larger (overhead): *(64+8) / 64 * 100% - 100% = 12.5%.*
- Using a 7-bit SEC-DED for each 32 bits of memory, for a 22% overhead: *(32+7) / 32 * 100% - 100%= 21.9%.*

University of Glasgow

School of Computing Science

Synchronous DRAM (SDRAM)

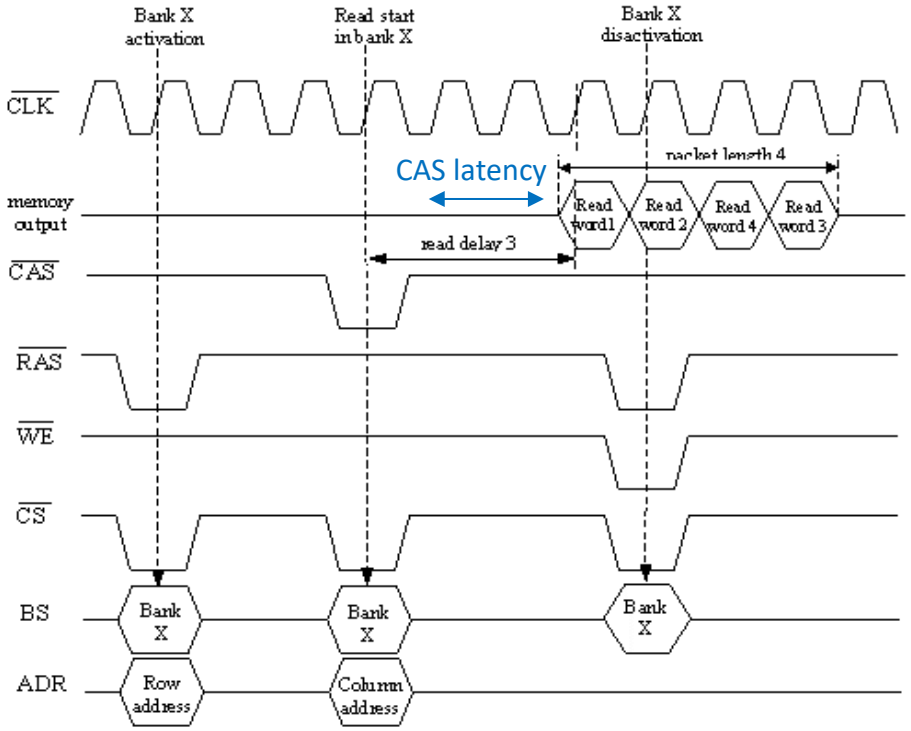
- SDRAM and DDR-SDRAM: currently dominate market.
- Unlike asynchronous DRAM, SDRAM exchanges data with CPU synchronized to a clock signal.
- Run at full speed of data bus without wait states.
- With synchronous access, SDRAM moves data in and out under control of system clock.
- CPU gives instruction and address, which is latched by SDRAM. SDRAM then responds after a set number of clock cycles (latency).

University of Glasgow

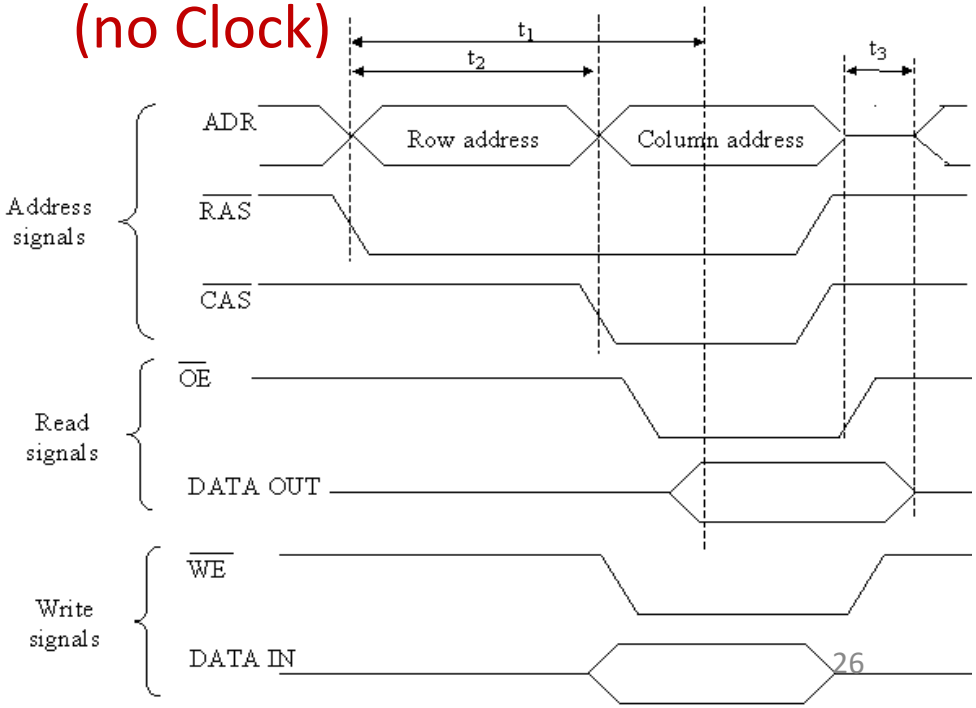
School of Computing Science

Example Waveforms of Synchronous DRAM (SDRAM)

Synchronous DRAM Waveform (with Clock)



Asynchronous DRAM Waveform (no Clock)

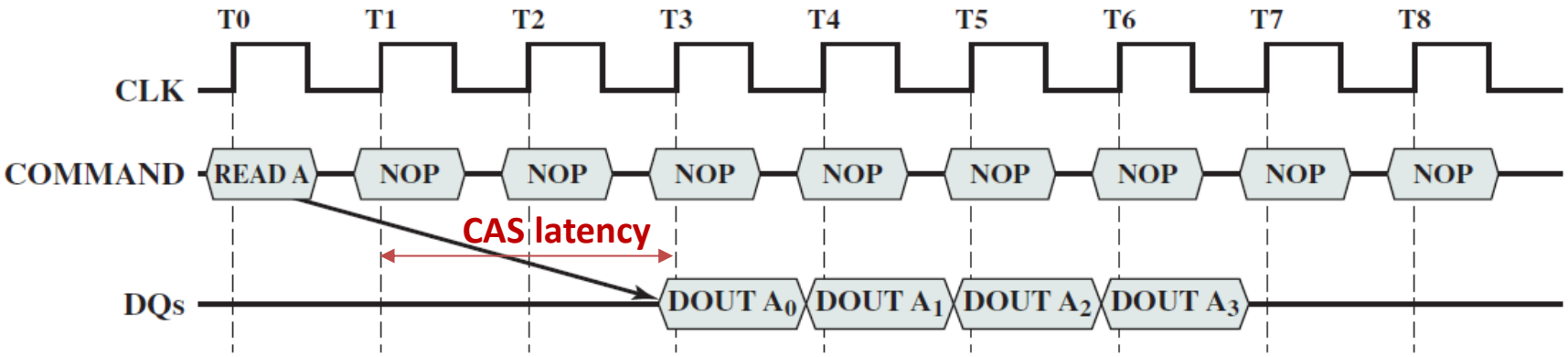


University of Glasgow

School of Computing Science

Example Waveform of SDRAM

- SDRAM uses a burst mode to eliminate address setup time; row and column line pre-charge time after the first access.
- In burst mode, a series of data can be clocked out rapidly once the 1st data has been accessed. (Burst length: 1, 2, 4, 8, full page.)
- SDRAM performs best when transferring large blocks of data sequentially, such as video and audio files.



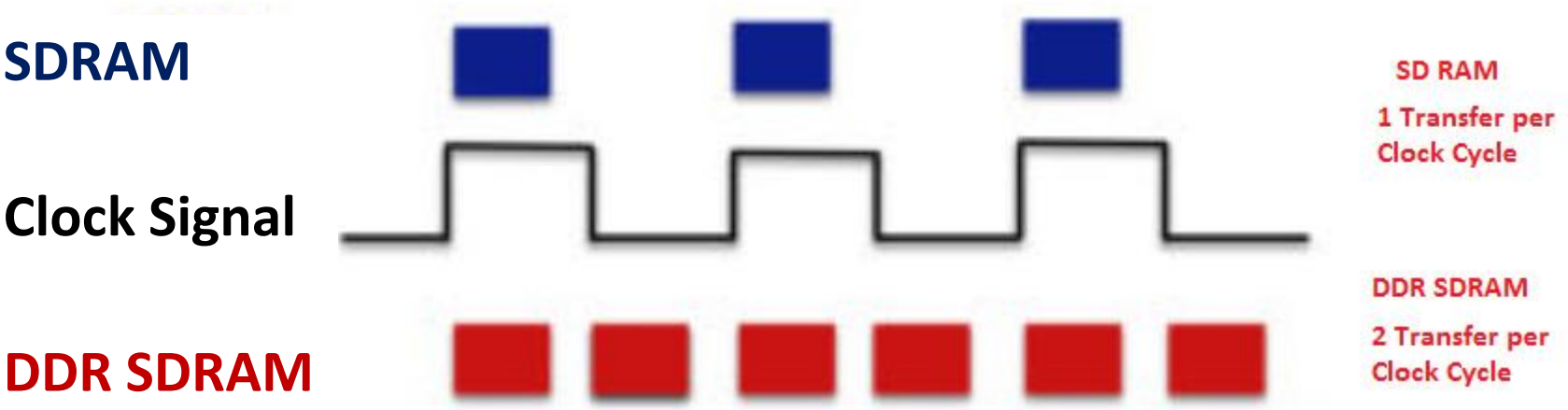
SDRAM Read Timing (burst length = 4, $\overline{\text{CAS}}$ latency = 2)

University of Glasgow

School of Computing Science

Double Data-Rate SDRAM (DDR DRAM)

- DDR DRAM: data transfer is synchronized to both rising and falling edge of the clock, rather than just the rising edge. This doubles the data rate.



External Memory

University of Glasgow

School of Computing Science

Solid State Drives (SSD)

- SSD is a memory device made by semiconductors electronic circuitry, replacement to a hard disk drive.
- SSDs v.s. HDDs:
  - **High-performance input/output operations per second**.
  - **Durability:** Less susceptible to physical shock and vibration.
  - **Longer lifespan**: SSDs are not susceptible to mechanical wear.
  - **Lower power consumption:** use considerably less power.
  - **Quieter and cooler:** Less dimension, lower energy costs.
  - **Lower access times and latency rates:** Over 10 times faster than spinning disks in an HDD.
  - **But HDD is cheaper per GB, and larger storage capacity.**

Practical Issues of SSD

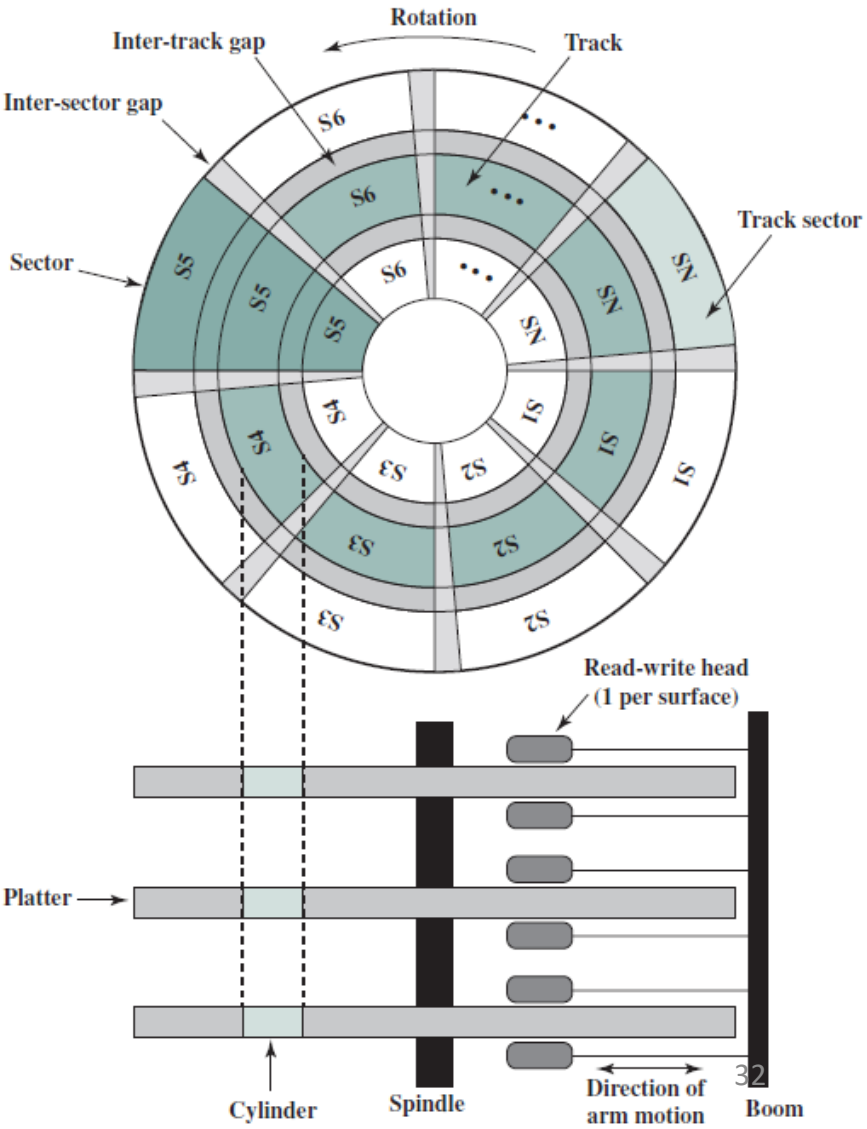
1) Performance has a tendency to slow down along usages.
   - Flash memory accessed in blocks, a typical block size 512 KB.
   - Become fragmented over time, pages scattered over multiple blocks.
   - The more space occupied, the more fragmentation. Writing of a new file into multiple blocks becomes slower.
2) SSD becomes unusable after a typical number of 100,000 writes as lifetime of flash cell.
   - Most flash devices can estimate their own remaining lifetimes, can anticipate failure and take preemptive action.

University of Glasgow

School of Computing Science

Magnetic Disk

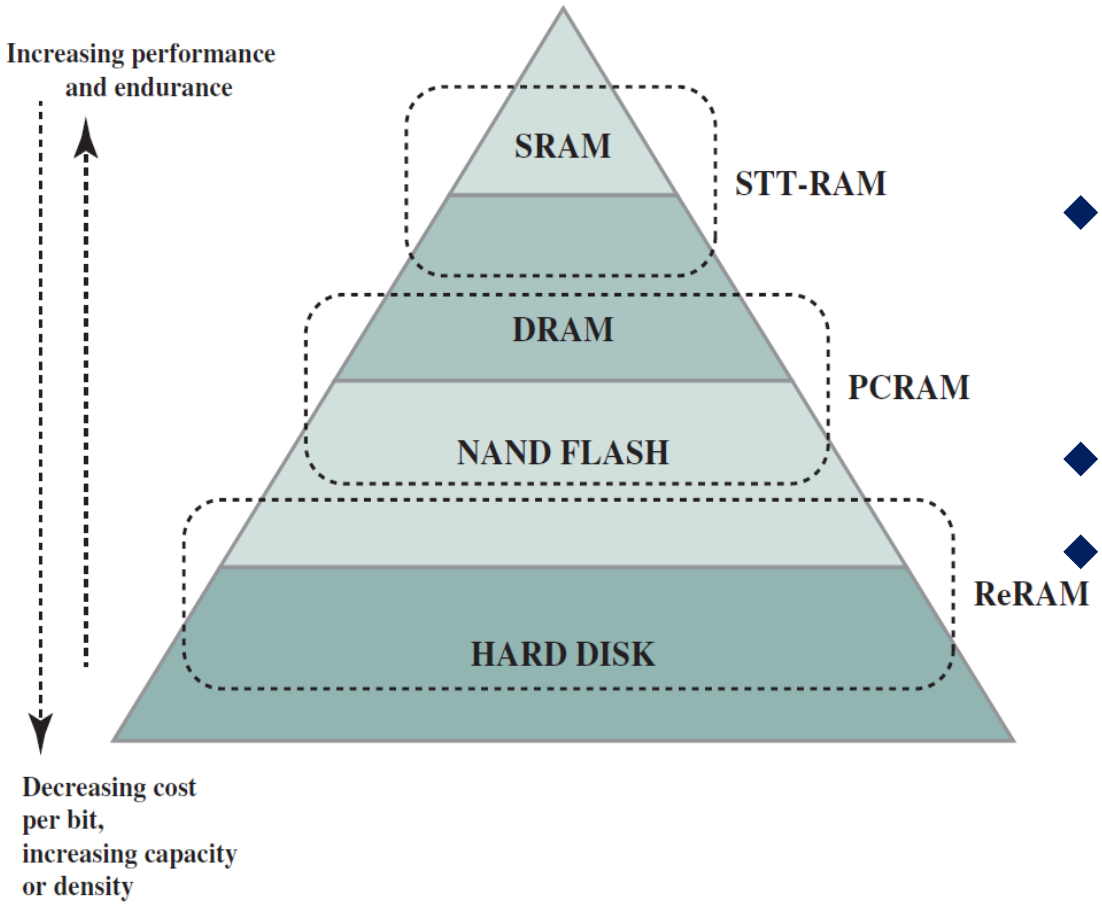
- A disk is a circular platter constructed of nonmagnetic substrate, coated with a magnetizable material.
- Organization of data on platter surface is in a concentric set of rings, called ***tracks***.
- Thousands of **tracks** per surface.
- Hundreds of **sectors** per **track**, with size of 512 bytes per sector.
- Adjacent tracks separated by intertrack gaps. Adjacent sectors separated by intersector gaps.



University of Glasgow

School of Computing Science

Nonvolatile RAM within the Memory Hierarchy



- SRAM: Rapid access time, but the most expensive and least bit density. Suitable to cache memory.
- DRAM: Cheaper, denser, and slower than SRAM, Suitable to off-chip main memory.
- Flash Memory and SSD.
- Hard disk: very high bit density, very low cost per bit, with relatively slow access times. Suitable to external storage.

Next Lecture

Lecture 4: Input/Output and Operating System