Main Memory

COE 501

Computer Architecture

Prof. Muhamed Mudawar

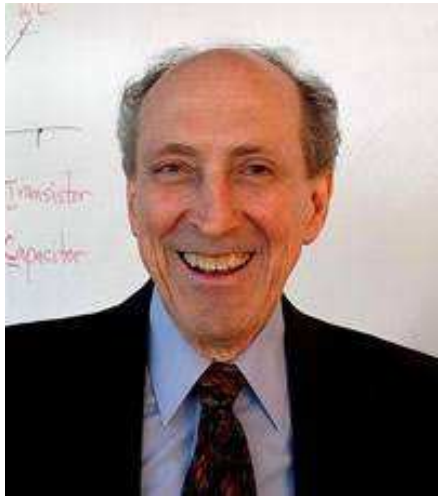
Computer Engineering Department

King Fahd University of Petroleum and Minerals

SRAM and DRAM

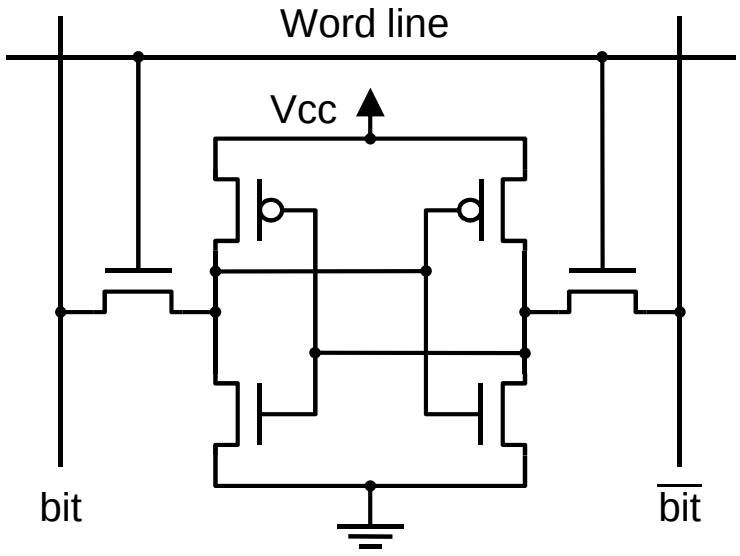
Two semiconductor memory technologies since the 1970's

- SRAM = Static RAM
  - Invented by John Schmidt at Fairchild Semiconductor in 1964
  - 6-Transistor Cell in CMOS technology (low power to retain bit)
  - Faster than DRAM and easier to use ☺
  - SRAM is used for: cache memory, register files, tables, and buffers
- DRAM = Dynamic RAM
  - Invented by Robert Dennard in 1968
  - One Transistor + Capacitor per bit
  - More dense and cheaper than SRAM ☺
  - Must be refreshed periodically ☹
  - Typical choice for main memory

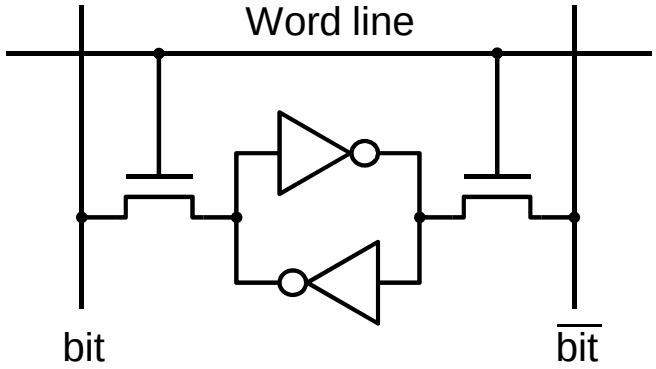


Static RAM Storage Cell

- 6-Transistor (6T) cell in CMOS
- Cell Implementation:
  - Cross-coupled inverters to store bit (4T)
  - Two access transistors (2T)
  - Does not require refreshing to maintain bit
  - Word line enables access to the cell
  - Two bit lines: bit and its complement $\overline{bit}$
  - Smaller and denser than a flip-flop
- Provides fast access time
- SRAM operation: standby, read, write

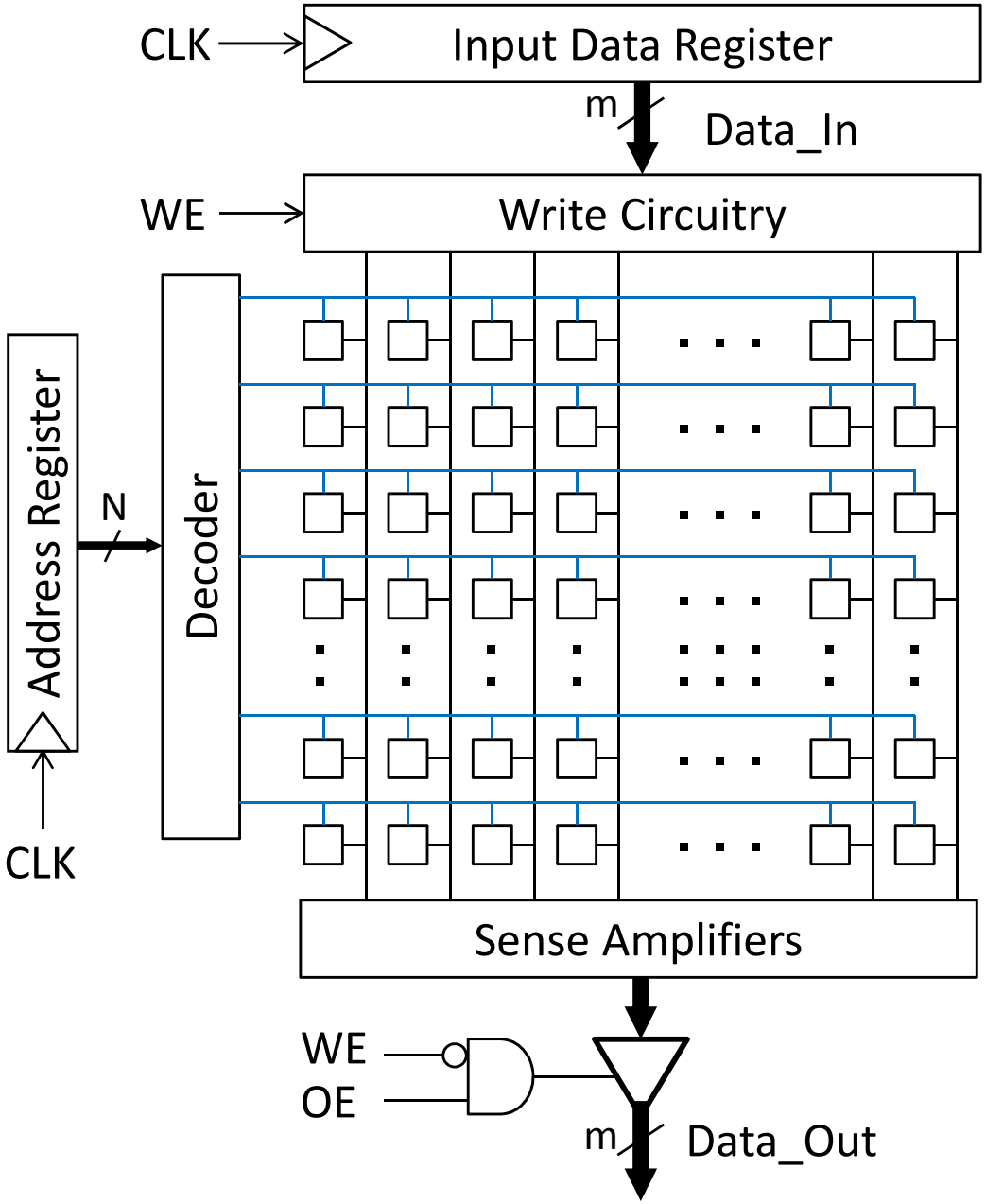


Typical SRAM cell



Memory Array Architecture

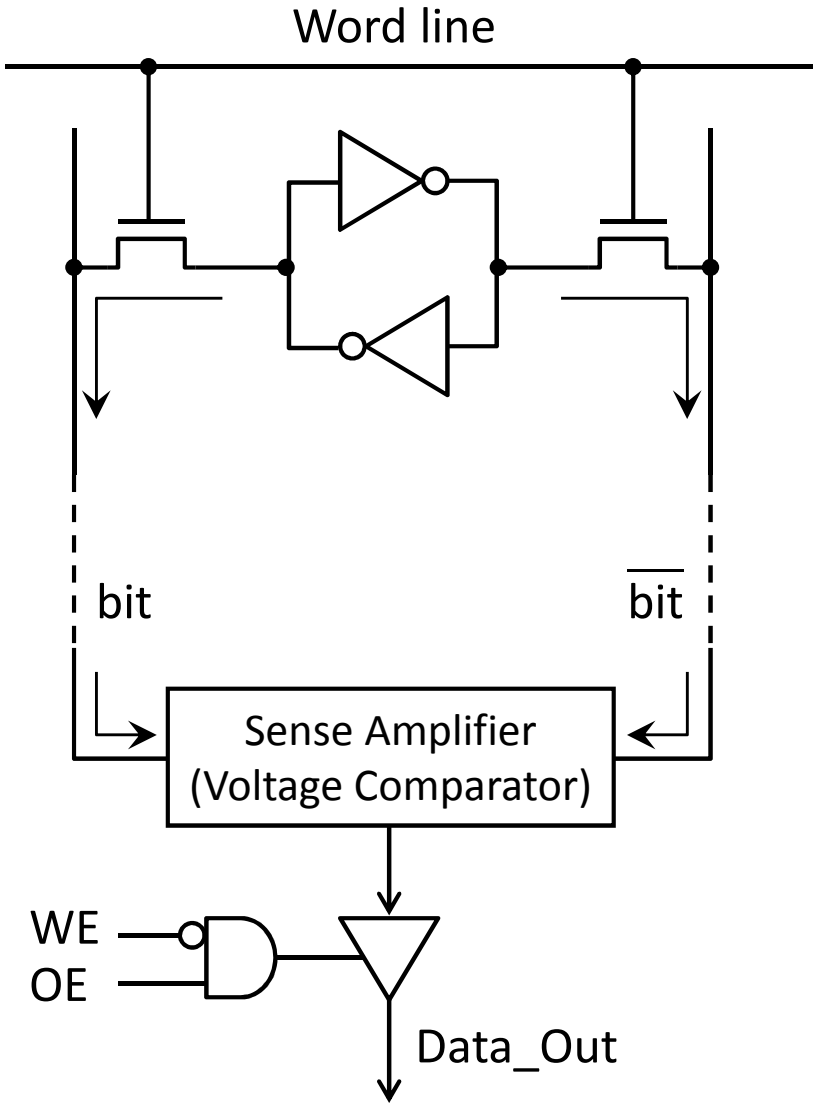
- 2D Memory Array
  - Good regularity, high density
- n-bit address ➔ 2^n rows
- Each row has m cells
- Capacity = $2^n \times m$ bits
- The row size (m) can be large
  - Cache: m can be 512 bits
  - Register File: m can be 64 bits
- Control signals:
  - WE = Write Enable
  - OE = Output Enable
  - CLK = Clock Signal



SRAM Read Operation

For a read operation:

- The address is decoded
- One row is accessed
- Each row consists of m cells
- The accessed cells drive the bit lines and change their voltages
- Each pair of bit lines is sensed by an amplifier that compares the voltage values and outputs the logic state
- A tri-state buffer puts the output data on the data bus. Output must be enabled (OE) and no write (~WE)

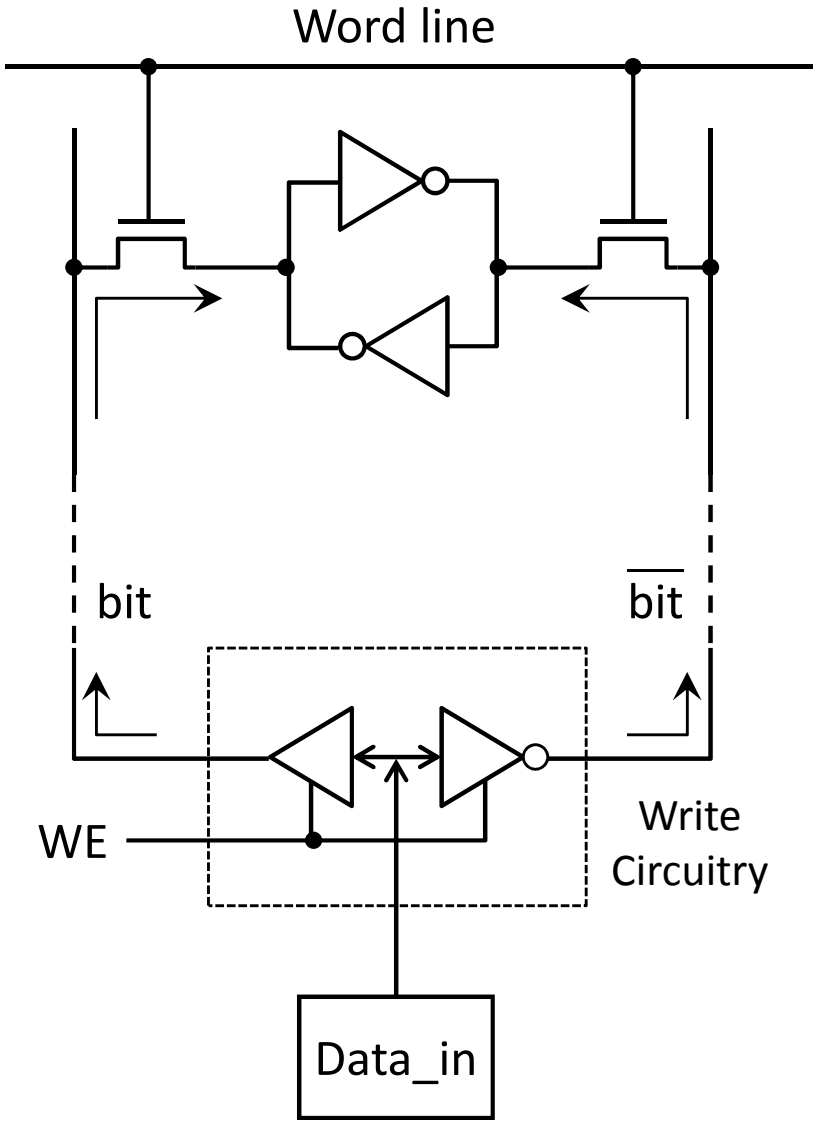


Read Operation

SRAM Write Operation

For a write operation:

- The data comes from an input register
- The data moves to the write circuitry
- Write buffers drive the bit lines
- The row address is decoded
- One row is accessed (m cells)
- The write circuitry drivers are stronger than the cell transistors
- The selected cells are forced to change their state value



Write Operation

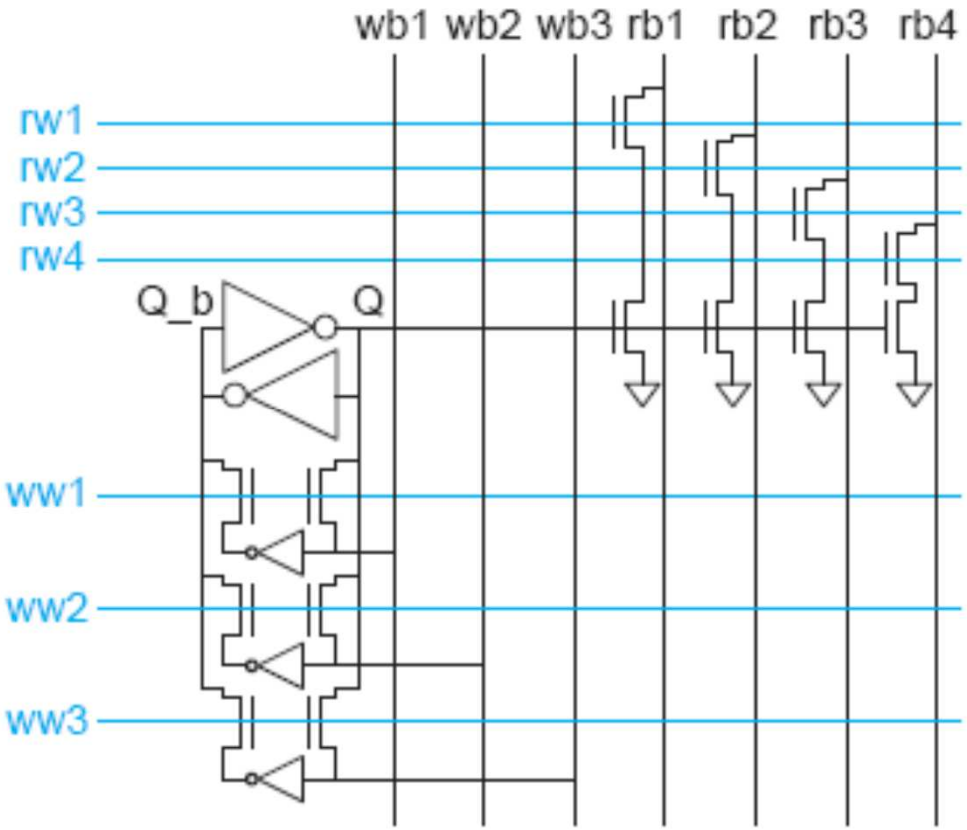
Multiple Ports

- So far we considered single-ported SRAM
  - One read or one write on each clock cycle, but not both
- **Multi-ported** SRAM are needed for register files
- **Time-multiplexing** of read and write operations:
  - Write operation during first half of a clock cycle, if latency < half cycle
  - Read operation during second half a clock cycle (read after write)
- Time-multiplexing is useful, but not sufficient for register files
  - A simple pipelined processor typically requires two read (source) register ports and one write (destination) port each clock cycle.
  - A superscalar processor requires multiple read and write ports. It can read more than two registers and write more than one each clock cycle.

Multi-Ported SRAM Cell

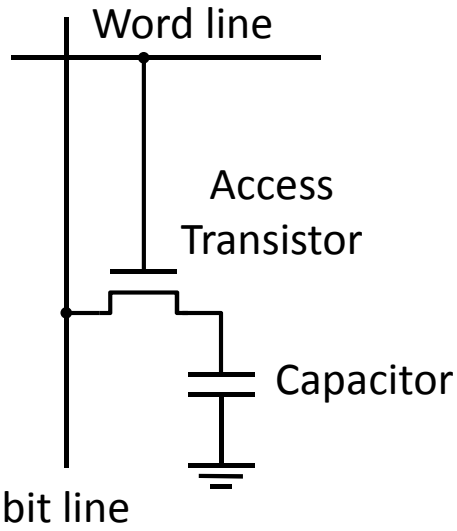
- SRAM cell design must be changed to allow multiple ports
- Additional word lines, bit lines, and access transistors per cell
- For N ports, cell area = O(N^2)
- Area is dominated by wires
- Single bit lines save area
- Custom design for processors

Example:
4 Read ports + 3 Write ports
7 Decoders (1 per port)
7 word lines per cell
7 bit lines per cell (Q only)



Dynamic RAM

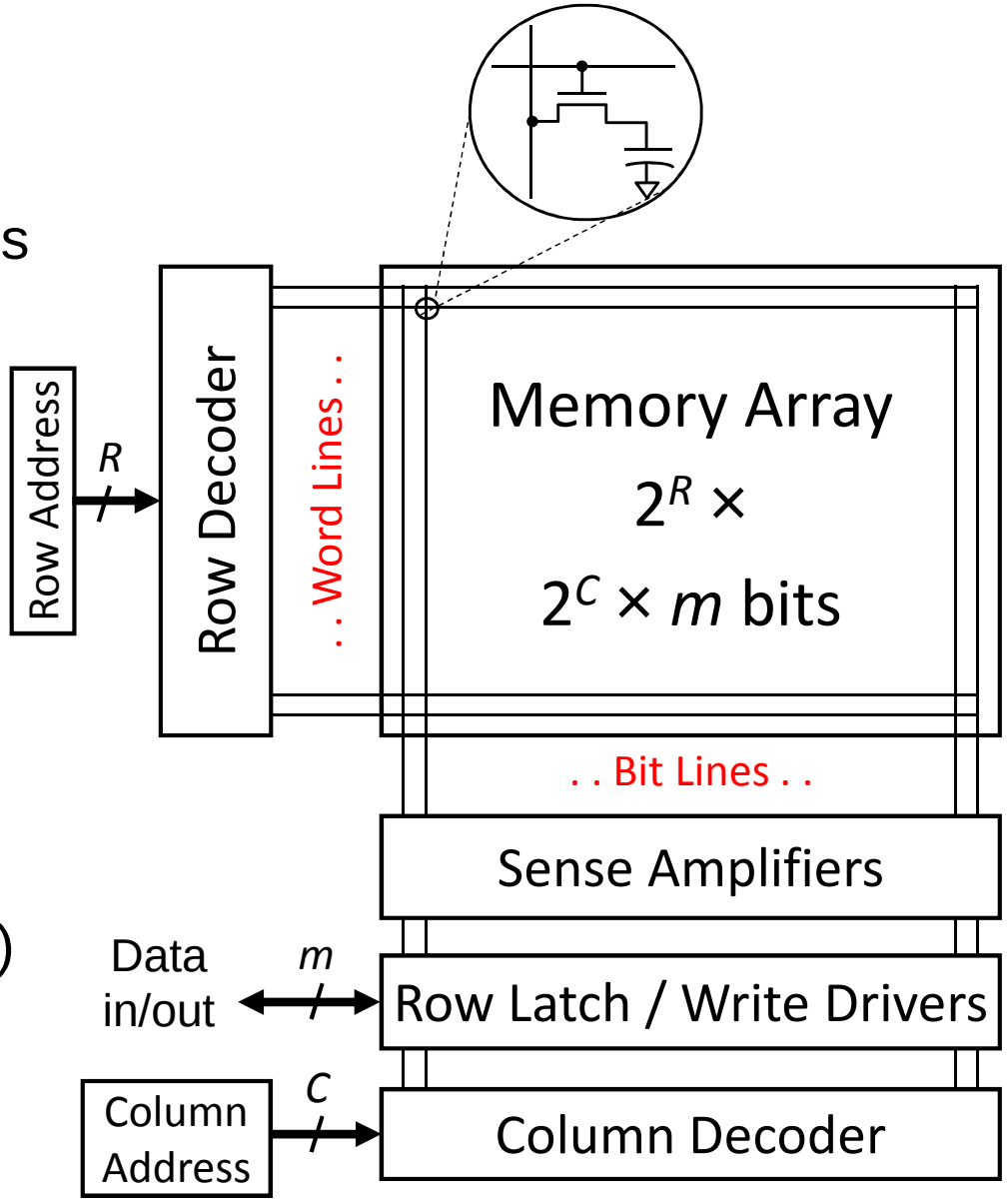
- Slower, Cheaper, and Denser memory than SRAM
- Typical choice for **main memory**
- Cell Implementation:
  - 1-Transistor (1T) Cell (access transistor)
  - Trench capacitor (stores bit)
  - Cell area is 10X to 20X smaller than SRAM
- Bit is stored as a **charge** on capacitor
- Must be **refreshed periodically**
  - Because of leakage of charge from tiny capacitor
- Refreshing for a memory row
  - Reading each row and writing it back to restore the charge



DRAM cell

DRAM Memory Structure

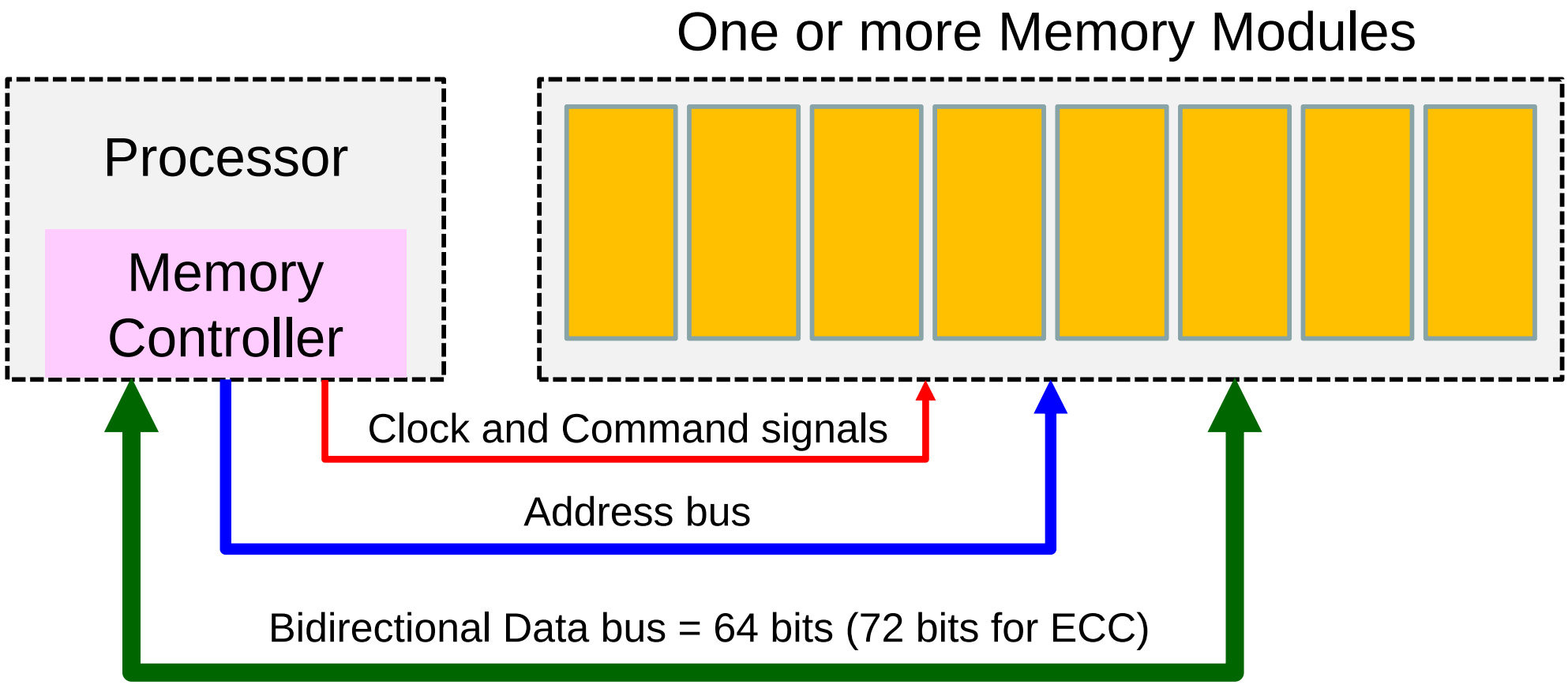
- 2D Memory Array of DRAM cells
  - *R* bits for the row address ➔ 2^R rows
  - *C* bits for the column address ➔ 2^C cols
  - Data bus width = *m* bits
- Capacity = $2^R \times 2^C \times m$ bits
- Row decoder: select row to open
  - Sense Amplifiers read data on bit lines
  - A complete row is read and latched
- Column decoder
  - Select column to read/write (within row)
  - Bidirectional (in/out) data bus = *m* bits
- Write Drivers
  - Write row back into the memory array



SDRAM and DDR SDRAM

- Old DRAMs were asynchronous (no clock input)
- SDRAM is **Synchronous Dynamic RAM**
  - Added clock input to DRAM chip interface
- SDRAM is synchronous with the memory bus clock
  - As memory bus clock speed improved, SDRAM delivered higher performance than asynchronous DRAM
- DDR is **Double Data Rate** SDRAM
  - Like SDRAM, DDR is synchronous with the bus clock, but DDR transfers data on both the rising and falling edges of the clock

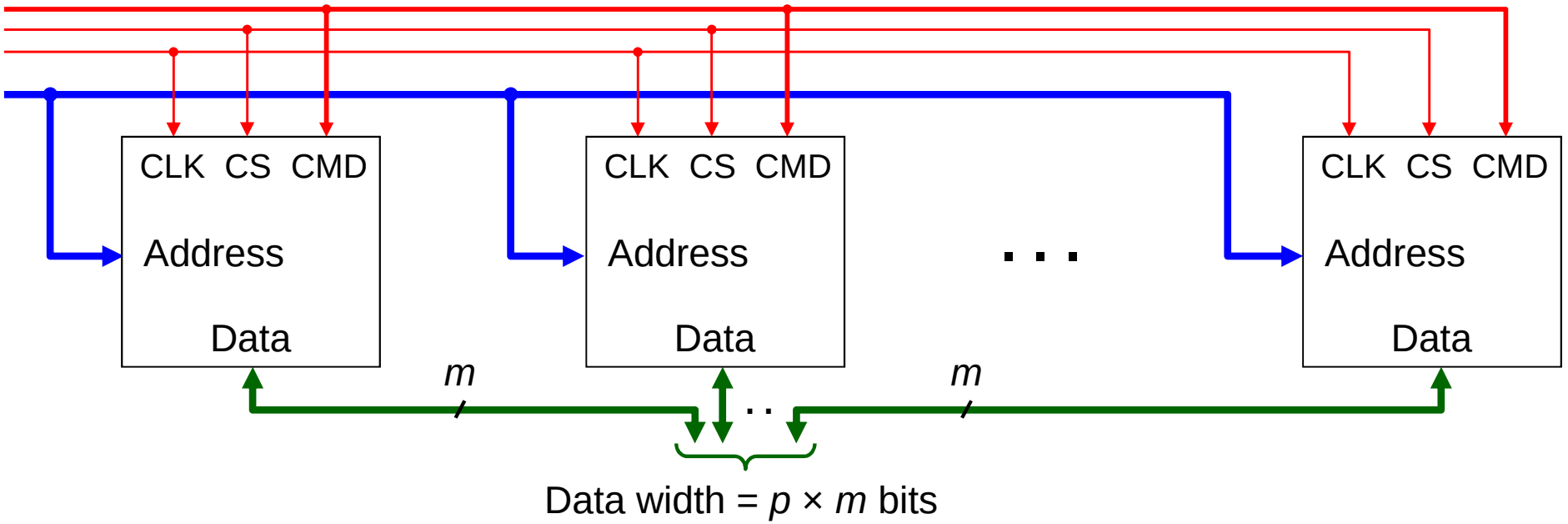
Main Memory Subsystem



- Memory controller is integrated on the processor chip
- Sends commands and memory addresses to the memory modules
- Sends and receives data on the bidirectional data bus

Memory Module

- Group of DRAM chips accessed in parallel on a small PCB
  - Same clock and command signals
  - Same address, but separate data lines
  - Increases memory capacity and bandwidth
  - Example: **Four x16** DRAM chips (p = 4, m = 16 bits)

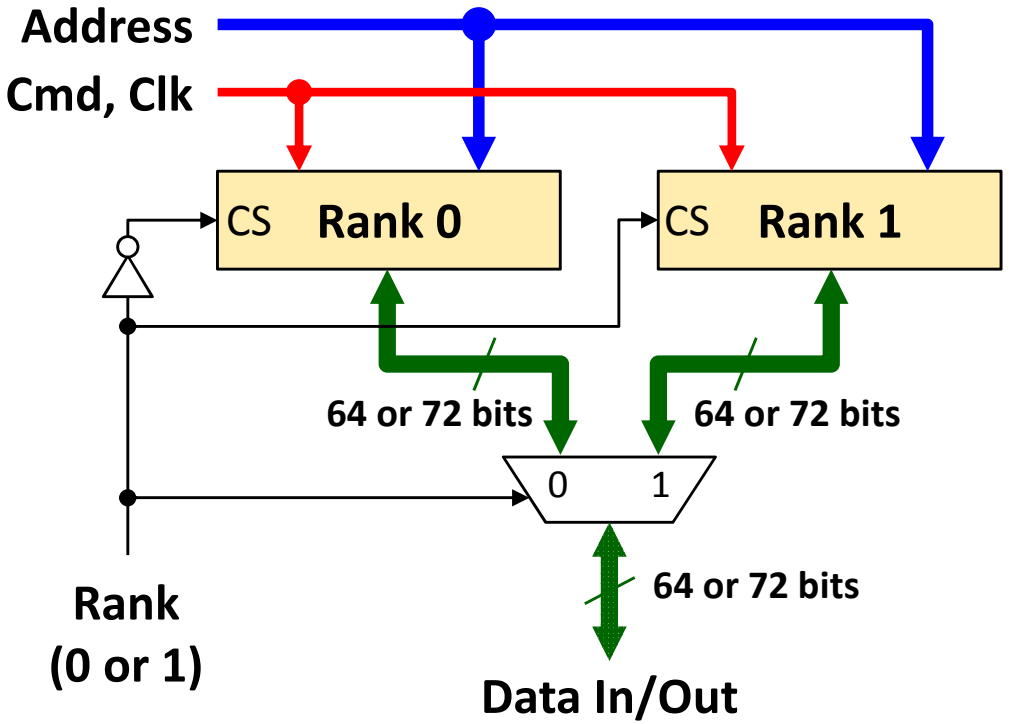


Module Characteristics

- Different forms: DIMM versus SO-DIMM (Small Outline)
- Different module interfaces for DDR, DDR2, 3, and 4
  - NOT backward compatible: different pins, signaling, and timing
- Chip density and data width
  - Size of each DRAM chip and width of data in bits: x4, x8, or x16
- ECC (Error Correcting Code) versus non-ECC module
  - Non-ECC module uses 64-bit data bus, while ECC uses 72-bit data bus
- Number of DRAM chips on a Module:
  - Four x16 chips, Eight x8 chips, or Sixteen x4 chips ➔ 64-bit data bus
  - Nine x8 chips or Eighteen x4 chips ➔ 72-bit data bus (ECC module)
- Registered (RDIMM) versus Unregistered (UDIMM):
  - Registered RDIMM add a register between the DRAMs and controller

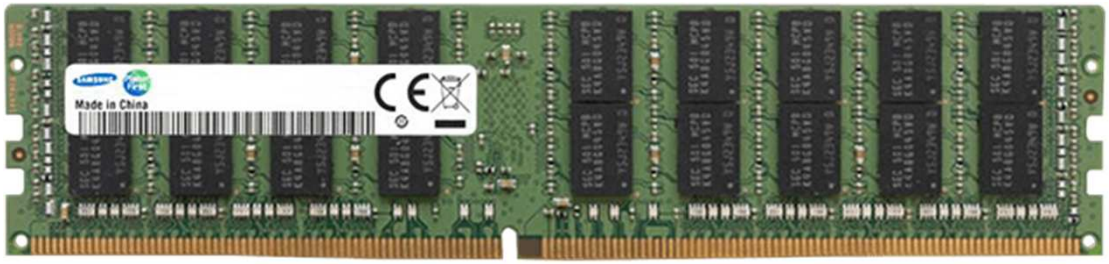
Multiple Memory Ranks

- A memory rank is a set of DRAM chips on a memory module
  - Connected to the **same chip select** and accessed simultaneously
  - Driven by the same command and address
- A memory module can have 1, 2, 4, or 8 ranks
  - Multiple ranks share the same 64-bit data bus (ECC adds 8 bits)



Given a Dual-Ranked Module

- If Rank input is 0, select Rank-0 chips
- If Rank input is 1, select Rank-1 chips

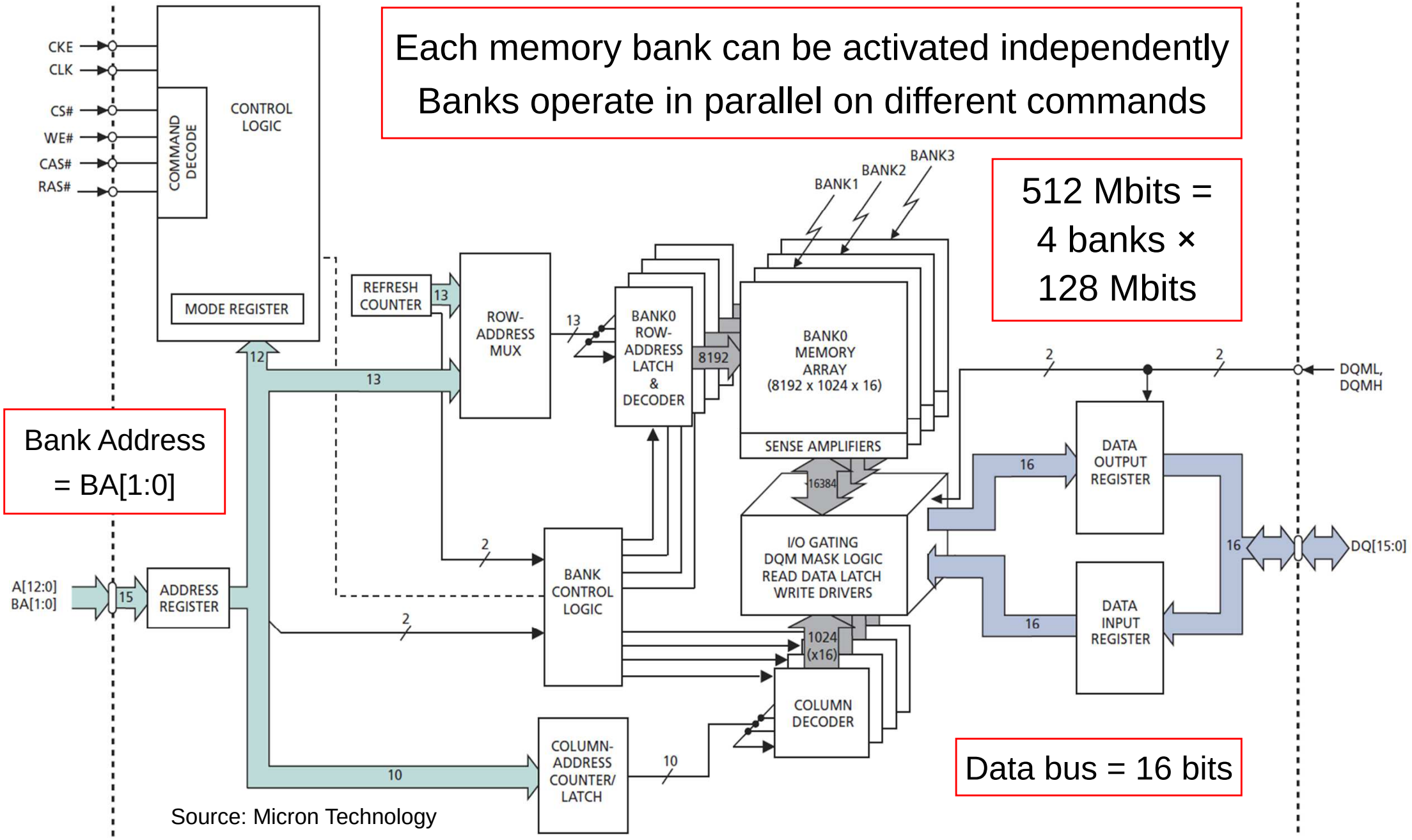


Memory Module Cost and Capacity

- Example:
  - Suppose we use 16-Gbit DRAMs each costing $10, to build DIMMs
  - We have 3 choices for the data width: x4, x8, or x16 (16-bit data per chip)
  1. Which DRAM chip would you choose to build the lowest cost single-ranked non-ECC module, and what will be its capacity?
  2. Which DRAM chip would you choose to built the highest capacity quad-ranked module with ECC, and what will be its cost?

- **Solution:**
  1. Choose **x16** DRAMs. Only **Four** x16 DRAMs are needed for 64-bit data. Cost = 4×$10 = **$40** (ignoring cost of board). Capacity = **8 GBytes**
  2. Choose **x4** DRAMs. **Eighteen** x4 DRAMs are needed for 72-bit data bus (single rank with ECC). For a quad-ranked module, we need 4 × 18 = 72 DRAM chips. Cost = 72×$10 = **$720**. Capacity = 64×16 Gb = **128 GBytes**

Inside an SDRAM Chip with Four Banks



Source: Micron Technology

Memory Banks

- When a memory array becomes large …
  - The wordlines and bitlines become long
  - Long lines have high capacitance, power consumption, and long delay
- Solution is to partition a large array into subarrays, called **banks**
- Memory banks can be accessed in parallel
- Memory address = 

  - Memory addresses are mapped to different banks at the **row level**
- Each bank must have the following:
  - A subarray of memory cells, row address, and decoder
  - Sense amplifiers and a row latch for storing an open row
  - A column decoder for read/write operations
  - Write drivers for writing the open row back and closing it

SDRAM Chip Interface

CLK: Memory Bus Clock input

CKE: Clock Enable

BA0 – BA1: Bank Address

A0 – A12: Address lines

DQ0 – DQ15: Data in/out

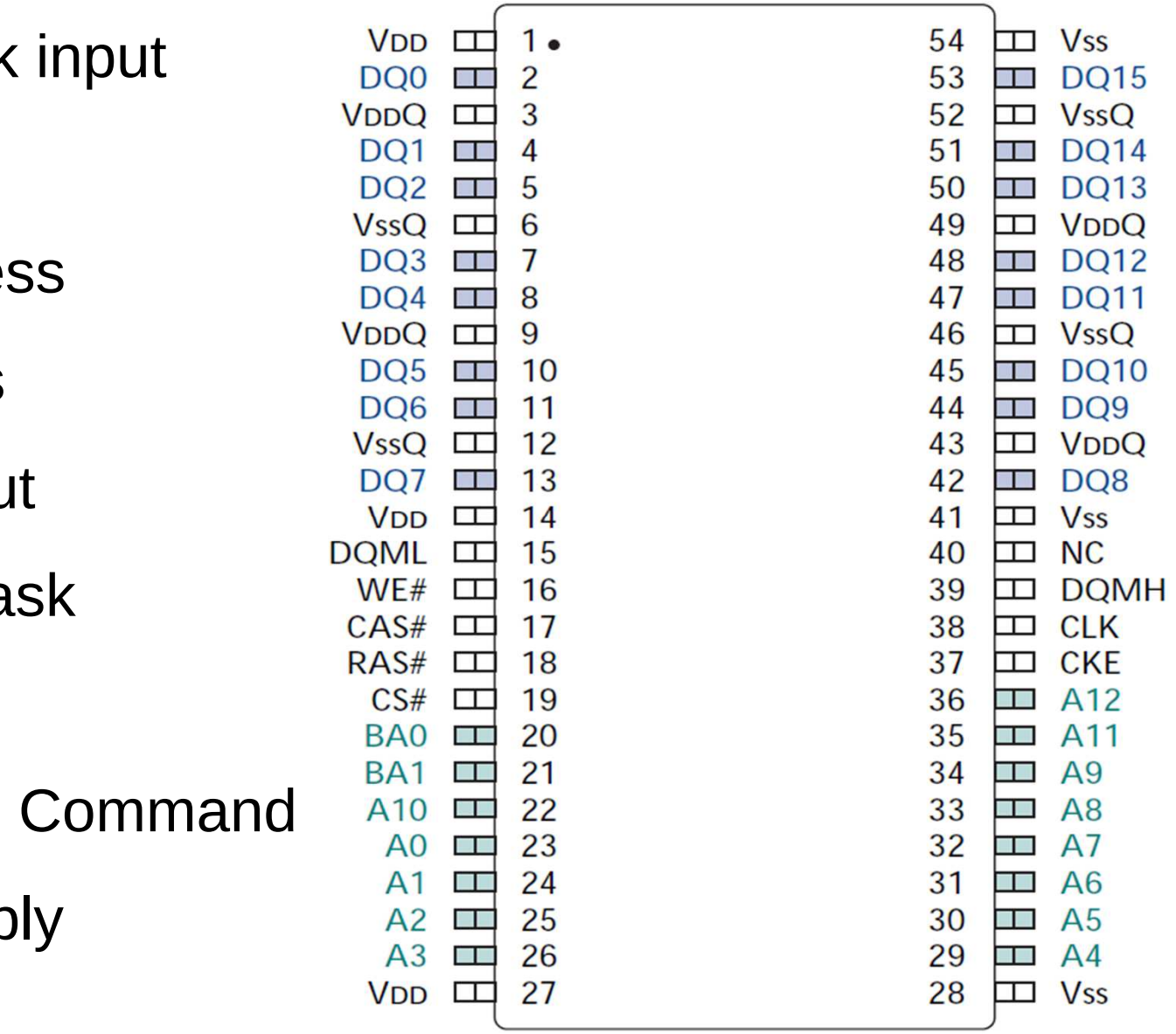
DQMH, DQML: Data Mask

CS: Chip Select

RAS, CAS, WE: Control Command

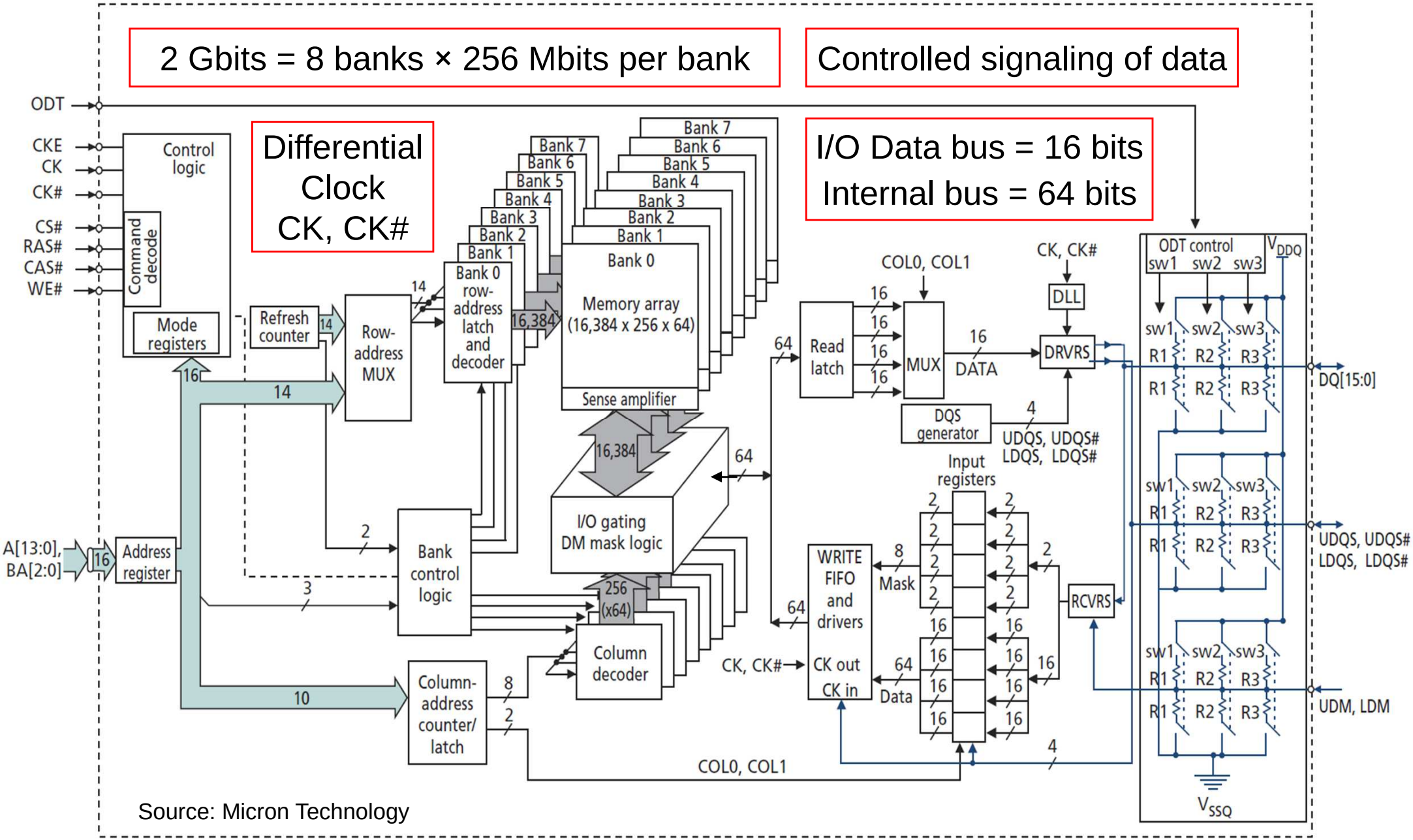
VDD, VDDQ: Power supply

VSS, VSSQ: Ground



Source: Micron Technology

DDR2 SDRAM Chip (Eight Banks)



Source: Micron Technology

Main Memory Commands

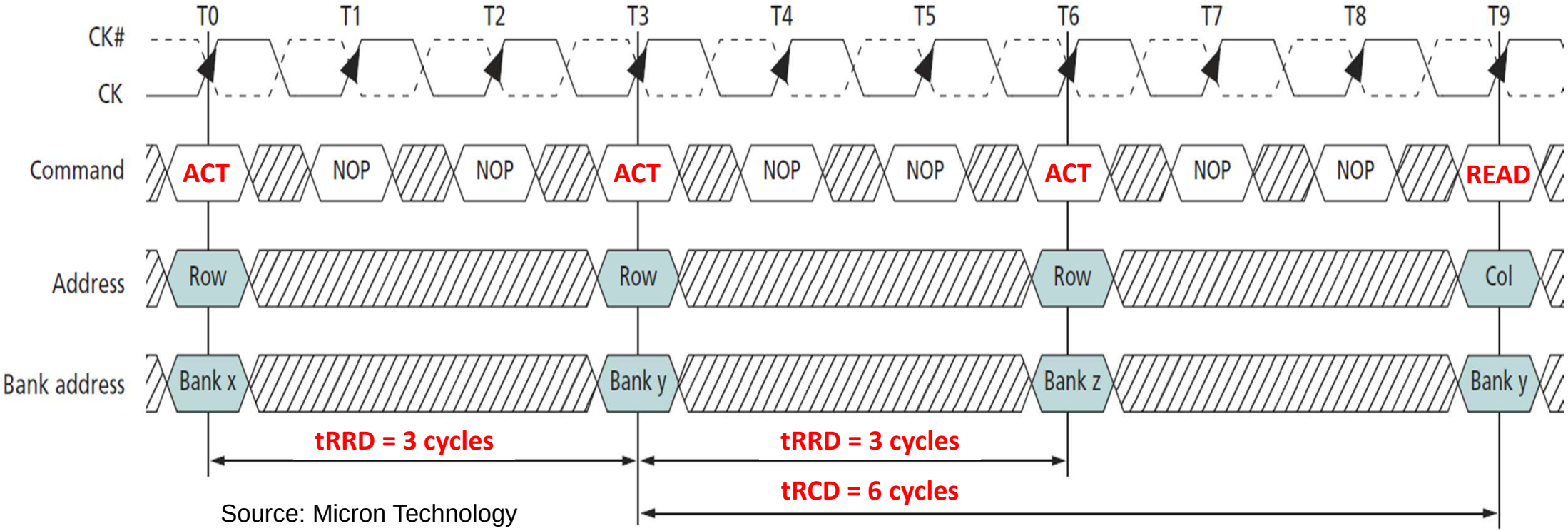
- **NOP:** No Operation
- **Activate:** Open a new row in a bank for Read/Write commands
- **Read:** Read data from a currently open row of a selected bank
- **Write:** Write data to a currently open row of a selected bank
- **Precharge:** Close an open row of selected bank and write it back
- **Precharge All:** Close the open rows of all banks and write back
- **Read with Precharge:** Read data from an open row then close it
- **Write with Precharge:** Write data to an open row then close it
- **Refresh:** Refresh one row using an internal counter
- **Self Refresh:** Retain data in memory without external clocking

ACTIVATE Command

- **Activate:** Open a Row in a Bank for Read or Write commands
  - The bit lines are pre-charged to half voltage (between logic 0 and 1)
  - The bit lines act like capacitors that store a charge momentarily
  - Bank address specifies the memory bank
  - Row address is latched and decoded in the selected bank
  - One row is accessed in the selected bank
  - Bit lines share charge with storage cell ➔ Destructive read
  - Sense amplifiers detect a small change in voltage on the bit lines
  - Sense amplifiers output and save the row bits in a **Row Latch**
- Row remains open until a **Precharge** command

Timing of ACTIVATE Command

- Row-to-Column Delay: **tRCD**
  - Minimum cycles between ACTIVATE and READ/WRITE commands
- Row-to-Row Delay: **tRRD**
  - Minimum cycles between two ACTIVATE commands to different banks
  - Access to multiple banks can be overlapped, which hides part of **tRCD** delay



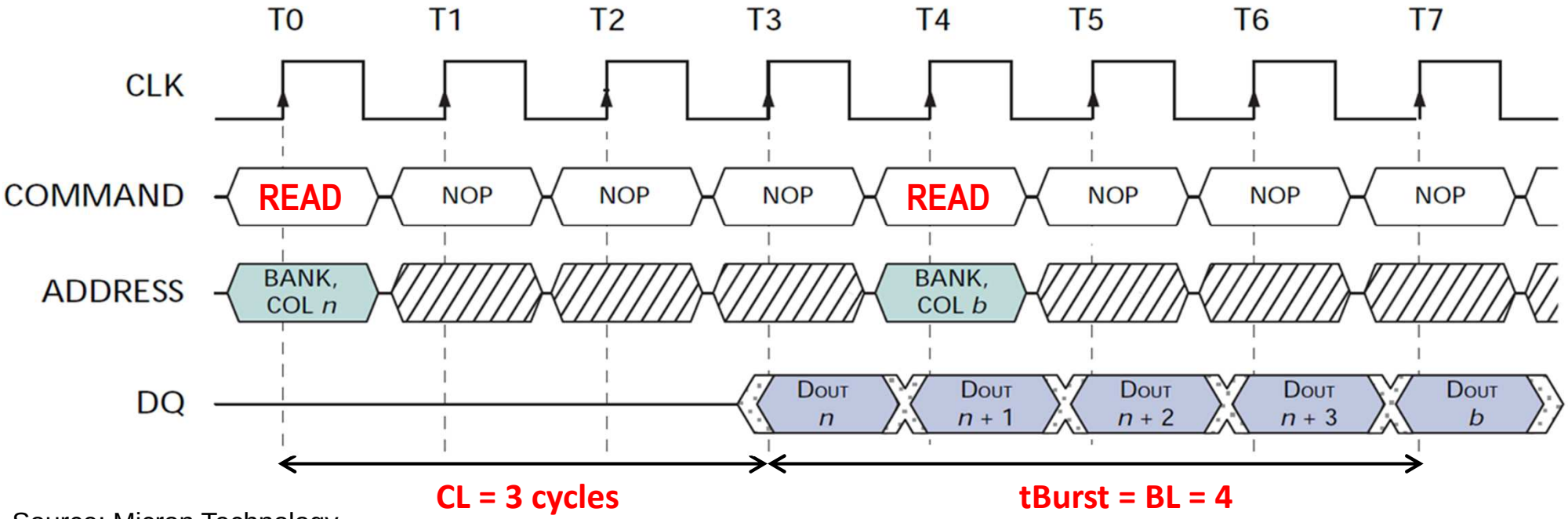
Source: Micron Technology

READ and WRITE Commands

- Read and Write commands are burst oriented
  - Start at a specific column address and continue for a number of locations
  - **Burst Length:** BL = 4 or 8 columns in most SDRAMs (Mode register)
- **Read:** Initiate a burst read access to an active row in a bank
  - Bank and Column addresses are latched and decoded
  - Column address specifies the start address of data within an active row
  - Multiple columns are output on the data bus, according to BL
- **Write:** Initiate a burst write access to an active row in a bank
  - Bank and Column addresses are latched and decoded (same as READ)
  - Input data is received on the data bus, according to BL
  - Input data is stored in the open row starting at the column address

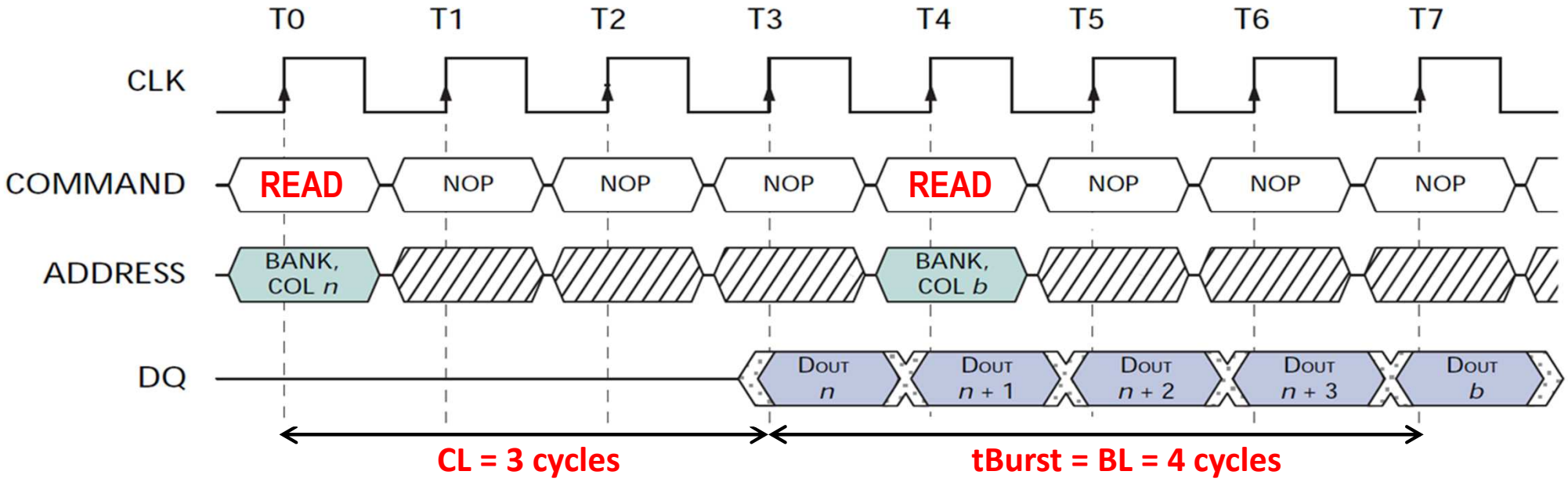
Timing of Read Command

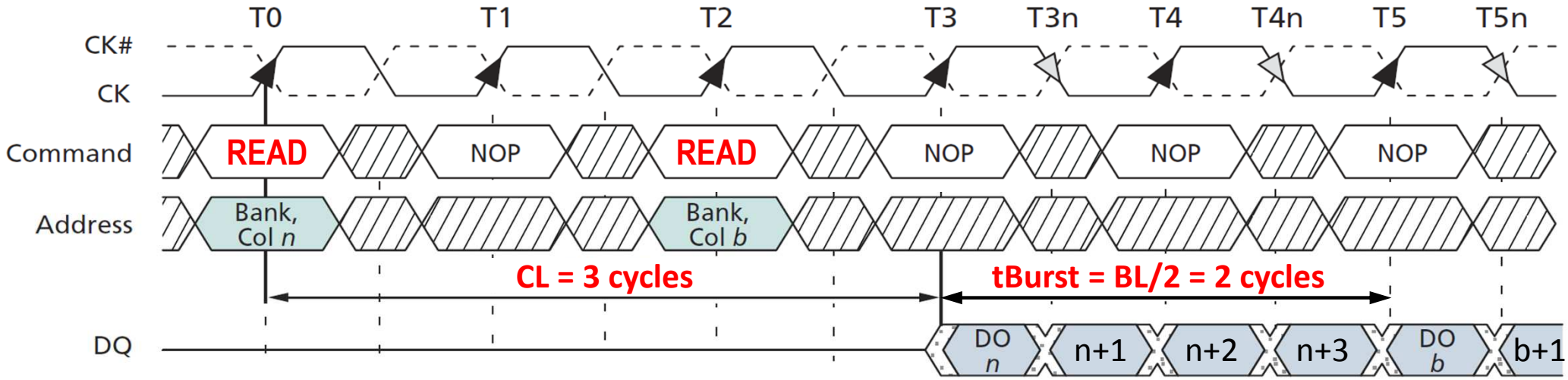
- Column Latency: **CL** or **tCAS**
  - Number of cycles between READ and first output data on the data bus
- Burst Time: **tBurst** = number of cycles for a burst read or write
  - For single data rate, **tBurst = BL** (Older SDRAMs)
  - For double data rate, **tBurst = BL/2**



Source: Micron Technology

Single versus Double Data Rate (BL = 4)

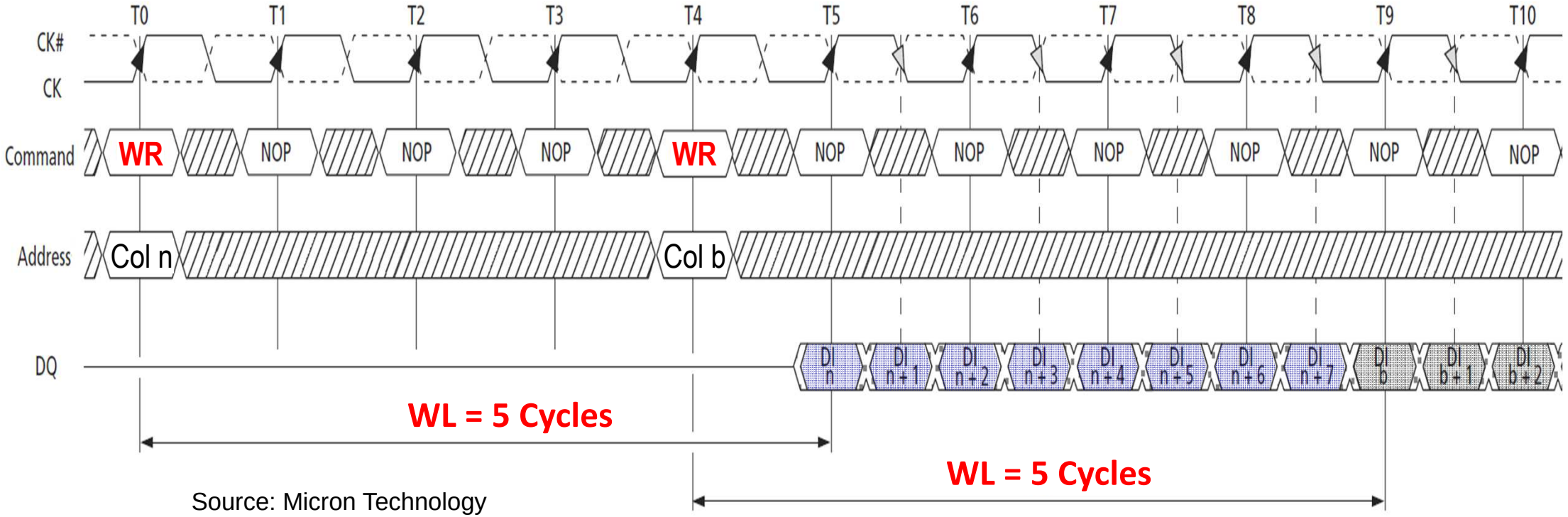




Number of cycles between two consecutive READ commands = tBurst

Timing of Write Command

- Write Latency: **WL**
  - Number of cycles between WRITE and first input data on the data bus
- In the following timing diagram …
  - WL = 5 cycles, BL = 8, tBurst = BL / 2 = 4 cycles
  - Two consecutive WRITE commands with tBurst = 4 cycles in between



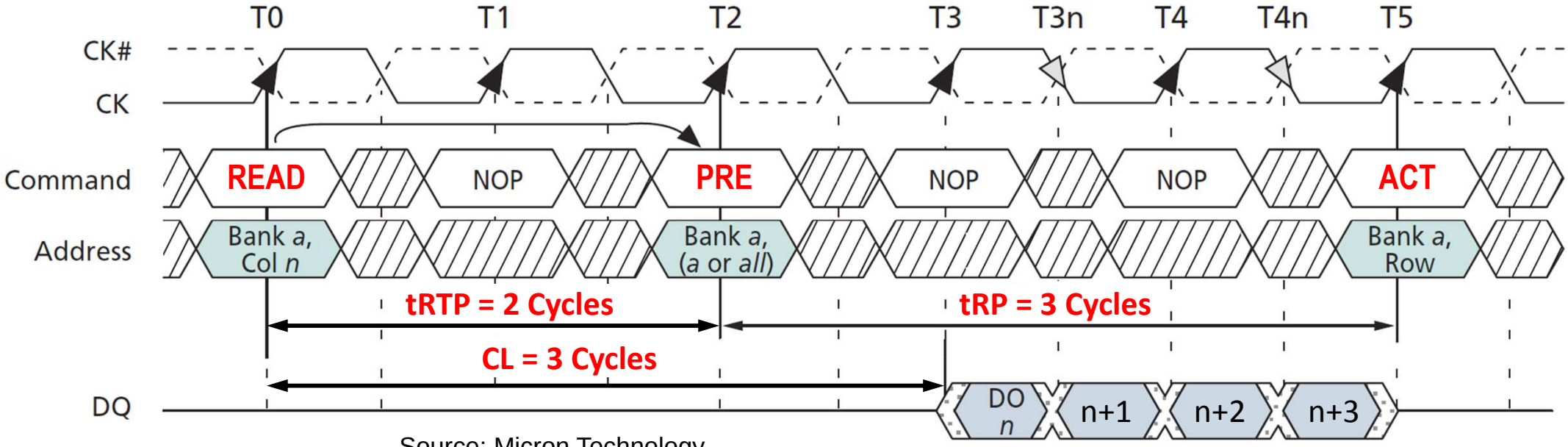
Source: Micron Technology

Precharge Commands

- **Precharge:** Close an active row of a selected bank
  - This command causes all the bits of an active row to be written back into the memory array of the selected bank
  - Only the Bank Address (BA) is required as input
  - After a bank has been precharged, it becomes in the idle state. It must be activated again prior to any new read or write commands
- **Precharge ALL:** All banks are precharged
  - All active rows are closed in parallel (No need for a bank address)
- **Read with Precharge:**
  - Row is read then closed in one command (Bank becomes idle)
- **Write with Precharge:**
  - Row is written then closed in one command (Bank becomes idle)

READ then Precharge

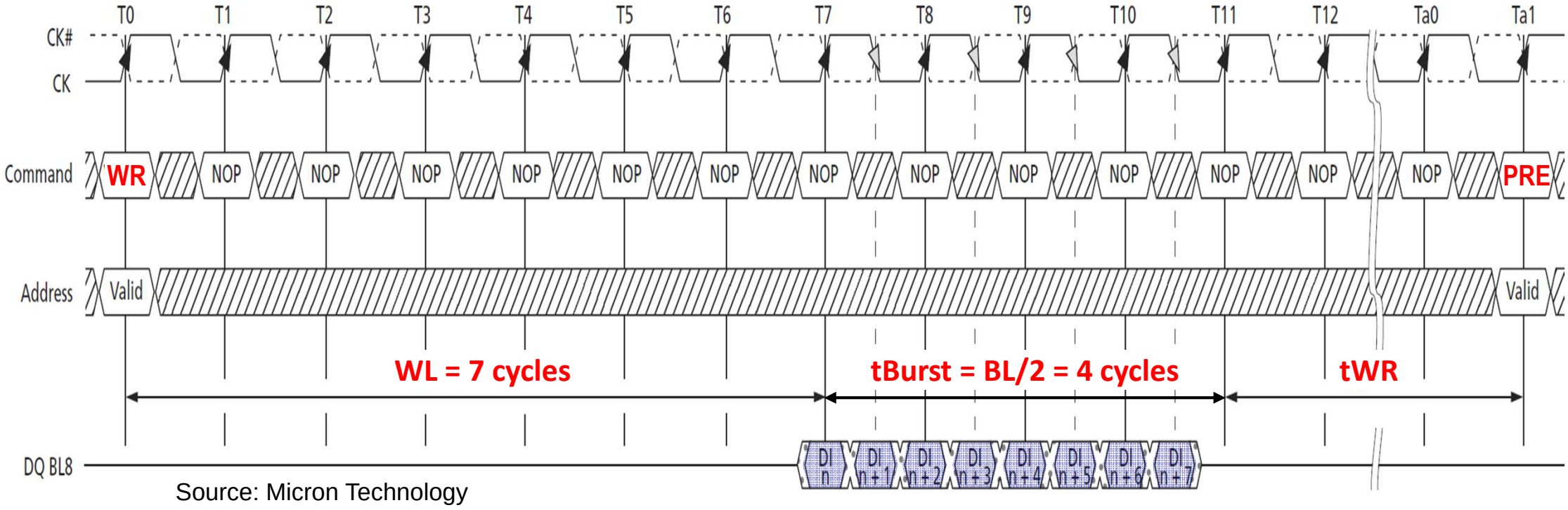
- READ-to-Precharge Latency = **tRTP**
  - Minimum cycles between READ and next PRECHARGE command
- Row Precharge Latency: **tRP**
  - Minimum cycles between PRECHARGE and next ACTIVE command
- In the diagram: tRTP = 2 cycles, and CL = tRP = 3 cycles



Source: Micron Technology

Write then Precharge

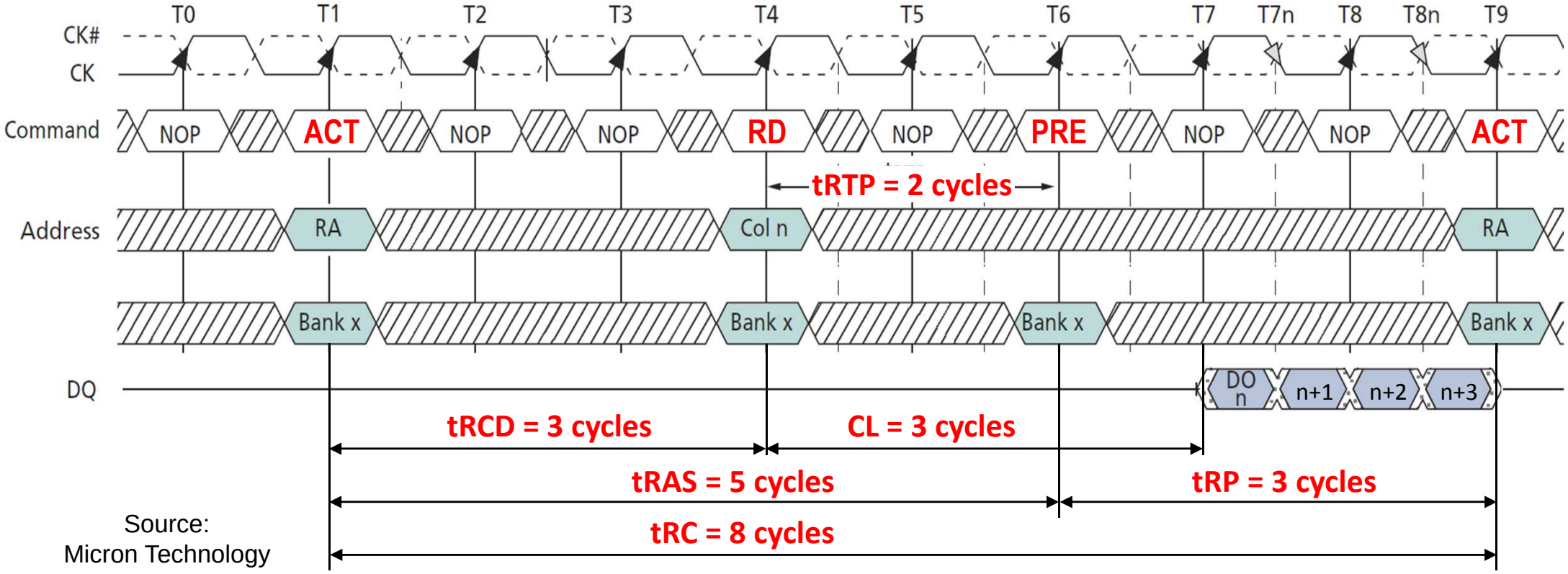
- Write Latency: **WL**
  - Number of cycles between WRITE and first input data on the data bus
- Write Recovery Time: **tWR**
  - Minimum number of cycles between last input data and PRECHARGE
  - Input data must be written in the open row before PRECHARGE



Source: Micron Technology

Row Active and Cycle Times

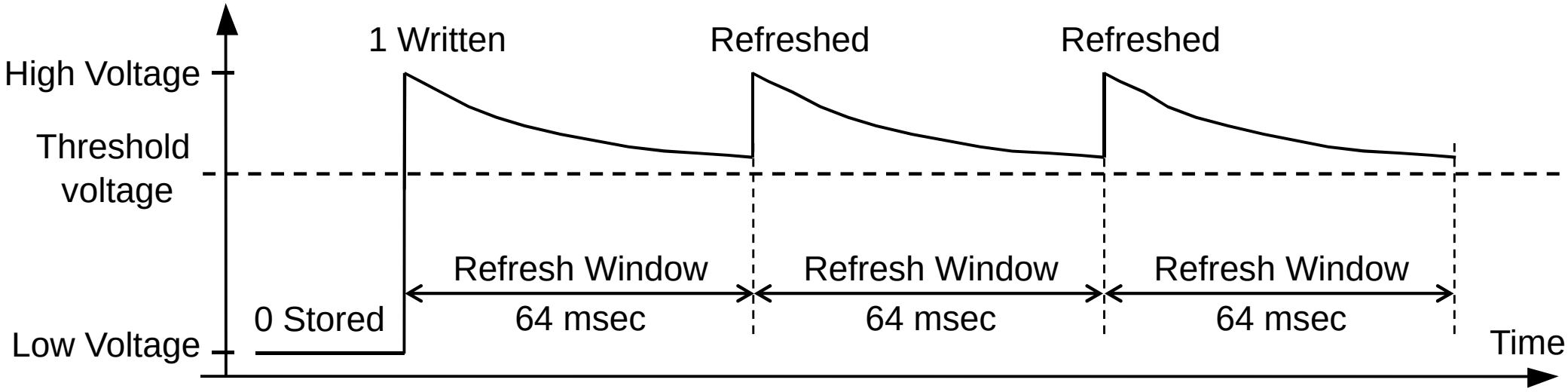
- Row Active Time: **tRAS**
  - Minimum cycles between ACTIVE and next PRECHARGE command
- Row Cycle Time: **tRC = tRAS + tRP**
  - Minimum cycles between two ACTIVE commands to same bank



Source:
Micron Technology

Refresh Window

- The capacitors in a DRAM are not perfect and leak their charge
- A capacitor's voltage drops over time, but should not be allowed to drop below a threshold voltage to avoid the loss of bits
- **Refresh Window:** time interval in which **all rows** are refreshed
  - Refresh window is typically 64 msec, according to recent standards
  - **All rows** must be **read and rewritten** (refreshed) at least once every 64 ms

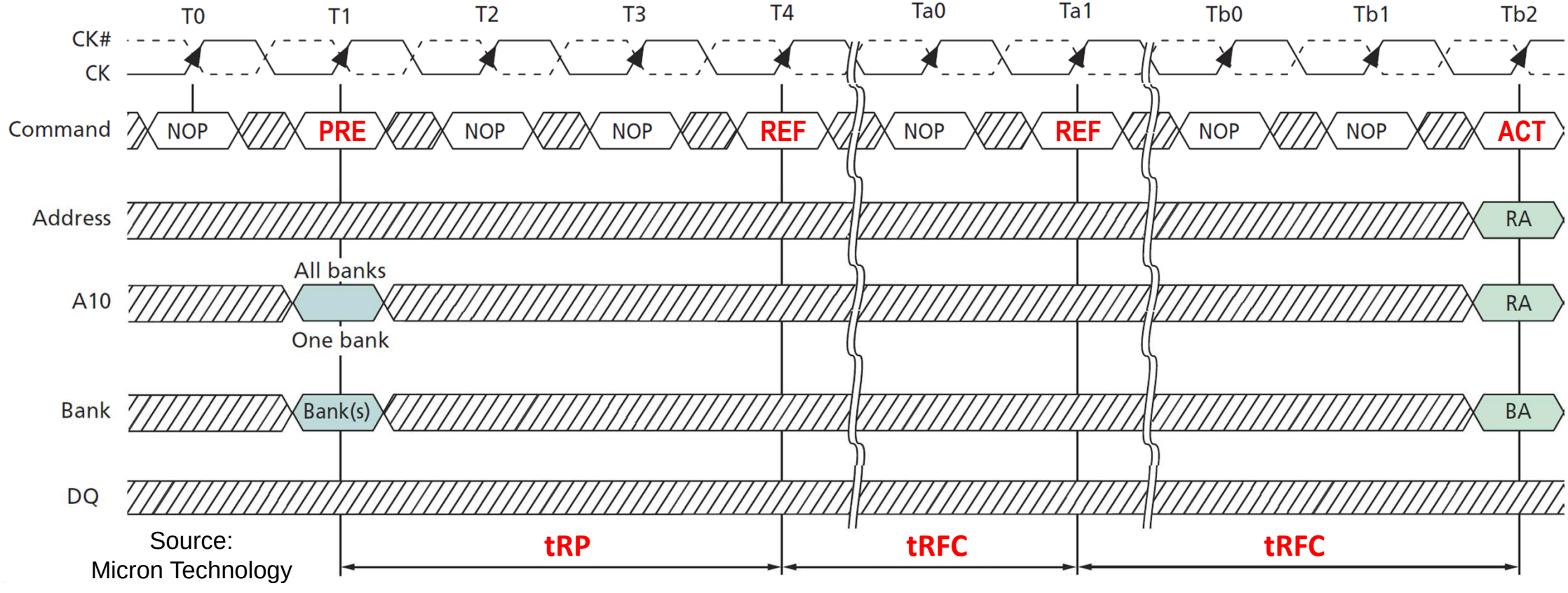


Refresh Commands

- **Refresh:** used during normal operation of the SDRAM
  - **All banks** must be **precharged** (idle state) prior to issuing this command
  - Same row address across **ALL banks** is refreshed
  - Row address is specified internally by a **refresh counter** in the SDRAM
  - The refresh counter is incremented after completing this command
  - This command must be repeated to refresh all rows
  - Refresh commands can be distributed or back-to-back
- **Self Refresh:** retains data in the SDRAM when system is idle
  - Self Refresh retains data without external clocking (clock is disabled)
  - Refreshing is done internally in the SDRAM
  - Reduces power consumption

Timing of Refresh Operations

- Refresh Cycle Time: **tRFC**
  - Number of clock cycles to refresh a single row across all banks
  - Minimum clock cycles between REFRESH and ACTIVATE commands
- Refreshing can be back-to-back or distributed



Source:
Micron Technology

Loss of Bandwidth to Refresh Operations

- During a REFRESH operation, all banks must be **idle**
- Some memory bandwidth is lost to REFRESH operations
- Example:
  - 2 Gbit DRAM organized as: 8 banks, 8K rows × 2K columns × 16 bits
  - Refresh window = 64 msec (specified by standard)
  - Refresh operation takes 40 ns (tRFC = 40 ns)
  - What fraction of the memory bandwidth is lost to refresh operations?
  - If refresh commands are distributed, what is the average refresh interval?
- **Solution:**
  - Refreshing all 8K rows takes: 8 × 1024 × 40 ns = 327680 ns
  - Loss of 327680 ns every 64 ms
  - Fraction of lost memory bandwidth = 0.32768 / 64 = 0.512%
  - Average refresh interval = 64 msec / 8192 rows = 7.8125 µsec

Summary of Timing Parameters

1. Row-to-Column Delay: **tRCD**
   - ✧ Minimum cycles between ACTIVE and READ/WRITE commands
2. Row-to-Row Delay: **tRRD**
   - ✧ Minimum cycles between two ACTIVE commands to different banks
3. Column Latency: **CL** (or **tCAS**)
   - ✧ Number of cycles between READ and first output data on the data pins
4. Write Latency: **WL**
   - ✧ Number of cycles between WRITE and first input data on the data pins
5. Write Recovery Time: **tWR**
   - ✧ Minimum cycles between last input data and PRECHARGE command
6. Burst Length and Time: **tBurst = BL / 2** (DDR)

More Timing Parameters

7. Row Precharge Latency: **tRP**
   - ✧ Minimum cycles between PRECHARGE and next ACTIVE command
8. Row Active Time: **tRAS**
   - ✧ Minimum cycles between ACTIVE and next PRECHARGE command
9. Row Cycle Time is calculated as: **tRC = tRAS + tRP**
   - ✧ Minimum cycles between two ACTIVE commands to same bank
10. Refresh Cycle Time: **tRFC**
    - ✧ Minimum cycles between REFRESH and next ACTIVE command

❖ Other timing parameters exist

- ✧ Omitted here for simplicity

Converting Timing Parameters into Latency

- Memory timing is often described with three parameters:

  Example: **7-8-8** means **CL = 7,** and **tRCD = tRP = 8** cycles

- Smaller numbers are usually better, but can be misleading
- Example: consider two memory modules
  - DDR3-1600 memory with timings: CL-tRCD-tRP = 8-8-8 cycles
  - DDR3-2666 memory with timings: CL-tRCD-tRP = 12-12-12 cycles
  - Which memory module has lower latency?
- **Solution:**
  - DDR3-1600 ➔ Bus Clock = 800 MHz ➔ Cycle = 1.25 ns ➔ CL = 10 ns
  - DDR3-2666 ➔ Bus Clock = 1333 MHz ➔ Cycle = 0.75 ns ➔ CL = 9 ns
  - DDR3-2666 has lower latencies (CL, tRCD, tRP) than DDR3-1600

Transfer Rates & Peak Bandwidth

Standard Name	Memory Bus Clock	Millions Transfers per second	Module Name	Peak Bandwidth
DDR-200	100 MHz	200 MT/s	PC-1600	1600 MB/s
DDR-333	167 MHz	333 MT/s	PC-2700	2667 MB/s
DDR-400	200 MHz	400 MT/s	PC-3200	3200 MB/s
DDR2-667	333 MHz	667 MT/s	PC-5300	5333 MB/s
DDR2-800	400 MHz	800 MT/s	PC-6400	6400 MB/s
DDR2-1066	533 MHz	1066 MT/s	PC-8500	8533 MB/s
DDR3-1066	533 MHz	1066 MT/s	PC-8500	8533 MB/s
DDR3-1333	667 MHz	1333 MT/s	PC-10600	10667 MB/s
DDR3-1600	800 MHz	1600 MT/s	PC-12800	12800 MB/s
DDR4-3200	1600 MHz	3200 MT/s	PC-25600	25600 MB/s

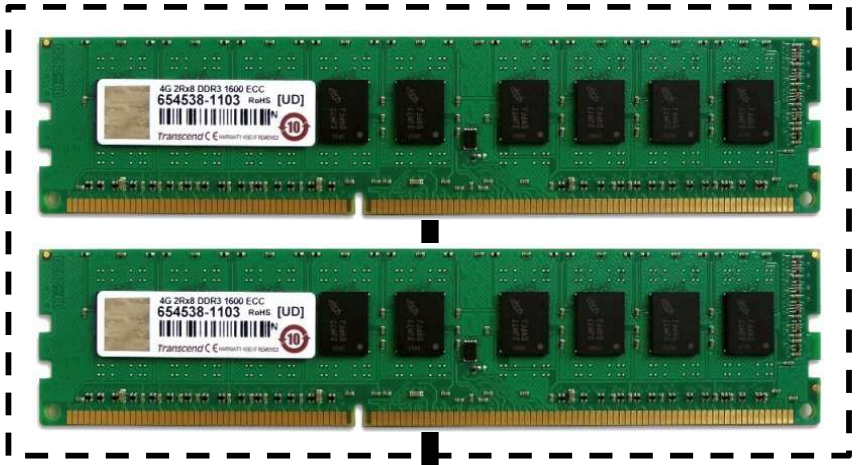
1 Transfer = 64 bits = 8 bytes of data (72 bits = 9 bytes for ECC)

Multiple Memory Channels

- First introduced in the IBM System 360/91 in the 1960s
- Multiple memory channels use **separate data buses**
- They increase the memory bandwidth
  - **Total Bandwidth = Bandwidth per channel × Number of channels**
- Modern processors support multiple memory channels
  - Number of channels can vary between 2 and 8
- Memory controller must support multiple memory channels
- The number of I/O pins is increased on the processor chip
- The number of memory modules is also increased
  - For best performance, channels must use identical memory modules

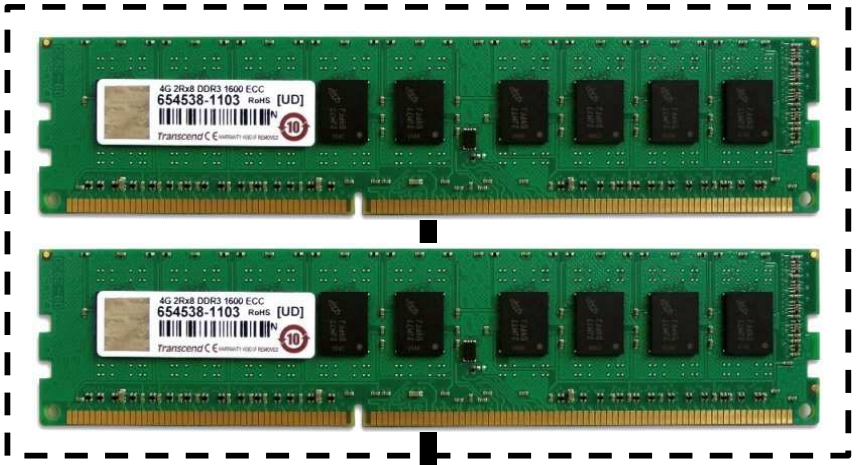
Memory Channels, Ranks, and Banks

Multiple DIMMs per Channel
Multiple Ranks per DIMM
Multiple Banks per Rank
The Banks exist inside the chips



Memory Channel 0

Multiple DIMMs per Channel
Multiple Ranks per DIMM
Multiple Banks per Rank
The Banks exist inside the chips



Memory Channel 1

Address Mapping

- Many ways of mapping addresses to channels, ranks, and banks
- Row (or Page) Interleaving ➔ favors **row locality**
  - Consecutive memory addresses are mapped to rows ➔ **row hit**
  - Then rows are mapped to channels, ranks, and banks
  - Different ways to map rows to channels, ranks, and banks
  - Byte offset is 3 bits because the data bus is 8-byte wide

3 bits

Row Address	**Bank**	**Rank**	**Ch**	**Column Address**	**Byte**
Row Address	**Rank**	**Bank**	**Ch**	**Column Address**	**Byte**
Row Address	**Rank**	**Ch**	**Bank**	**Column Address**	**Byte**
Row Address	**Bank**	**Ch**	**Rank**	**Column Address**	**Byte**
Row Address	**Ch**	**Rank**	**Bank**	**Column Address**	**Byte**
Row Address	**Ch**	**Bank**	**Rank**	**Column Address**	**Byte**

Address Mapping (cont'd)

- Cache Block Interleaving
  - Consecutive cache blocks are mapped to different channels
  - Cache block size = 64 bytes ➔ Byte offset = 6 bits
  - Burst Length = 8 transfers = 64 bytes
  - The lower 3 bits of the column address are used in the byte offset
  - High column = Column address excluding the lower 3 bits
  - Favors **Parallelism** ➔ Channel **Ch** appears next to **Byte offset**