İstanbul
Bilgi Üniversitesi

CMPE321 - Computer Architecture

Lecture 5
Memory Basics

Hakan Ayral, PhD.

Definitions

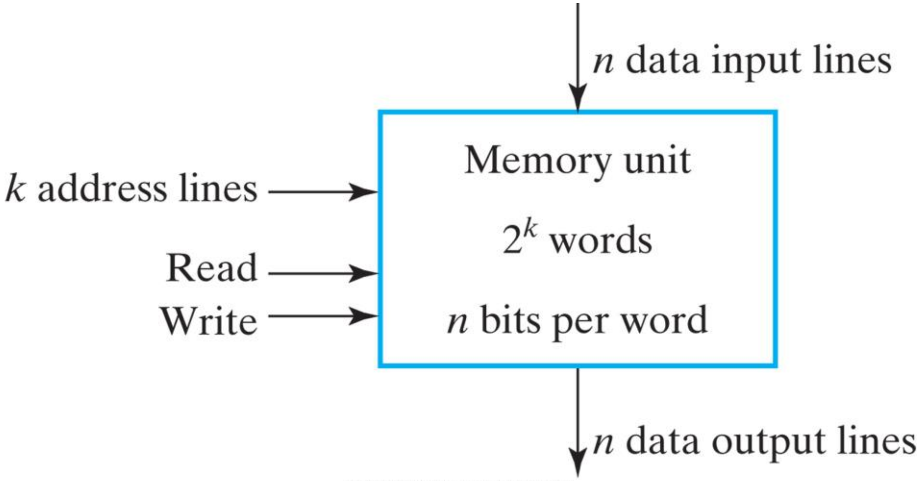
- Two types of memories are used in various parts of a computer: **random access memory** (RAM) and **read-only memory** (ROM).
- RAM accepts new information for storage to be available later for use.
- The process of storing new information in memory is referred to as a **memory write operation**.
- The process of transferring the stored information out of memory is referred to as a **memory read operation**.
- RAM can perform both the write and the read operations, whereas ROM performs only read operations.
- RAM sizes may range from hundreds to billions of bits.

Random-Access Memory

- Binary information is stored in memory in groups of bits, each group of which is called a ***word***.
- A word is an entity of bits that moves in and out of memory as a unit—a group of 1s and 0s that represents a number, an instruction, one or more alphanumeric characters, or other binary-coded information.
- A group of eight bits is called a ***byte***.
- Most computer memories use words that are multiples of eight bits in length.
- Thus, a 16-bit word contains two bytes, and a 32-bit word is made up of four bytes.
- The capacity of a memory unit is usually stated as the total number of bytes that it can store.
- Communication between a memory and its environment is achieved through **data input** and **output** lines, **address selection** lines, and **control** lines that specify the direction of transfer of information.

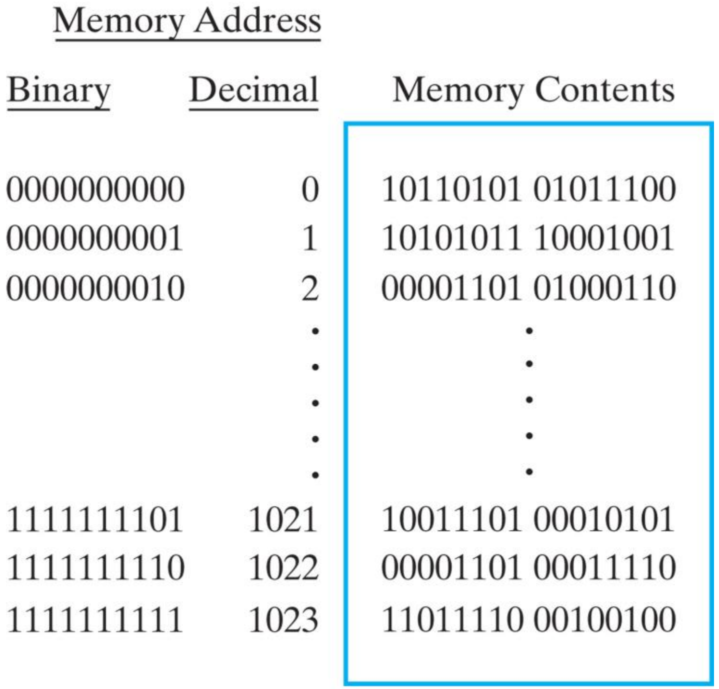
Random-Access Memory

- A block diagram of a memory is shown in Figure.
- The ***n* data input lines** provide the information to be stored in memory, and the ***n* data output lines** supply the information coming out of memory.
- The ***k* address lines** specify the word chosen among all the available.
- The **two control inputs** specify the direction of transfer desired:
  - the Write input causes binary data to be transferred into memory,
  - the Read input causes binary data to be transferred out of memory.
- The memory unit is specified by the <u>number of words </u>it contains and the <u>number of bits in each word</u>.
- The address lines select one specific word.
- Each word in memory is assigned an identification number called an *address*.



Random-Access Memory

- Addresses range from 0 to 2^*k* - 1, where *k* is the number of address lines.
- The selection of a specific word inside memory is done by applying the *k*-bit binary address to the address lines.
- A decoder accepts this address and opens the paths needed to select the word specified.
- It is customary to refer to the number of words (or bytes) in memory with one of the letters K (kilo), M (mega), or G (giga); where K is equal to 2^10, M to 2^20, and G to 2^30.
- Consider, for example, a memory with a capacity of 1K words of 16 bits each.
- Since 1K = 1024 = 2^10, and 16 bits constitute two bytes, we can say that the memory can accommodate 2048, or 2K, bytes.
  - Each word contains 16 bits that can be divided into two bytes.
- The words are recognized by their decimal addresses from 0 to 1023.
  - An equivalent binary address consists of 10 bits.
  - The first address is specified using ten 0s, and the last address is specified with ten 1s. This is because 1023 in binary is equal to 1111111111.



Memory Address		
Binary	Decimal	Memory Contents
0000000000	0	10110101 01011100
0000000001	1	10101011 10001001
0000000010	2	00001101 01000110
	·	·
	·	·
	·	·
	·	·
	·	·
1111111101	1021	10011101 00010101
1111111110	1022	00001101 00011110
1111111111	1023	11011110 00100100

Random-Access Memory

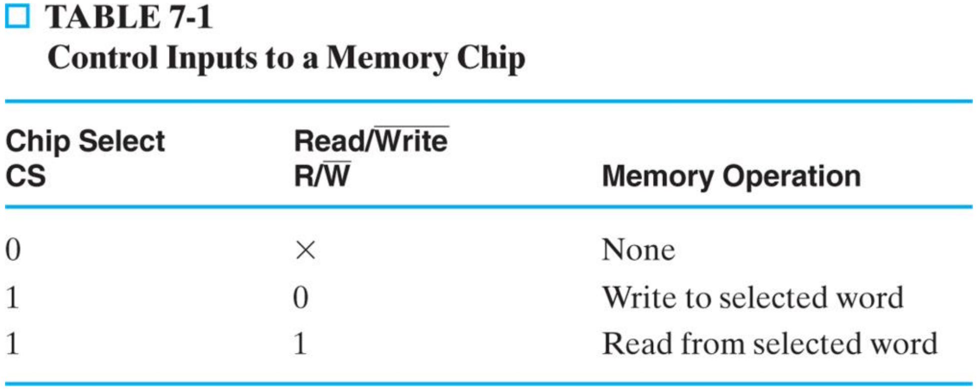
- A word in memory is selected by its binary address.
- When a word is read or written, the memory on previous figure operates on all 16 bits as a single unit.
- The 1K×16 memory on the figure has 10 bits in the address and 16 bits in each word.
- The number of address bits needed in memory is dependent on the total number of words that can be stored and is independent of the number of bits in each word.
- The number of bits in the address for a word is determined from the relationship $2^k \geq m$, where *m* is the total number of words and *k* is the minimum number of address bits satisfying the relationship.

Read and Write Operations

- The two operations that a random-access memory can perform are write and read.
  - A ***write*** is a transfer into memory of a new word to be stored.
  - A ***read*** is a transfer of a copy of a stored word out of memory.
- A Write signal specifies the transfer-in operation, and a Read signal specifies the transfer-out operation.
- On accepting one of these control signals, the internal circuits inside memory provide the desired function.
- The steps that must be taken for a write are as follows:
  1. Apply the binary address of the desired word to the address lines.
  2. Apply the data bits that must be stored in memory to the data input lines.
  3. Activate the Write input.
- The memory unit will then take the bits from the data input lines and store them in the word specified by the address lines.
- The steps that must be taken for a read are as follows:
  1. Apply the binary address of the desired word to the address lines.
  2. Activate the Read input.
- The memory will then take the bits from the word that has been selected by the address and apply them to the data output lines. The contents of the selected word are not changed by reading them.

Read and Write Operations

- Memory is made up of RAM integrated circuits (chips), plus additional logic circuits.
- RAM chips usually provide the two control inputs for the read and write operations in a somewhat different configuration from that just described.
- Instead of having separate Read and Write inputs to control the two operations, most integrated circuits provide at least a Chip Select that selects the chip to be read from or written to and a $\text{Read}/\overline{\text{Write}}$ that determines the particular operation.
- The memory operations that result from these control inputs are shown in Table below.
- The Chip Select is used to enable the particular RAM chip or chips containing the word to be accessed.
  - When Chip Select is inactive, the memory chip or chips are not selected, and no operation is performed.
  - When Chip Select is active, the $\text{Read}/\overline{\text{Write}}$ input determines the operation to be performed.
  - While Chip Select accesses chips, a signal is also provided that accesses the entire memory. We will call this signal the Memory Enable.



□ **TABLE 7-1**
Control Inputs to a Memory Chip

Chip Select CS	$\text{Read}/\overline{\text{Write}}$ $R/\overline{W}$	Memory Operation
0	×	None
1	0	Write to selected word
1	1	Read from selected word

Timing Waveforms

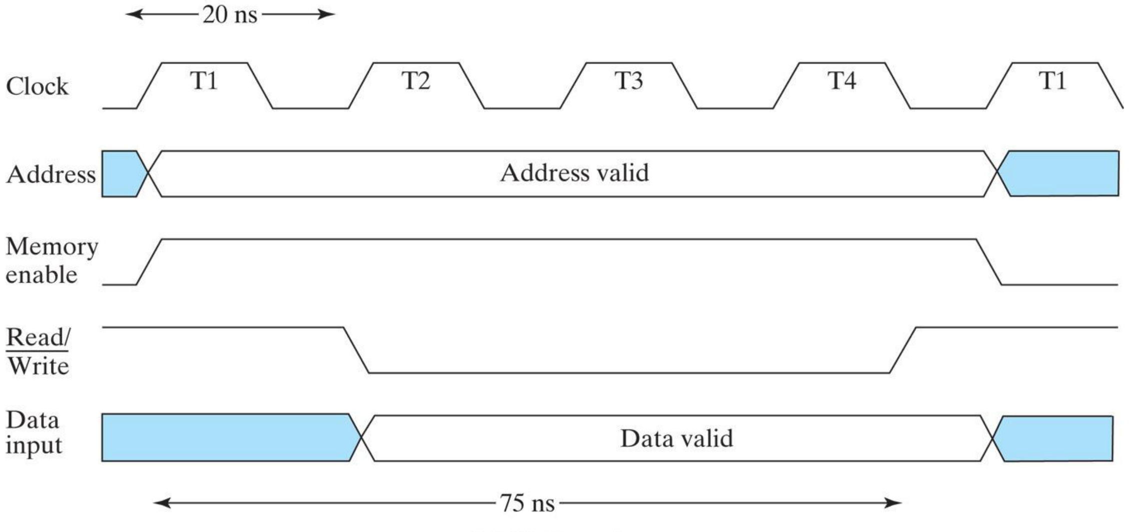
- The operation of the memory unit is controlled by an external device, such as a CPU.
- The CPU is synchronized by its own clock pulses. The memory, however, does not employ the CPU clock. Instead, its read and write operations are timed by changes in values on the control inputs.
- The *access time* of a memory read operation is the maximum time from the application of the address to the appearance of the data at the Data Output.
- Similarly, the *write cycle time* is the maximum time from the application of the address to the completion of all internal memory operations required to store a word.
- Memory writes may be performed one after the other at the intervals of the cycle time.
- The CPU must provide the memory control signals in such a way as to synchronize its own internal clocked operations with the read and write operations of memory.
- This means that the access time and the write cycle time of the memory must be related within the CPU to a period equal to a fixed number of CPU clock periods.

Timing Waveforms

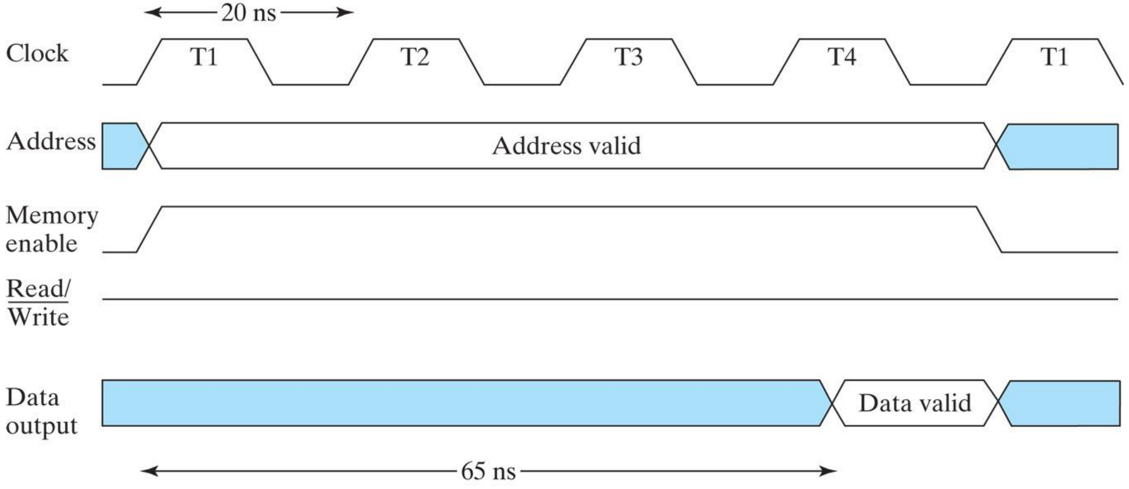
- Assume, as an example, that a CPU operates with a clock frequency of 50 MHz, giving a period of 20 ns (1 ns = 10^-9 sec) for one clock pulse.
- Suppose now that the CPU communicates with a memory with an access time of 65 ns and a write cycle time of 75 ns.
- The number of clock pulses required for a memory request is the integer value greater than or equal to the larger of the access time and the write cycle time, divided by the clock period.
- Since the period of the CPU clock is 20 ns, and the larger of the access time and write cycle time is 75 ns, it will be necessary to devote at least four clock pulses to each memory request.

Timing Waveforms

- The memory cycle timing shown in Figure is for a CPU with a 50 MHz clock and memory with a 75 ns write cycle time and a 65 ns access time.
- The write cycle in part (a) shows four pulses *T*1, *T*2, *T*3, and *T*4 with a cycle of 20 ns.
- For a write operation, the CPU must provide the address and input data to the memory.
- The address is applied, and Memory Enable is set to the high level at the positive edge of the *T*1 pulse.
- The data, needed somewhat later in the write cycle, is applied at the positive edge of *T*2.
- The two lines that cross each other in the address and data waveforms designate a possible change in value of the multiple lines.
- The shaded areas represent unspecified values.
- A change of the R/$\overline{\text{W}}$ signal to 0 to designate the write operation is also at the positive edge of *T*2.
- To avoid destroying data in other memory words, it is important that this change occur after the signals on the address lines have become fixed at the desired values.
- Otherwise, one or more other words might be momentarily addressed and accidentally written over with different data.



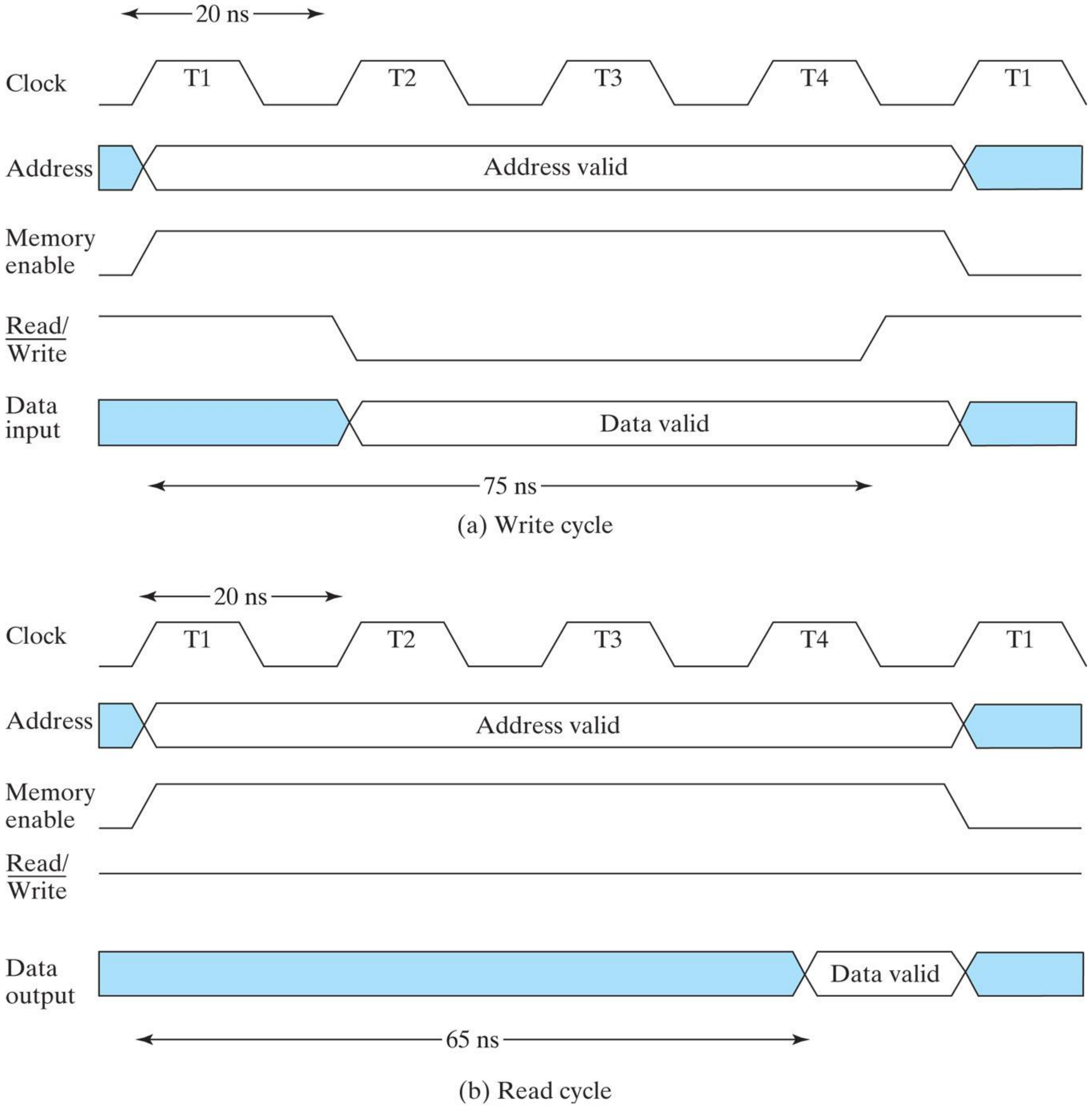
(a) Write cycle



(b) Read cycle

Timing Waveforms

- The R/$\overline{\text{W}}$ signal must stay at 0 long enough after application of the address and Memory Enable to allow the write operation to complete.
- Finally, the address and data signals must remain stable for a short time after the R/$\overline{\text{W}}$ goes to 1, again to avoid destroying data in other memory words.
- At the completion of the fourth clock pulse, the memory write operation has ended with 5 ns to spare, and the CPU can apply the address and control signals for another memory request with the next T1 pulse.
- The read cycle shown in previous Figure’s (b) has an address for the memory that is provided by the CPU.
- The CPU applies the address, sets the Memory Enable to 1, and sets to 1 to designate a read operation, all at the positive edge of *T*1.
- The memory places the data of the word selected by the address onto the data output lines within 65 ns from the time that the address is applied and the memory enable is activated.
- Then, the CPU transfers the data into one of its internal registers during the positive transition of the next *T*1 pulse, which can also change the address and controls for the next memory request.



(a) Write cycle

(b) Read cycle

Properties of Memory

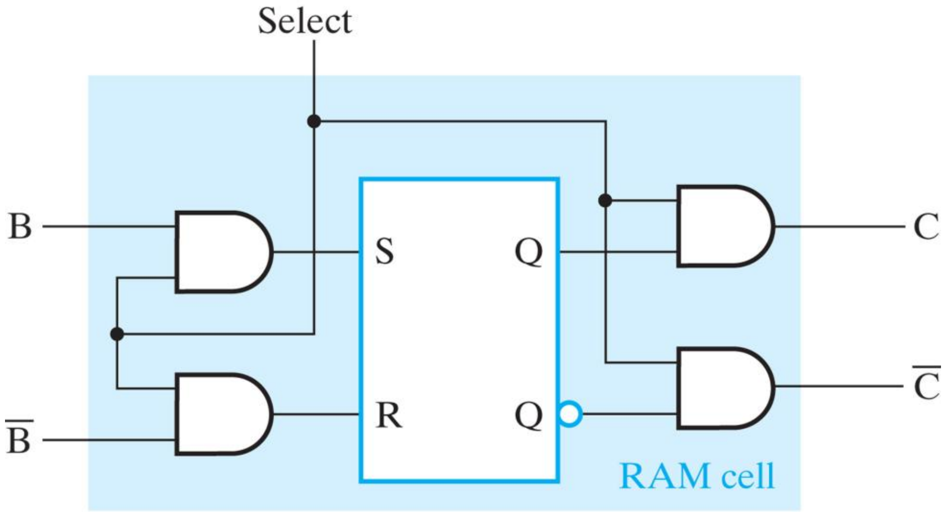
- Integrated-circuit RAM may be either **static** or **dynamic**.
- ***Static* RAM** (SRAM) consists of internal latches that store the binary information.
- The stored information remains valid as long as power is applied to the RAM.
- ***Dynamic* RAM** (DRAM) stores the binary information in the form of electric charges on capacitors.
- The capacitors are accessed inside the chip by *n*-channel MOS transistors.
- The stored charge on the capacitors tends to discharge with time, and the capacitors must be periodically recharged by ***refreshing*** the DRAM.
- This is done by cycling through the words every few milliseconds, reading and rewriting them to restore the decaying charge.
- DRAM offers reduced power consumption and larger storage capacity in a single memory chip, but SRAM is easier to use and has shorter read and write cycles.
- Also, <u>no refresh is required for SRAM</u>.
- Memory units that lose stored information when power is turned off are said to be ***volatile***.
- Integrated-circuit RAMs, both static and dynamic, are of this category, since the binary cells need external power to maintain the stored information.
- In contrast, a ***nonvolatile memory***, such as magnetic disk, retains its stored information after the removal of power.
- This is because the data stored on magnetic components is represented by the direction of magnetization, which is retained after power is turned off.
- Another nonvolatile memory is ROM.

SRAM Integrated Circuits

- As indicated earlier, memory consists of RAM chips plus additional logic.
- We will consider the internal structure of the RAM chip first.
- Then we will study combinations of RAM chips and additional logic used to construct memory.
- The internal structure of a RAM chip of ***m* words** with ***n* bits per word** consists of an array of *m×n* binary storage cells and associated circuitry.
- The circuity is made up of decoders to select the word to be read or written, read circuits, write circuits, and output logic.
- The ***RAM cell*** is the basic binary storage cell used in the RAM chip, which is typically designed as an electronic circuit rather than a logic circuit.
- Nevertheless, it is possible and convenient to model the RAM chip using a logic model.

SRAM Integrated Circuits

- We first present RAM cell logic on a static RAM chip for storing a single bit and then use the cell in a hierarchy to describe the RAM chip.
- Figure 4 shows the logic model of the RAM cell.
- The storage part of the cell is modeled by an *SR* latch.
- The inputs to the latch are enabled by a Select signal.
- For Select equal to 0, the stored content is held.

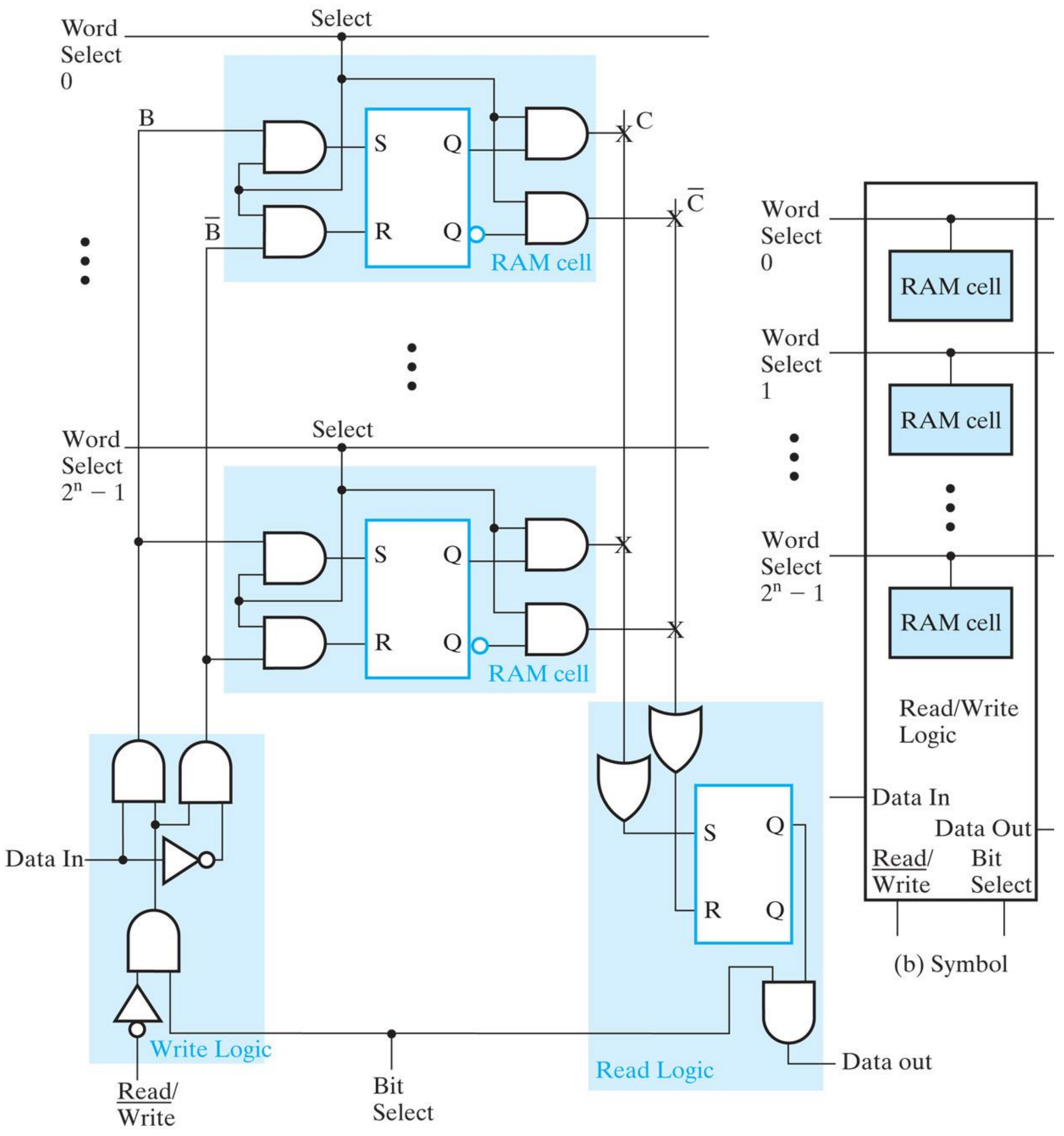


Copyright ©2016 Pearson Education, All Rights Reserved

- For Select equal to 1, the stored content is determined by the values on B and $\overline{B}$.
- The outputs from the latch are gated by Select to produce cell outputs C and $\overline{C}$.
- For Select equal to 0, both C and are 0, and for Select equal to 1, C is the stored value and $\overline{C}$ is its complement.

SRAM Integrated Circuits

- To obtain simplified static RAM diagrams, we interconnect a set of RAM cells and read and write circuits to form a ***RAM bit slice*** that contains all of the circuitry associated with a single bit position of a set of RAM words.
- The logic diagram for a RAM bit slice is shown in (a).
- The portion of the model representing each RAM cell is highlighted in blue.
- The loading of a cell latch is now controlled by a Word Select input.
- If this is 0, then both *S* and *R* are 0, and the cell latch contents remain unchanged. If the Word Select input is 1, then the value to be loaded into the latch is controlled by two signals *B* and $\overline{B}$ from the Write Logic.
- In order for either of these signals to be 1 and potentially change the stored value, Read/Write must be 0 and Bit Select must be 1.
- Then the Data In value and its complement are applied to *B* and $\overline{B}$, respectively, to set or reset the latch in the RAM cell selected.
- If Data In is 1, the latch is set to 1, and if Data In is 0, the latch is reset to 0, completing the write operation.

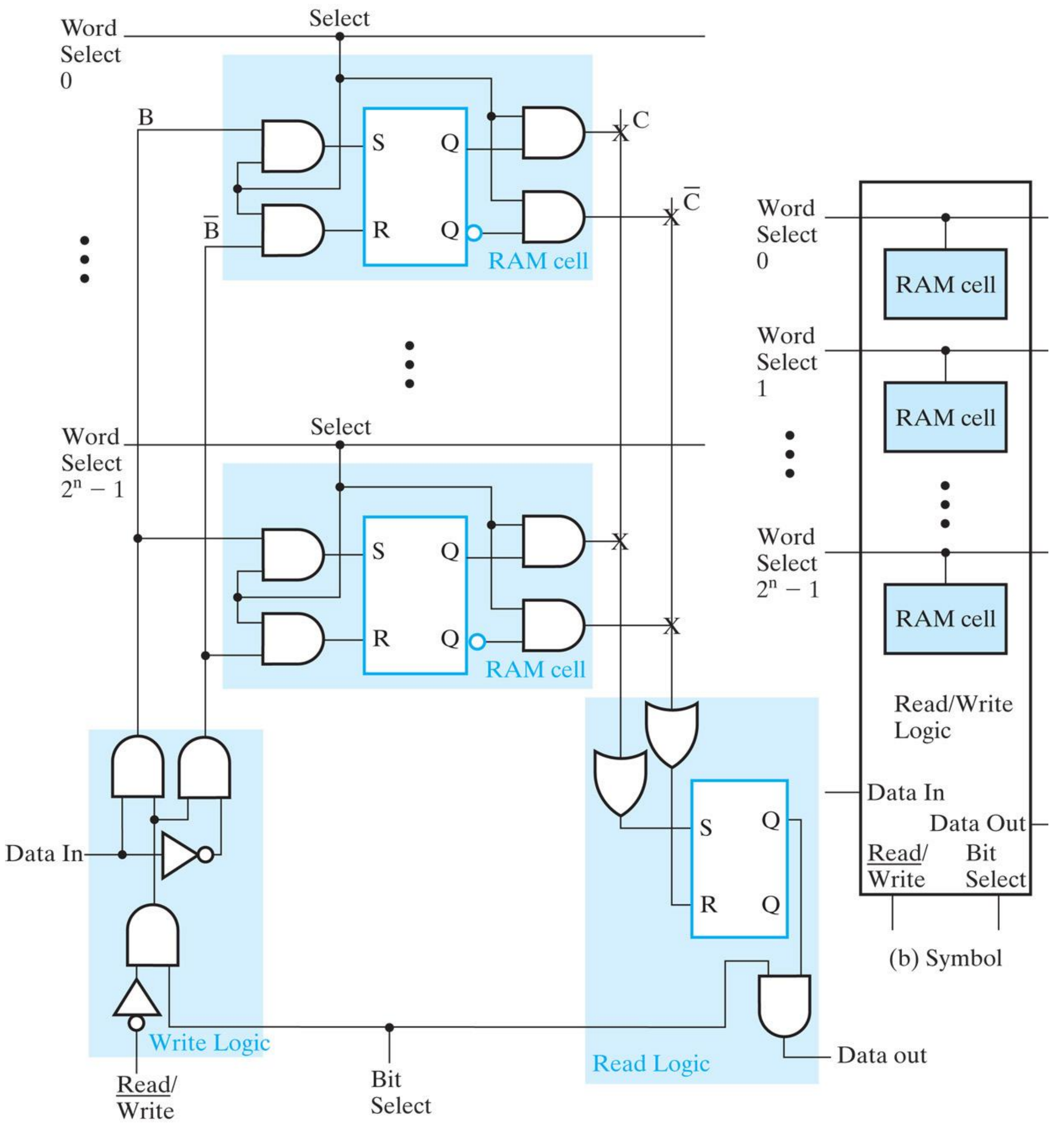


(b) Symbol

(a) Logic diagram

SRAM Integrated Circuits

- Only one word is written at a time. That is, only one Word Select line is 1, and all other Word Select lines are 0.
- Thus, only one RAM cell attached to *B* and $\overline{B}$ is written.
- The Word Select also controls the reading of the RAM cells, using shared Read Logic.
- If Word Select is 0, then the stored value in the *SR* latch is prevented by the AND gates from reaching the pair of OR gates in the Read Logic.
- But if Word Select is 1, the stored value passes through to the OR gates and is captured in the Read Logic *SR* latch.
- If Bit Select is also 1, the captured value appears on the Data Out line of the RAM bit slice.
- Note that for this particular Read Logic design, the read occurs regardless of the value of Read/$\overline{\text{Write}}$.
- The symbol for the RAM bit slice given in (b) is used to represent the internal structure of RAM chips.

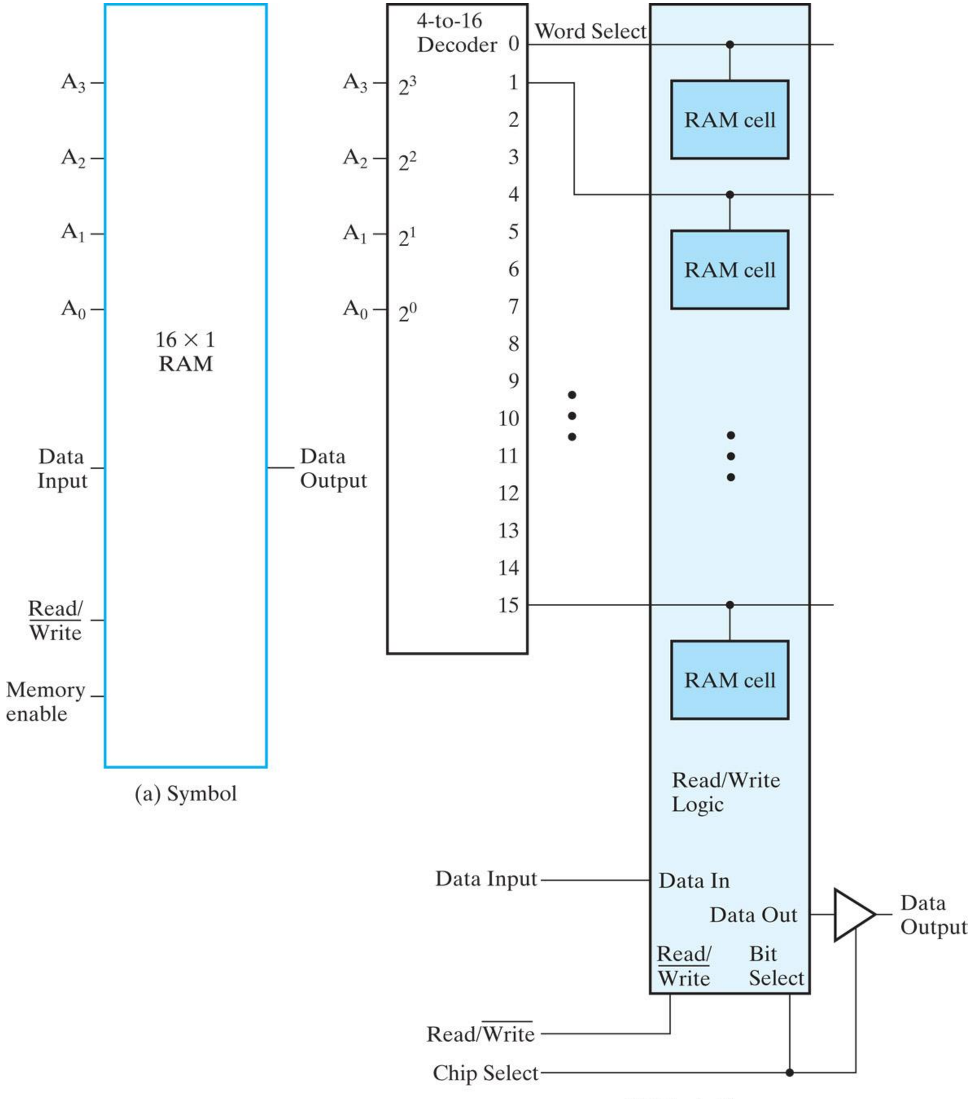


(b) Symbol

(a) Logic diagram

SRAM Integrated Circuits

- Each Word Select line extends beyond the bit slice, so that when multiple RAM bit slices are placed side by side, corresponding Word Select lines connect.
- The other signals in the lower portion of the symbol may be connected in various ways, depending on the structure of the RAM chip.
- The symbol and block diagram for a 16×1 RAM chip are shown in Figure.
- Both have four address inputs for the 16 one-bit words stored in RAM.
- There are also Data Input, Data Output, and Read/$\overline{\text{Write}}$ signals.
- The Chip Select at the chip level corresponds to the Memory Enable at the level of a RAM consisting of multiple chips.
- The internal structure of the RAM chip consists of a RAM bit slice having 16 RAM cells.
- Since there are 16 Word Select lines to be controlled such that one and only one has the value logic 1 at a given time, a 4-to-16-line decoder is used to decode the four address bits into 16 Word Select bits.

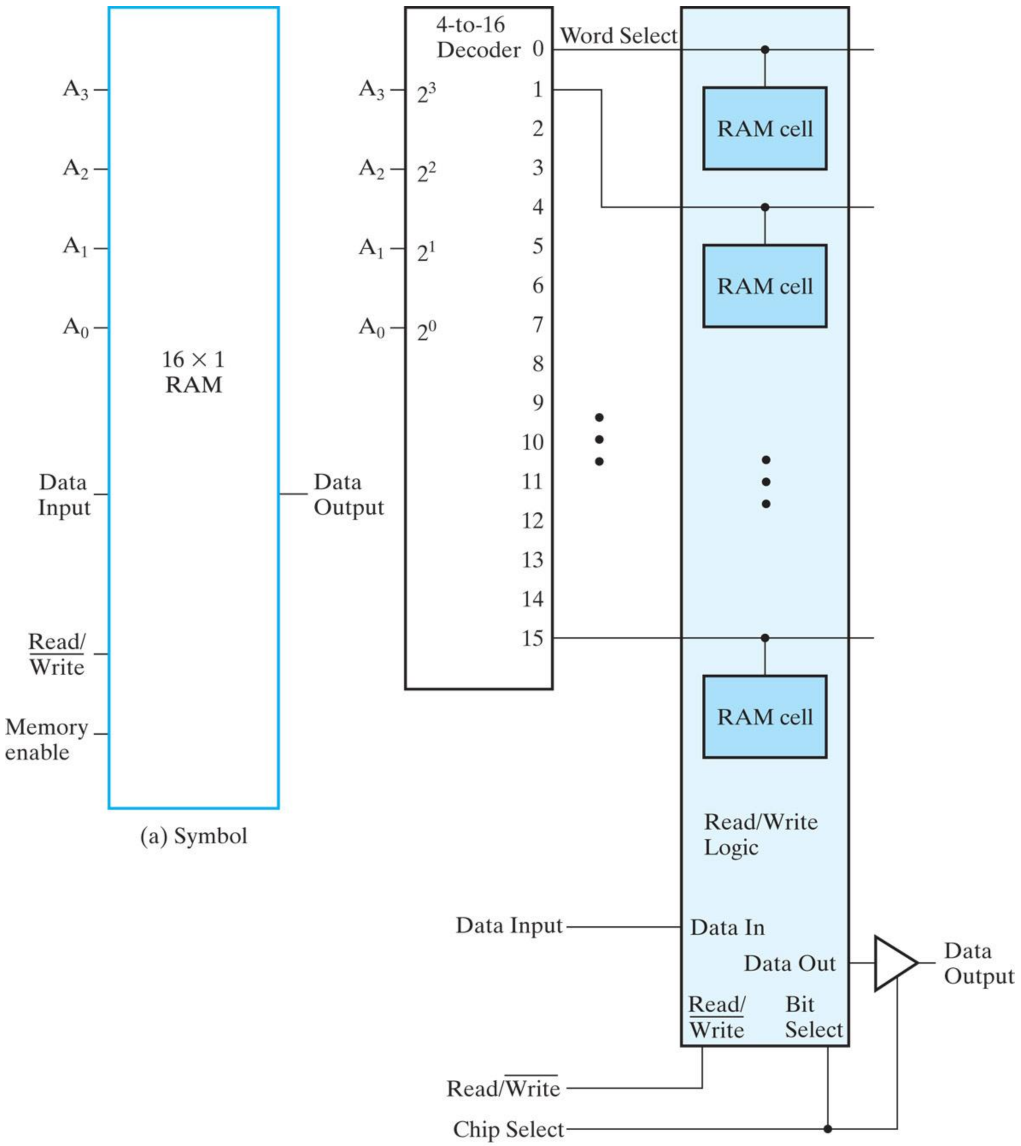


(a) Symbol

(b) Block diagram

SRAM Integrated Circuits

- The only additional logic in the figure is a triangular symbol with one normal input, one normal output, and a second input on the bottom of the symbol.
- This symbol is a three-state buffer that allows construction of a multiplexer with an arbitrary number of inputs.
- Three-state outputs are connected together and properly controlled using the Chip Select inputs.
- By using three-state buffers on the outputs of RAM chips, these outputs can be connected together to provide the word from the chip being read on the bit lines attached to the RAM outputs.
- The enable signals in the preceding discussion correspond to the Chip Select inputs on the RAM chips.
- To read a word from a particular RAM chip, the Chip Select value for that chip must be 1, and for all other chips attached to the same output bit lines, the Chip Select must be 0.
- These combinations containing a single 1 can be obtained from a decoder.



(a) Symbol

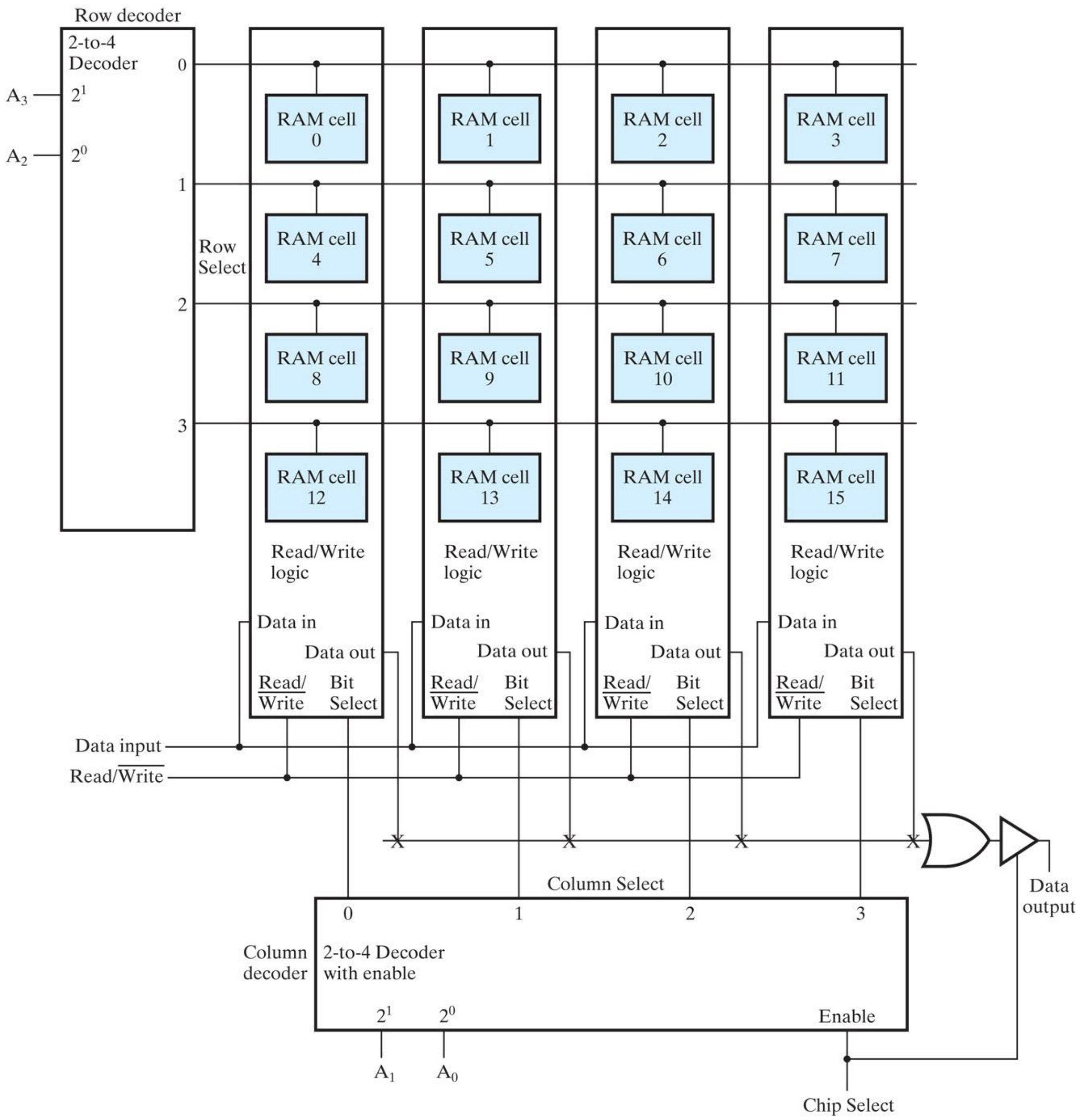
(b) Block diagram

Coincident selection

- Inside a RAM chip, the decoder with *k* inputs and 2^*k* outputs requires 2^*k* AND gates with *k* inputs per gate if a straightforward design approach is used.
- In addition, if the number of words is large, and all bits for one bit position in the word are contained in a single RAM bit slice, the number of RAM cells sharing the read and write circuits is also large.
- The electrical properties resulting from both of these situations cause the access and write cycle times of the RAM to become long, which is undesirable.
- The total number of decoder gates, the number of inputs per gate, and the number of RAM cells per bit slice can all be reduced by employing two decoders with a *coincident selection* scheme.
- In one possible configuration, two *k*/2-input decoders are used instead of one *k*-input decoder.
- One decoder controls the word select lines and the other controls the bit select lines.
- The result is a two-dimensional matrix selection scheme.
- If the RAM chip has *m* words with 1 bit per word, then the scheme selects the RAM cell at the intersection of the Word Select row and the Bit Select column.
- Since the Word Select is no longer strictly selecting words, its name is changed to ***Row Select***.
- An output from the added decoder that selects one or more bit slices is referred to as a ***Column Select***.

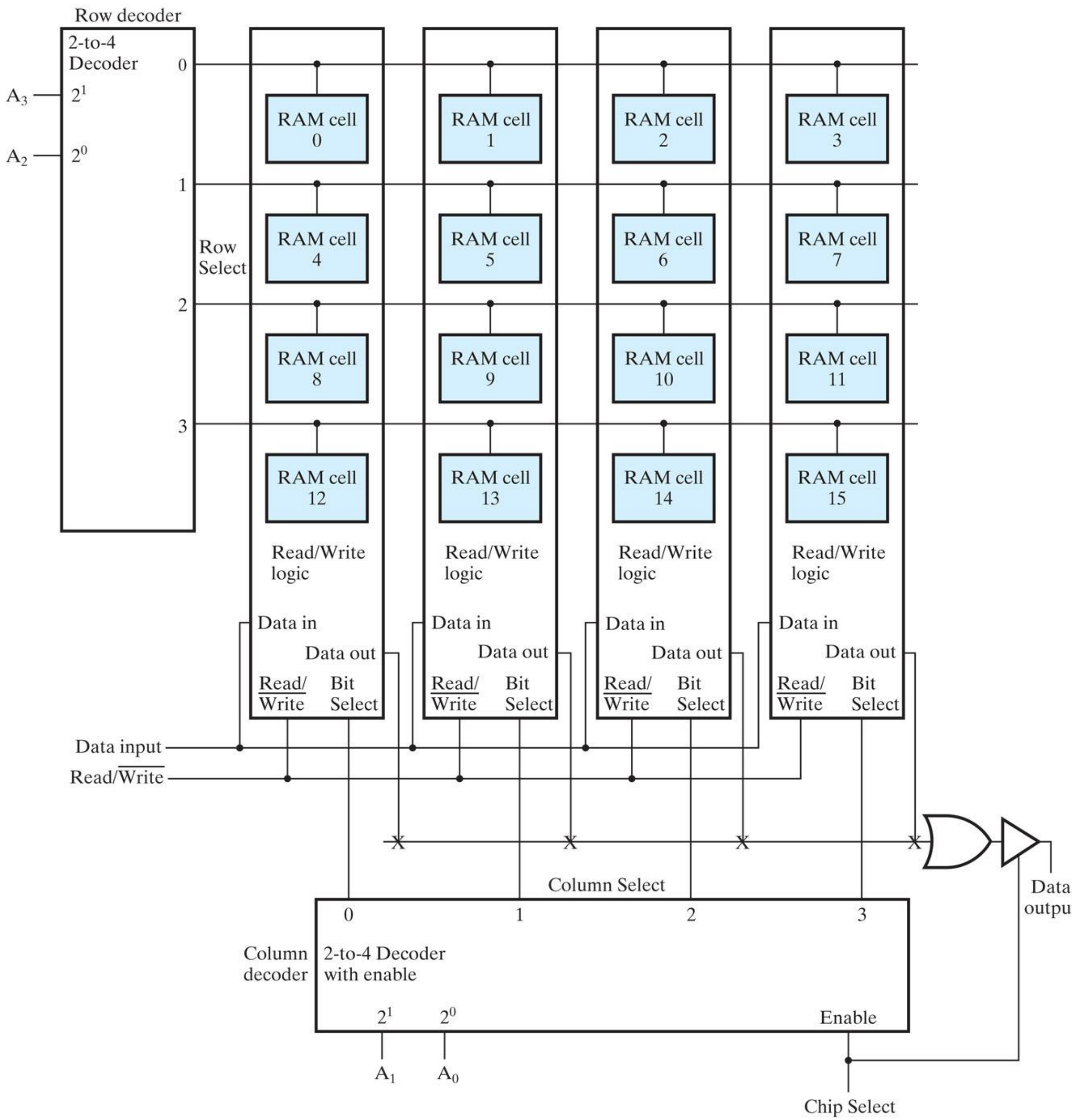
Coincident selection

- Coincident selection is illustrated for the 16×1 RAM chip with the structure shown in Figure.
- The chip consists of four RAM bit slices of four bits each and has a total of 16 RAM cells in a two dimensional array.
    - The two most significant address inputs go through the 2-to-4-line row decoder to select one of the four rows of the array.
    - The two least significant address inputs go through the 2-to-4-line column decoder to select one of the four columns (RAM bit slices) of the array.
- The column decoder is enabled with the Chip Select input. When the Chip Select is 0, all outputs of the decoder are 0 and none of the cells is selected. -> This prevents writing into any RAM cell in the array.
- With Chip Select at 1, a single bit in the RAM is accessed.



Coincident selection

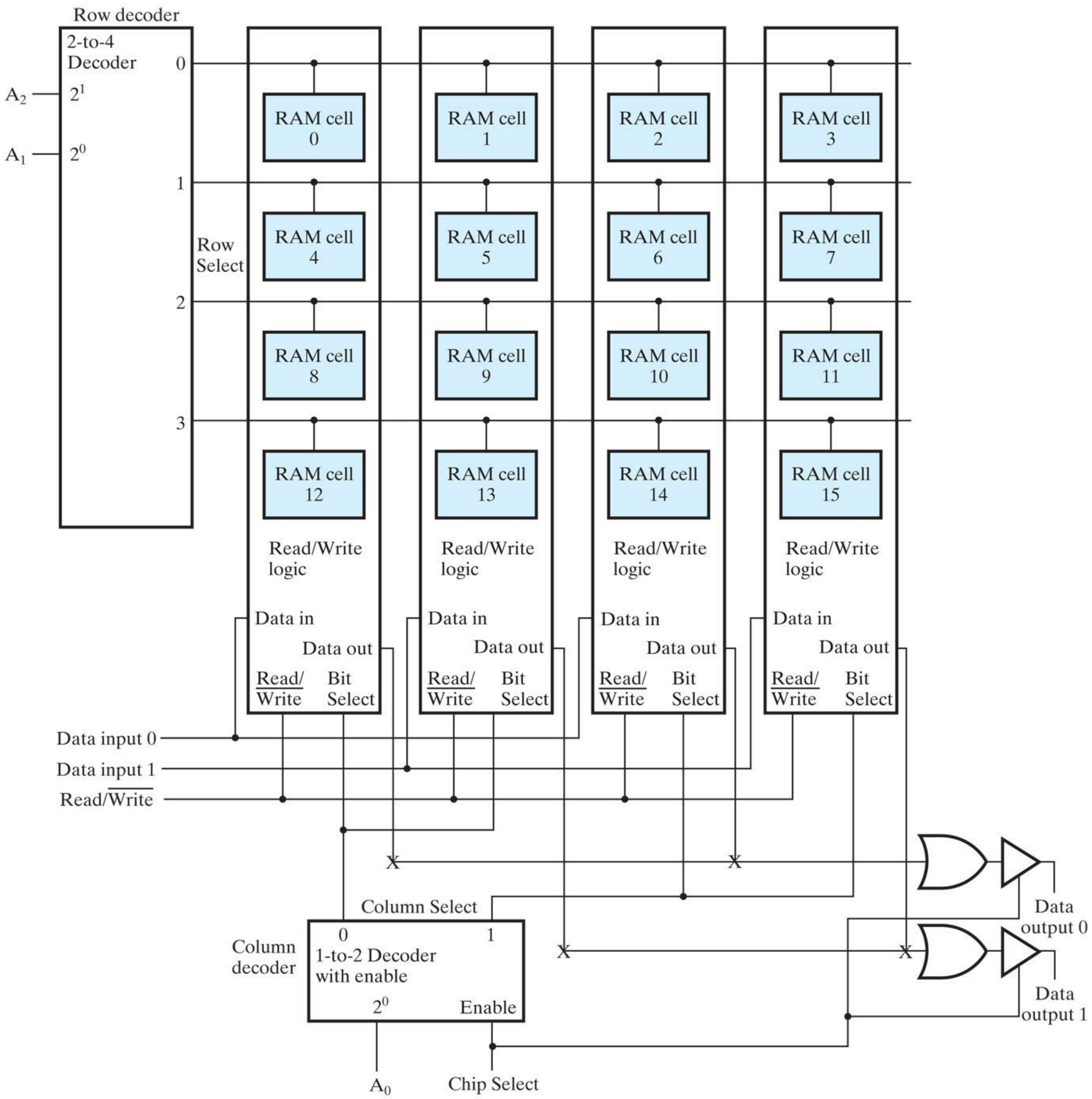
- For example, for the address 1001,
  - the first two address bits are decoded to select row 10 of the RAM cell array.
  - The second two address bits are decoded to select column 01 of the array.
- The RAM cell accessed, in row 2 and column 1 of the array, is cell 9 (10 , 01).
- With a row and column selected, the Read/$\overline{\text{Write}}$ input determines the operation.
- During the read operation (Read/$\overline{\text{Write}}$ = 1), the selected bit of the selected row goes through the OR gate to the three-state buffer.
  - Since the buffer is enabled by Chip Select, the value read appears at the Data Output.
- During the write operation (Read/$\overline{\text{Write}}$ = 0), the bit available on the Data Input line is transferred into the selected RAM cell.
  - Those RAM cells not selected are disabled, and their previous binary values remain unchanged.



Copyright ©2016 Pearson Education, All Rights Reserved

Coincident selection

- The same RAM cell array is used in Figure on right to produce an 8×2 RAM chip (eight words of two bits each).
- The row decoding is unchanged from that in previous Figure; the only changes are in the column and output logic.
- Since there are just three address bits, and two are handled by the row decoder, the column decoder has only one address bit and Chip Select as inputs and produces just two Column Select lines.
- Since two bits at a time are to be written or read, the Column Select lines go to adjacent pairs of RAM bit slices.
- Two input lines, Data Input 0 and Data Input 1, each go to a different bit in all of the pairs.
- Finally, corresponding bits of the pairs share output OR gates and three-state buffers, giving output lines Data Output 0 and Data Output 1.
- The operation of this structure can be illustrated by the application of the address 3 (011).
- The first two bits of the address, 01, access row 1 of the array.
- The final bit, 1, accesses column 1, which consists of bit slices 2 (10) and 3 (11).
- So the word to be written or read lies in RAM cells 6 and 7 (0110 and 0111), which contain bits 0 and 1, respectively, of word 3.



Copyright ©2016 Pearson Education, All Rights Reserved

Coincident selection

- We can demonstrate the savings of the coincident selection scheme by considering a more realistic static RAM size, 32K×8.
- This RAM chip contains a total of 256K bits.
- To make the number of rows and columns in the array equal, we take the square root of 256K, giving 512 = 2^9.
- So the first nine bits of the address are fed to the row decoder and the remaining six bits to the column decoder.
- Without coincident selection, the single decoder would have 15 inputs and 32,768 outputs.
- With coincident selection, there is one 9-to-512-line decoder and one 6-to-64-line decoder.
- The number of gates for a straightforward design of the single decoder would be 32,800.
- For the two coincident decoders, the number of gates is 608, reducing the gate count by a factor of more than 50.
- In addition, although it appears that there are 64 times as many Read/Write circuits, the column selection can be done between the RAM cells and the Read/Write circuits, so that only the original eight circuits are required.
- Because of the reduced number of RAM cells attached to each Read/Write circuit at any time, the access time of the chip is also improved.