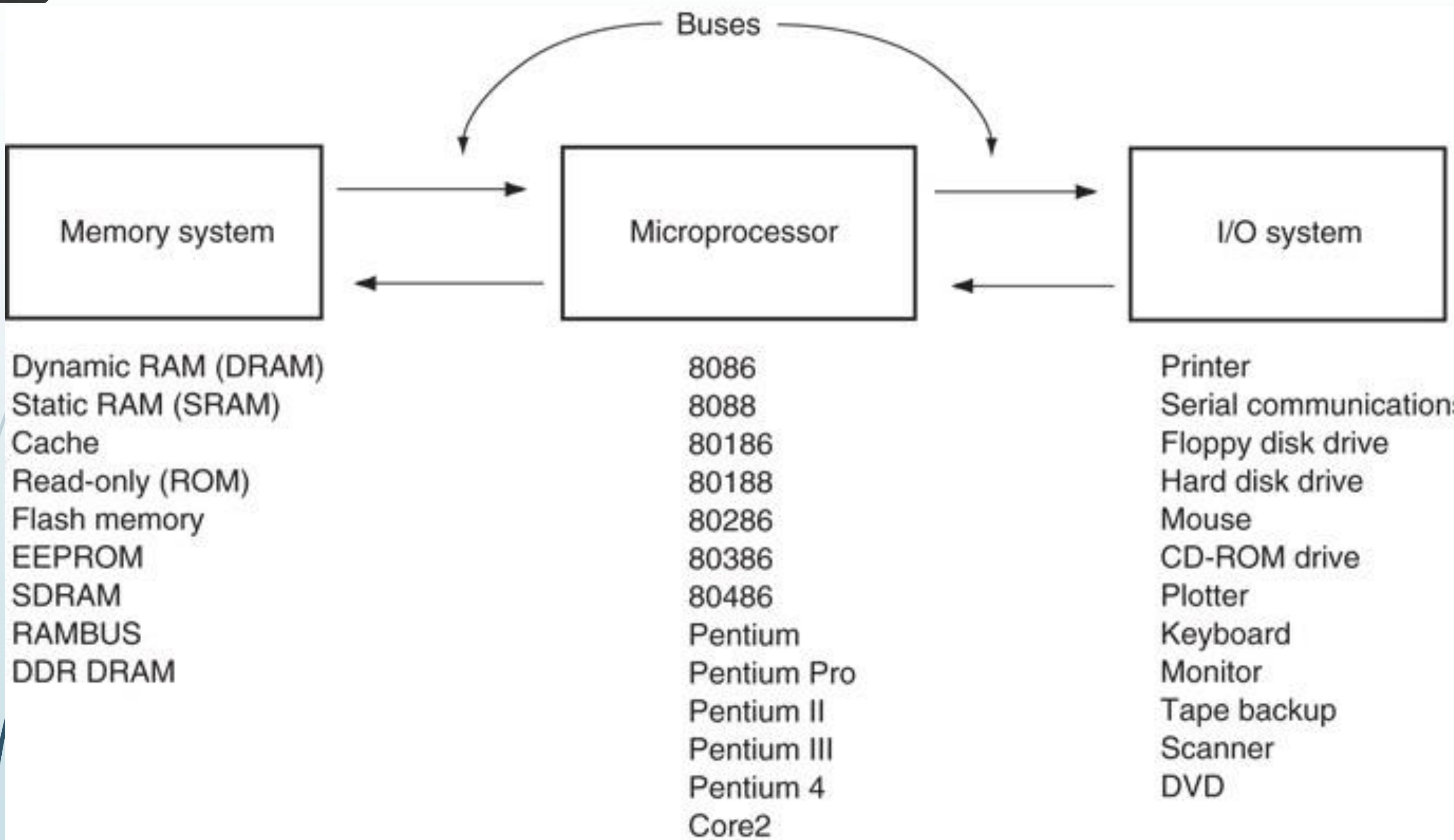
2. THE MICROPROCESSOR-BASED PERSONAL COMPUTER SYSTEM

- Computers have undergone many changes recently.
- Machines that once filled large areas reduced to small desktop computer systems because of the microprocessor.
  - although compact, they possess computing power only dreamed of a few years ago
- Figure 2-1 shows block diagram of the personal computer.
- Applies to any computer system, from early mainframe computers to the latest systems.
- Diagram composed of three blocks interconnected by buses.
  - a **bus** is the set of common connections that carry the same type of information

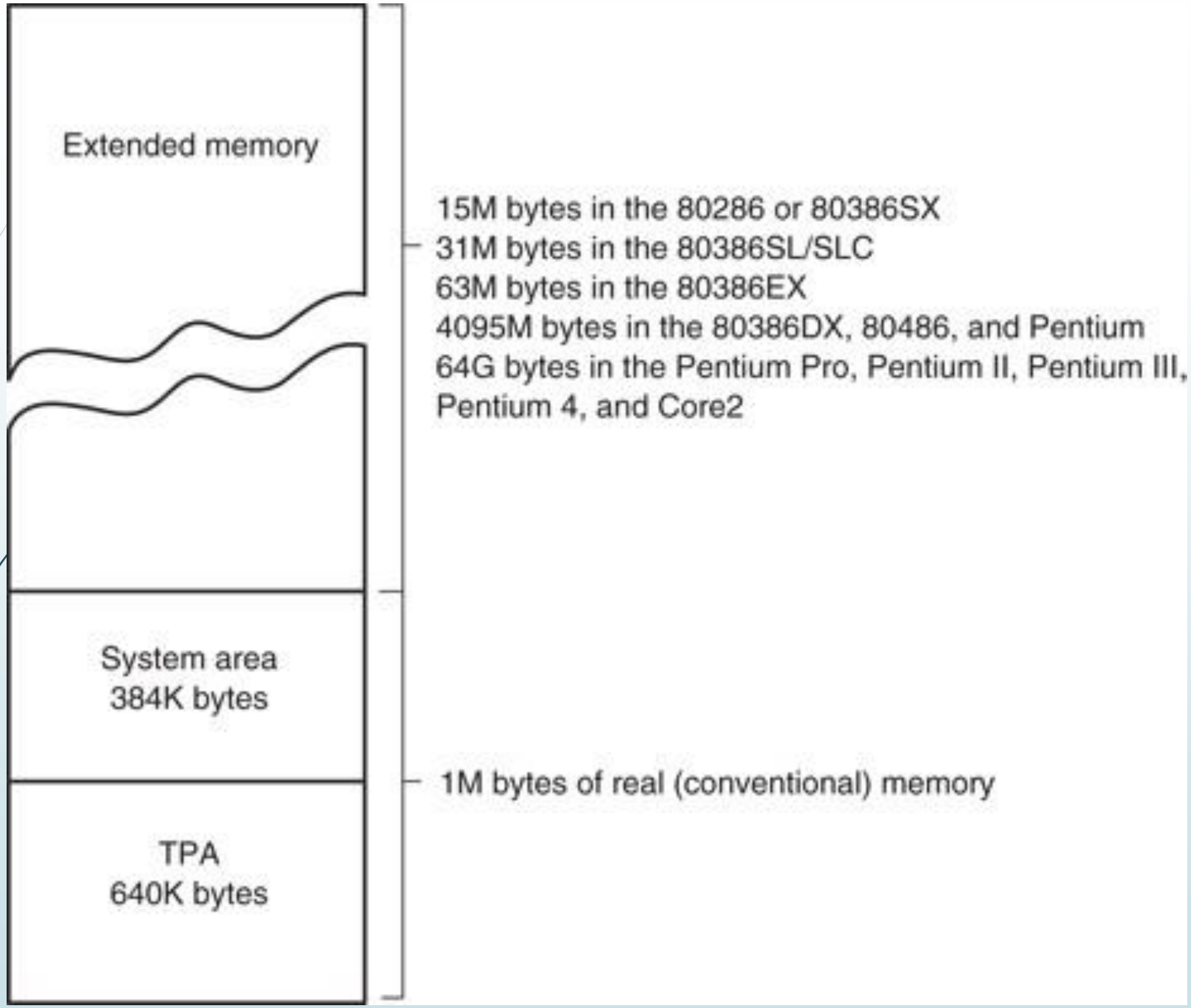
Figure 2-1 The block diagram of a microprocessor-based computer system.



The Memory and I/O System

- Memory structure of all Intel-based personal computers similar.
- Figure 2–2 illustrates memory map of a personal computer system.
- This map applies to any IBM personal computer.
  - also any IBM-compatible clones in existence

Figure 2–2 The memory map of a personal computer.



- Main memory system divided into three parts:
  - TPA (transient program area)
  - System area
  - XMS (extended memory system)
- Type of microprocessor present determines whether an extended memory system exists.
- First 1M byte of memory often called the real or conventional memory system.
  - Intel microprocessors designed to function in this area using real mode operation

- 80286 through the Core2 contain the TPA (640K bytes) and system area (384K bytes).
  - also contain extended memory
  - often called AT class machines
- The PS/l and PS/2 by IBM are other versions of the same basic memory design.
- Also referred to as ISA (industry standard architecture) or EISA (extended ISA).
- The PS/2 referred to as a micro-channel architecture or ISA system.
  - depending on the model number

- Pentium and ATX class machines feature addition of the PCI (**peripheral component interconnect**) bus.
  - now used in all Pentium through Core2 systems
- Extended memory up to 15M bytes in the 80286 and 80386SX; 4095M bytes in 80486 80386DX, Pentium microprocessors.
- The Pentium Pro through Core2 computer systems have up to 1M less than 4G (32bit address)or 1M less than 64G (36 Bit address) of extended memory.
- Servers tend to use the larger memory map.

- Many 80486 systems use **VESA** local, VL bus to interface disk and video to the microprocessor at the local bus level.
  - allows 32-bit interfaces to function at same clocking speed as the microprocessor
  - recent modification supporting 64-bit data bus has generated little interest
- ISA/EISA standards function at 8 MHz.
- PCI bus is a 32- or 64-bit bus.
  - specifically designed to function with the Pentium through Core2 at a bus speed of 33 MHz.

- **Three** newer buses have appeared.
- USB (**universal serial bus**).
  - intended to connect peripheral devices to the microprocessor through a serial data path and a twisted pair of wires
- Data transfer rates are 10 Mbps for USB1.
- Increase to 480 Mbps in USB2.
- Increase to 480X10 Mbps in USB3.

- AGP (**advanced graphics port**) for video cards.
- The port transfers data between video card and microprocessor at higher speeds.
  - 66 MHz, with 64-bit data path
- Latest AGP speed 8X or 2G bytes/second.
  - video subsystem change made to accommodate new DVD players for the PC.

- Latest new buses are serial ATA interface (**SATA: Serial Advanced Technology Attachment)** for hard disk drives; PCI Express bus (Peripheral Component Interface) for the video card.
- The SATA bus transfers data from PC to hard disk at rates of 150M bytes per second; 300M bytes for SATA-2.
  - serial ATA standard will eventually reach speeds of 450M bytes per second
- PCI Express bus video cards operate at 16X speeds today.

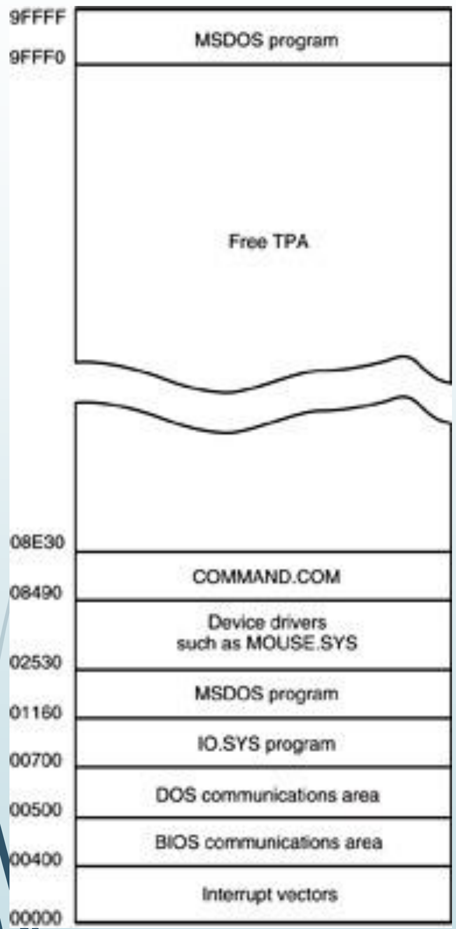
SATA

- **Serial ATA** (**SATA** or **Serial Advanced Technology Attachment**) is a computer bus interface for connecting host bus adapters to mass storage devices such as hard disk drives and optical drives.
- Serial ATA was designed to replace the older ATA (AT Attachment) standard (also known as EIDE), offering several advantages over the older parallel ATA (PATA) interface: reduced cable-bulk and cost (7 conductors versus 40), native hot swapping, faster data transfer through higher signalling rates, and more efficient transfer through an (optional) I/O queuing protocol.

The TPA

- The transient program area (TPA) holds the DOS (**disk operating system**) operating system; other programs that control the computer system.
  - the TPA is a DOS concept and not applicable in Windows
  - also stores any currently active or inactive DOS application programs
  - length of the TPA is 640K bytes

Figure 2–4 The memory map of the TPA in a personal computer. (Note that this map will vary between systems.)



- DOS memory map shows how areas of TPA are used for system programs, data and drivers.
  - also shows a large area of memory available for application programs
  - hexadecimal number to left of each area represents the memory addresses that begin and end each data area

- Hexadecimal memory addresses number each byte of the memory system.
  - a hexadecimal number is a number represented in radix 16 or base 16
  - each digit represents a value from 0 to 9 and from A to F
- Often a hexadecimal number ends with an H to indicate it is a hexadecimal value.
  - 1234H is 1234 hexadecimal
  - also represent hexadecimal data as 0x1234 for a 1234 hexadecimal
- Interrupt vectors access DOS, BIOS (basic I/O system), and applications.
- Areas contain transient data to access I/O devices and internal features of the system.
  - these are stored in the TPA so they can be changed as DOS operates

- The IO.SYS loads into the TPA from the disk whenever an MSDOS system is started.
- IO.SYS contains programs that allow DOS to use keyboard, video display, printer, and other I/O devices often found in computers.
- The IO.SYS program links DOS to the programs stored on the system BIOS ROM.

- **Drivers** are programs that control installable I/O devices.
  - mouse, disk cache, hand scanner, CD-ROM memory (**Compact Disk Read-Only Memory**), DVD (**Digital Versatile Disk; Digital Video Disk**), or installable devices, as well as programs
- Installable drivers control or drive devices or programs added to the computer system.
- DOS drivers normally have an extension of .SYS; MOUSE.SYS.
- DOS version 3.2 and later files have an extension of .EXE; EMM386.EXE.

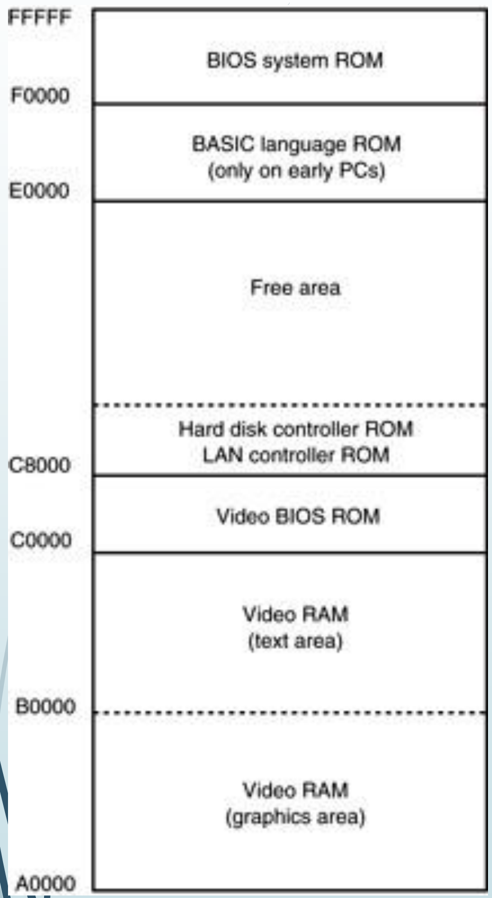
- Though not used by Windows, still used to execute DOS applications, even with Win XP.
- Windows uses a file called SYSTEM.INI to load drivers used by Windows.
- Newer versions of Windows have a registry added to contain information about the system and the drivers used.
- You can view the registry with the REGEDIT program.

- COMMAND.COM (**command processor**) controls operation of the computer from the keyboard when operated in the DOS mode.
- COMMAND.COM processes DOS commands as they are typed from the keyboard.
- If COMMAND.COM is erased, the computer cannot be used from the keyboard in DOS mode.
  - never erase COMMAND.COM, IO.SYS, or MSDOS.SYS to make room for other software
  - your computer will not function

The System Area

- Smaller than the TPA; just as important.
- The **system area** contains programs on read-only (ROM) or flash memory, and areas of read/write (RAM) memory for data storage.
- Figure 2–5 shows the system area of a typical personal computer system.
- As with the map of the TPA, this map also includes the hexadecimal memory addresses of the various areas.

Figure 2–5 The system area of a typical personal computer.



- First area of system space contains video display RAM and video control programs on ROM or flash memory.
  - area starts at location A0000H and extends to C7FFFH
  - size/amount of memory depends on type of video display adapter attached

- Display adapters generally have video RAM at A0000H–AFFFFH.
  - stores graphical or bit-mapped data
- Memory at B0000H–BFFFFH stores text data.
- The video BIOS on a ROM or flash memory, is at locations C0000H–C7FFFH.
  - contains programs to control DOS video display
- C8000H–DFFFFH is often open or free.
  - used for expanded memory system (EMS) in PC or XT system; upper memory system in an AT

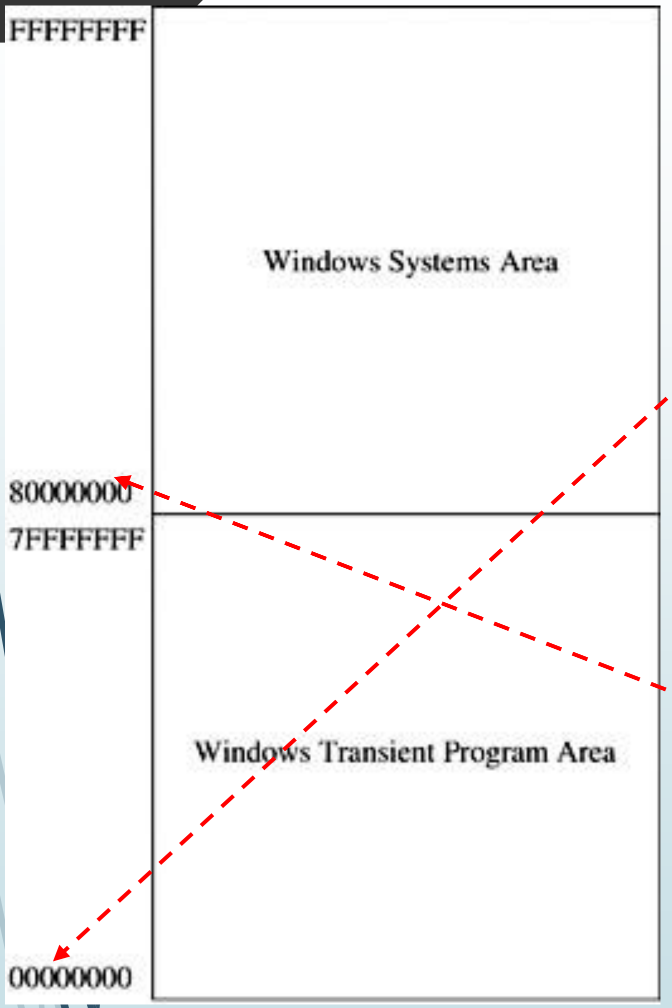
- Expanded memory system allows a 64K-byte page frame of memory for use by applications.
  - page frame (D0000H - DFFFFH) used to expand memory system by switching in pages of memory from EMS into this range of memory addresses
- Locations E0000H–EFFFFH contain cassette BASIC on ROM found in early IBM systems.
  - often open or free in newer computer systems
- Video system has its own BIOS ROM at location C0000H.

- System BIOS ROM is located in the top 64K bytes of the system area (F0000H–FFFFFH).
  - controls operation of basic I/O devices connected to the computer system
  - does not control operation of video
- The first part of the system BIOS (F0000H–F7FFFH) often contains programs that set up the computer.
- Second part contains procedures that control the basic I/O system.

Windows Systems

- Modern computers use a different memory map with Windows than DOS memory maps.
- The Windows memory map in Figure 2–6 has two main areas; a TPA and system area.
- The difference between it and the DOS memory map are sizes and locations of these areas.

Figure 2–6 The memory map used by Windows XP.



- TPA is first 2G bytes from locations 00000000H to 7FFFFFFFH.
- Every Windows program can use up to 2G bytes of memory located at linear addresses 00000000H through 7FFFFFFFH.
- System area is last 2G bytes from 80000000H to FFFFFFFFH.

The Microprocessor

- Called the CPU (**central processing unit**).
- The controlling element in a computer system.
- Controls memory and I/O through connections called buses.
  - buses select an I/O or memory device, transfer data between I/O devices or memory and the microprocessor, control I/O and memory systems
- Memory and I/O controlled via instructions stored in memory, executed by the microprocessor.

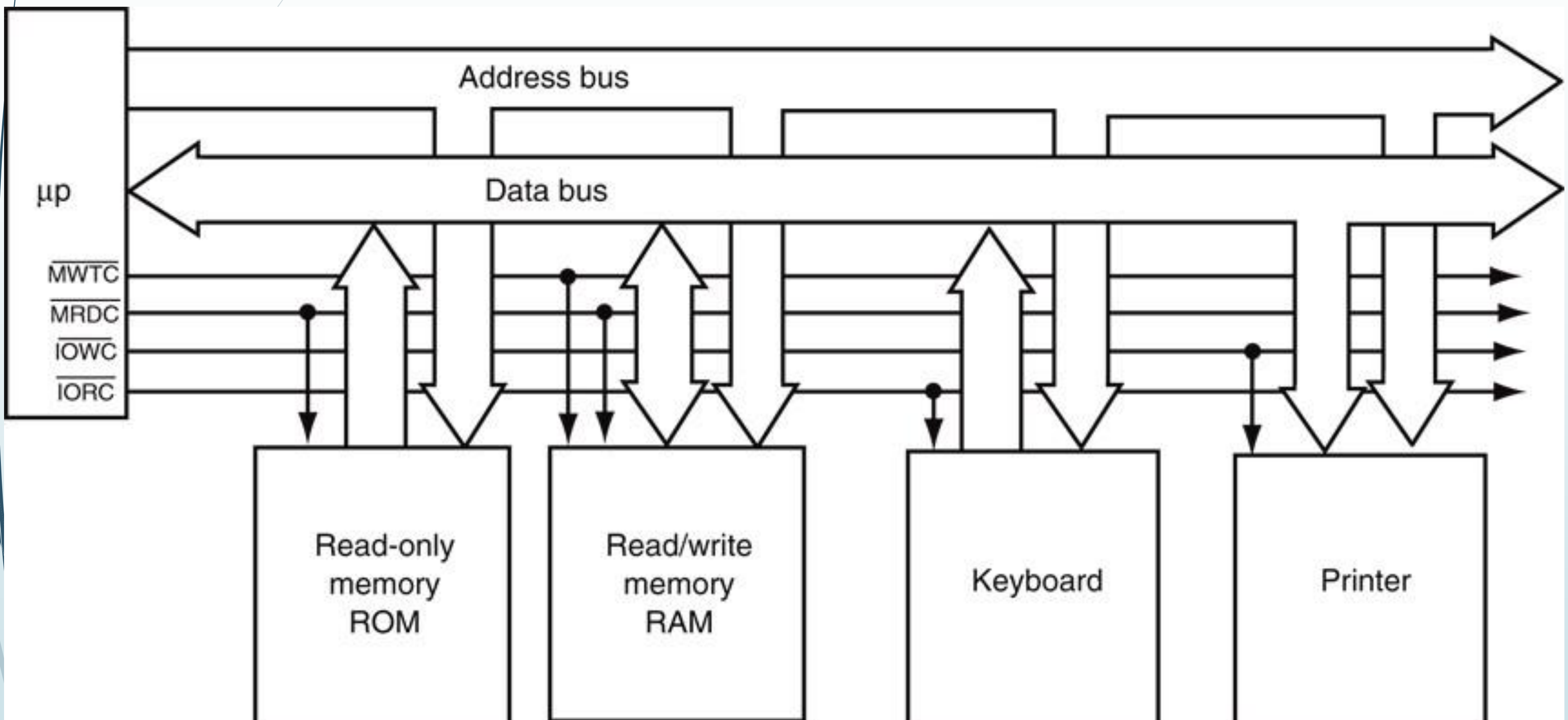
- Microprocessor performs three main tasks:
  - data transfer between itself and the memory or I/O systems
  - simple arithmetic and logic operations →processing
  - program flow via simple decisions
- Power of the microprocessor is capability to execute billions of millions of instructions per second from a program or software (**group of instructions**) stored in the memory system.
  - stored programs make the microprocessor and computer system very powerful devices

- Another powerful feature is the ability to make simple decisions based upon numerical facts.
  - a microprocessor can decide if a number is zero, positive, and so forth
- These decisions allow the microprocessor to modify the program flow, so programs appear to think through these simple decisions.

Buses

- A common group of wires that interconnect components in a computer system.
- Transfer address, data, & control information between microprocessor, memory and I/O.
- Three buses exist for this transfer of information: address, data, and control.
- Figure 2–7 shows how these buses interconnect various system components.

Figure 2–7 The block diagram of a computer system showing the address, data, and control bus structure.



- The address bus requests a memory location from the memory or an I/O location from the I/O devices.
  - if I/O is addressed, the address bus contains a 16-bit I/O address from 0000H through FFFFH.
  - if memory is addressed, the bus contains a memory address, varying in width by type of microprocessor.
- 64-bit extensions to Pentium provide 40 address pins, allowing up to 1T (2^{40} .=.10^{12}) byte of memory to be accessed.

- The data bus transfers information between the microprocessor and its memory and I/O address space.
- Data transfers vary in size, from 8 bits wide to 64 bits wide in various Intel microprocessors.
  - 8088 has an 8-bit data bus that transfers 8 bits of data at a time
  - 8086, 80286, 80386SL, 80386SX, and 80386EX transfer 16 bits of data
  - 80386DX, 80486SX, and 80486DX, 32 bits
  - Pentium through Core2 microprocessors transfer 64 bits of data

- Advantage of a wider data bus is speed in applications using wide data.
- Figure 2–8 shows memory widths and sizes of 8086 through Core2 microprocessors.
- In all Intel microprocessors family members, memory is numbered by byte.
- Pentium through Core2 microprocessors contain a 64-bit-wide data bus.

Figure 2–8a The physical memory systems of the 8086 through the Core2 microprocessors.

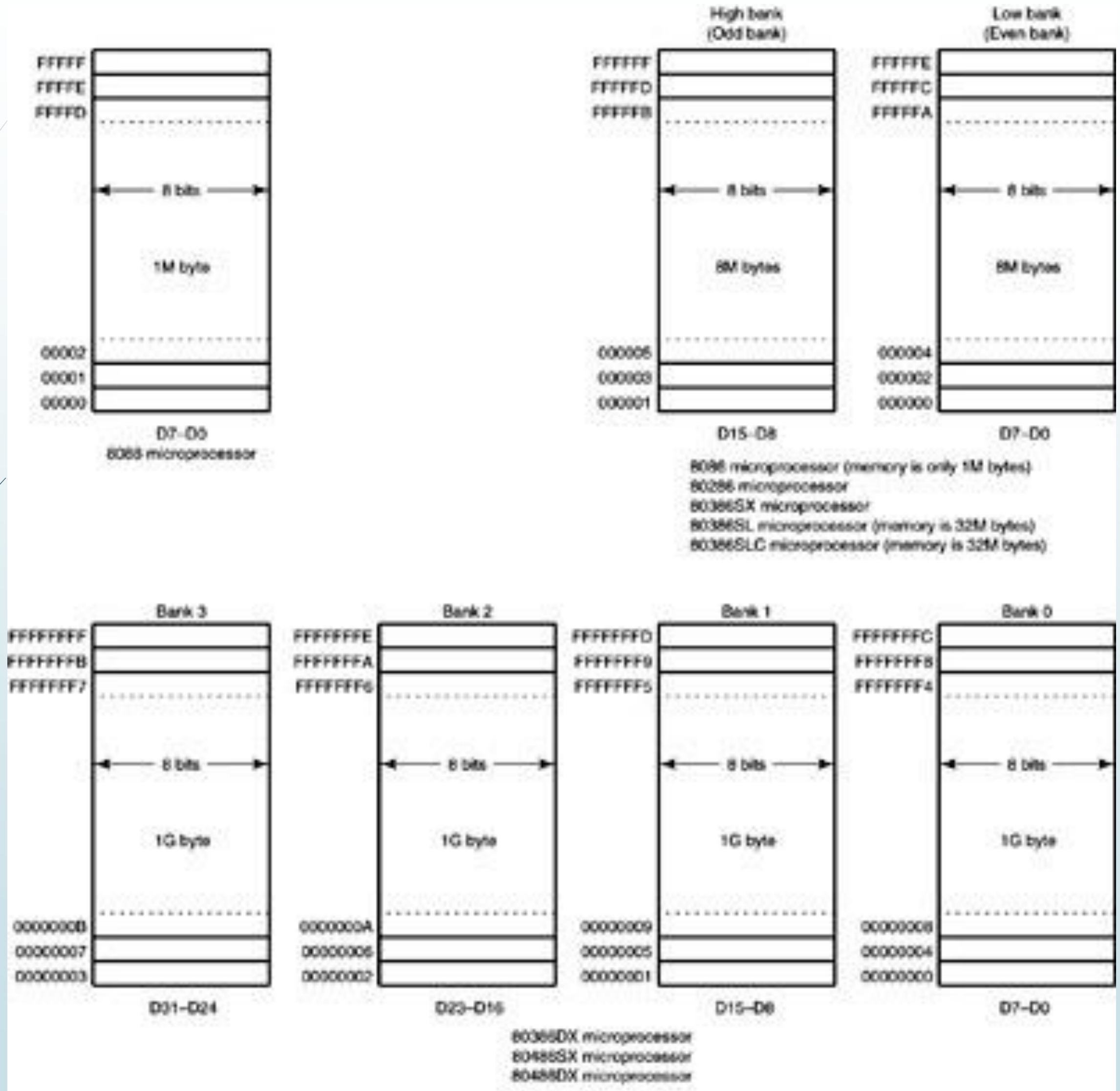
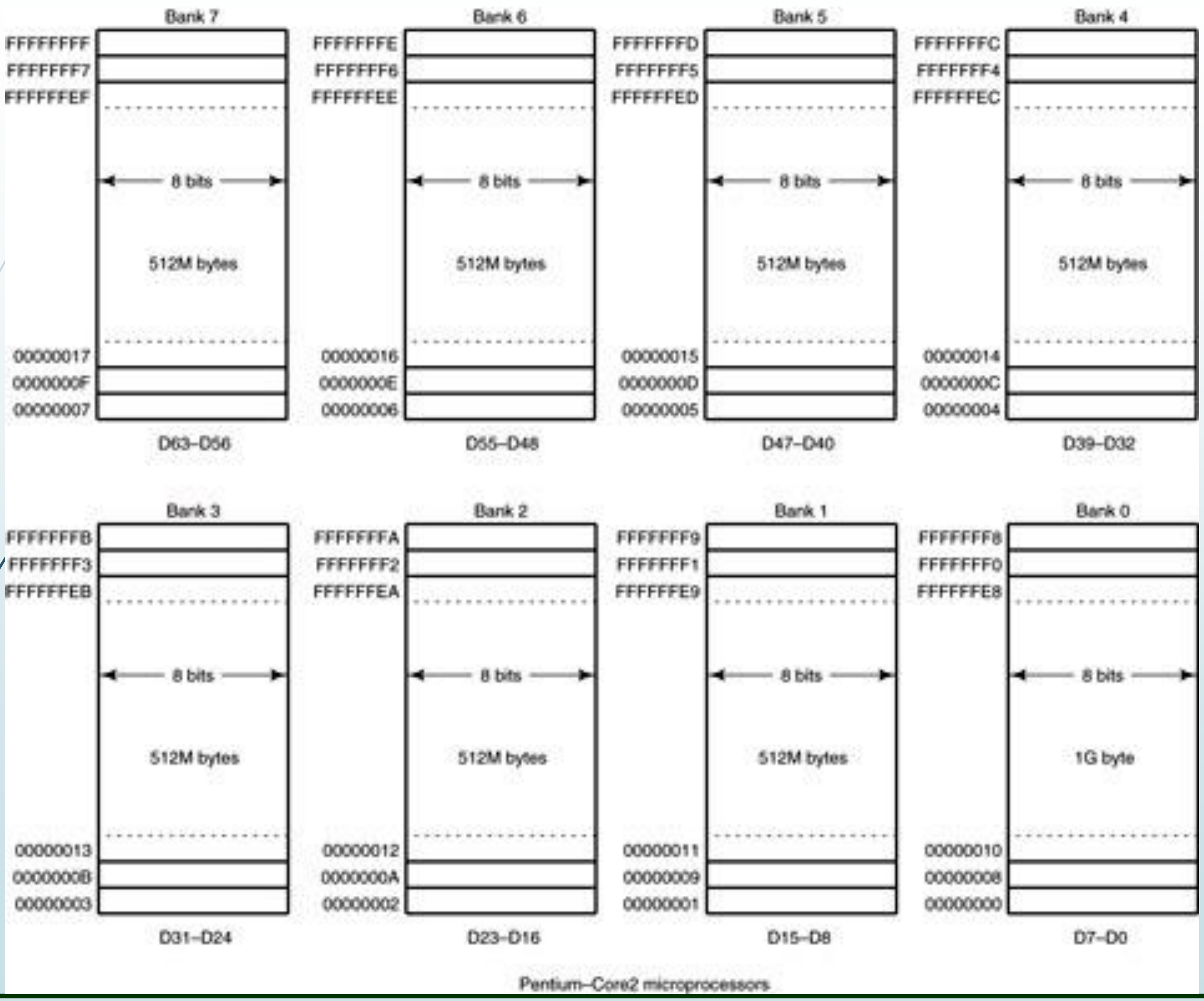


Figure 2–8b The physical memory systems of the 8086 through the Core2 microprocessors.



- Control bus lines select and cause memory or I/O to perform a read or write operation.
- In most computer systems, there are four control bus connections:
- $\overline{MRDC}$ (**memory read control**)
- $\overline{MWTC}$ (**memory write control**)
- $\overline{IORC}$ (**I/O read control**)
- $\overline{IOWC}$ (**I/O write control**).
- overbar indicates the control signal is active-low; (active when logic zero appears on control line)

- The microprocessor reads a memory location by sending the memory an address through the address bus.
- Next, it sends a memory read control signal to cause the memory to read data.
- Data read from memory are passed to the microprocessor through the data bus.
- Whenever a memory write, I/O write, or I/O read occurs, the same sequence ensues.

3 NUMBER SYSTEMS

- Use of a microprocessor requires working knowledge of numbering systems.
  - binary, decimal, and hexadecimal
- This section provides a background for these numbering systems.
- Conversions are described.
  - decimal and binary
  - decimal and hexadecimal
  - binary and hexadecimal

Digits

- Before converting numbers between bases, digits of a number system must be understood.
- First digit in any numbering system is always zero.
- A decimal (base 10) number is constructed with 10 digits: 0 through 9.
- A base 8 (**octal**) number; 8 digits: 0 through 7.
- A base 2 (**binary**) number; 2 digits: 0 and 1.

- If the base exceeds 10, additional digits use letters of the alphabet, beginning with an A.
  - a base 12 number contains 10 digits: 0 through 9, followed by A for 10 and B for 11
- Note that a base 10 number does contain a *10* digit.
  - a base 8 number does not contain an *8* digit
- Common systems used with computers are decimal, binary, and hexadecimal (base 16).
  - many years ago octal numbers were popular

Positional Notation

- Once digits are understood, larger numbers are constructed using positional notation.
  - position to the left of the units position is the tens position
  - left of tens is the hundreds position, and so forth
- An example is decimal number 132.
  - this number has 1 hundred, 3 tens, and 2 units
- Exponential powers of positions are critical for understanding numbers in other systems.

- Exponential value of each position:
  - the units position has a weight of 10^0, or 1
  - tens position a weight of 10^1, or 10
  - hundreds position has a weight of 10^2, or 100
- Position to the left of the radix (**number base**) point is always the units position in system.
  - called a decimal point only in the decimal system
  - position to left of the binary point always 2^0, or 1
  - position left of the octal point is 8^0, or 1
- Any number raised to its zero power is always one (1), or the units position.

- Position to the left of the units position always the number base raised to the first power.
    - in a decimal system, this is 10^1, or 10
    - binary system, it is 2^1, or 2
    - 11 decimal has a different value from 11 binary
- 11 decimal has different value from 11 binary.
    - decimal number composed of 1 ten, plus 1 unit; a value of 11 units
    - binary number 11 is composed of 1 two, plus 1 unit: a value of 3 decimal units
    - 11 octal has a value of 9 decimal units

- In the decimal system, positions right of the decimal point have negative powers.
  - first digit to the right of the decimal point has a value of 10^{-1}, or 0.1.
- In the binary system, the first digit to the right of the binary point has a value of 2^{-1}, or 0.5.
- Principles applying to decimal numbers also generally apply to those in any other system.
- To convert a binary number to decimal, add weights of each digit to form its decimal equivalent.

Conversion to Decimal

- To convert from any number base to decimal, determine the weights or values of each position of the number.
- Sum the weights to form the decimal equivalent.

Conversion from Decimal

- Conversions from decimal to other number systems more difficult to accomplish.
- To convert the whole number portion of a decimal number, divide by 1 radix.
- To convert the fractional portion, multiply by the radix.

Whole Number Conversion from Decimal

- To convert a decimal whole number to another number system, divide by the radix and save remainders as significant digits of the result.
- An algorithm for this conversion:
  - divide the decimal number by the radix
    (number base)
  - save the remainder
    (first remainder is the least significant digit)
  - repeat steps 1 and 2 until the quotient is zero

- To convert 10 decimal to binary, divide it by 2.
  - the result is 5, with a remainder of 0
- First remainder is units position of the result.
  - in this example, a 0
- Next, divide the 5 by 2; result is 2, with a remainder of 1.
  - the 1 is the value of the twos (2^1) position
- Continue division until the quotient is a zero.
- The result is written as 1010_2 from the bottom to the top.

- To convert 10 decimal to base 8, divide by 8.
  - a 10 decimal is a 12 octal.
- For decimal to hexadecimal, divide by 16.
  - remainders will range in value from 0 through 15
  - any remainder of 10 through 15 is converted to letters A through F for the hexadecimal number
  - decimal number 109 converts to a 6DH

Converting from a Decimal Fraction

- Conversion is accomplished with multiplication by the radix.
- Whole number portion of result is saved as a significant digit of the result.
  - fractional remainder again multiplied by the radix
  - when the fraction remainder is zero, multiplication ends
- Some numbers are never-ending (repetend).
  - a zero is never a remainder

- Algorithm for conversion from a decimal fraction:
  - multiply the decimal fraction by the radix (number base).
  - save the whole number portion of the result (even if zero) as a digit; first result is written immediately to the right of the radix point
  - repeat steps 1 and 2, using the fractional part of step 2 until the fractional part of step 2 is zero
- Same technique converts a decimal fraction into any number base.

Binary-Coded Hexadecimal

- **Binary-coded hexadecimal** (BCH) is a hexadecimal number written each digit is represented by a 4-bit binary number.
- BCH code allows a binary version of a hexadecimal number to be written in a form easily converted between BCH and hexadecimal.
- Hexadecimal represented by converting digits to BCH code with a space between each digit.

4. COMPUTER DATA FORMATS

- Successful programming requires a precise understanding of data formats.
- Commonly, data appear as ASCII, Unicode, BCD, signed and unsigned integers, and floating-point numbers (real numbers).
- Other forms are available but are not commonly found.

ASCII and Unicode Data

- ASCII (**American Standard Code for Information Interchange**) data represent alphanumeric characters in computer memory.
- Standard ASCII code is a 7-bit code.
  - eighth and most significant bit used to hold parity
- If used with a printer, most significant bits are 0 for alphanumeric printing; 1 for graphics.
- In PC, an extended ASCII character set is selected by placing 1 in the leftmost bit.

- Extended ASCII characters store:
  - some foreign letters and punctuation
  - Greek & mathematical characters
  - box-drawing & other special characters
- Extended characters can vary from one printer to another.
- ASCII control characters perform control functions in a computer system.
  - clear screen, backspace, line feed, etc.
- Enter control codes through the keyboard.
  - hold the Control key while typing a letter

- Many Windows-based applications use the **Unicode** system to store alphanumeric data.
  - stores each character as 16-bit data
- Codes 0000H–00FFH are the same as standard ASCII code.
- Remaining codes, 0100H–FFFFH, store all special characters from many character sets.
- Allows software for Windows to be used in many countries around the world.
- For complete information on Unicode, visit: *http://www.unicode.org*

BCD (Binary-Coded Decimal) Data

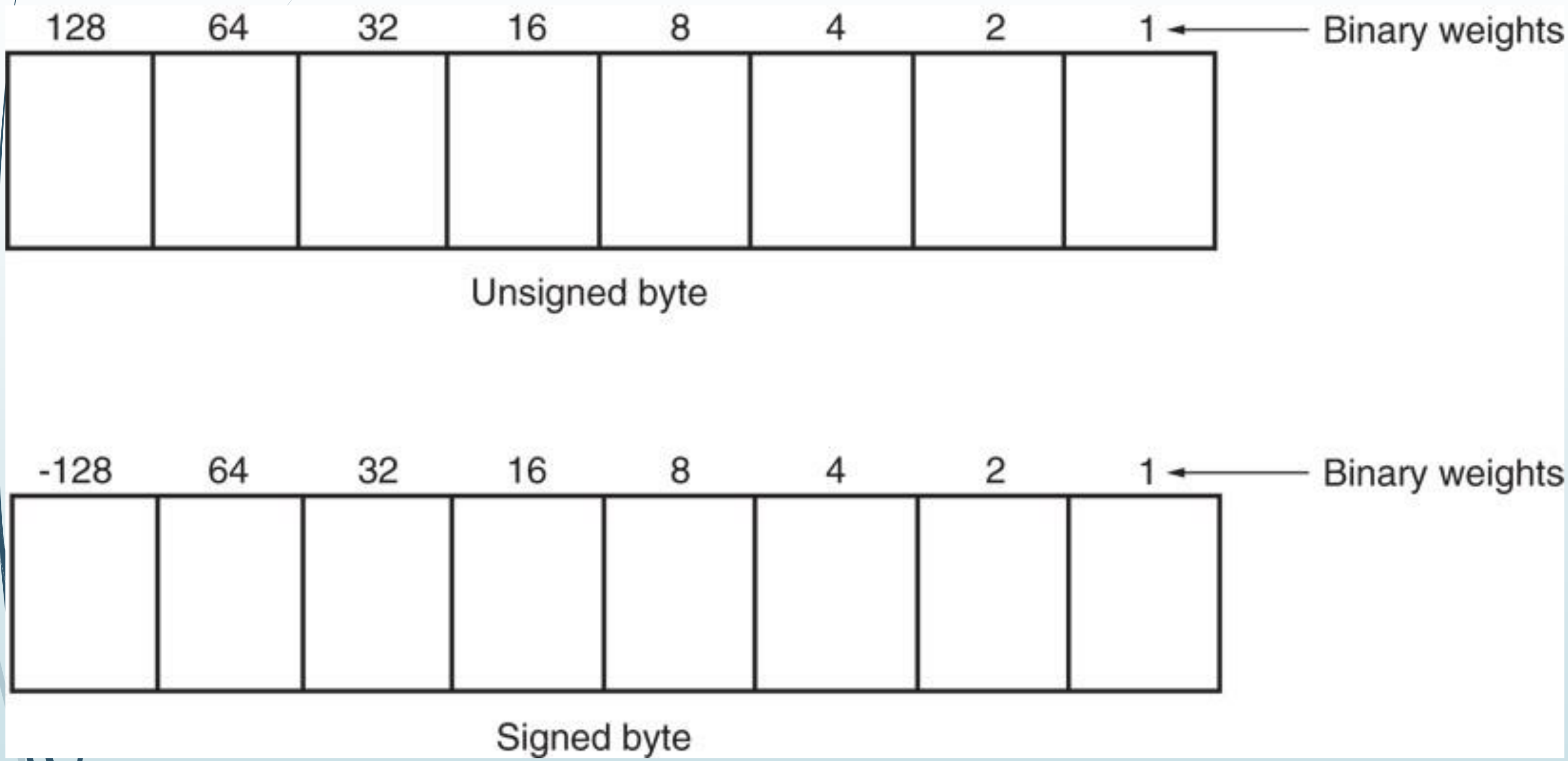
- The range of a BCD digit extends from 0000_2 to 1001_2, or 0–9 decimal, stored in two forms:
- Stored in packed form:
  - packed BCD data stored as two digits per byte;
  - used for BCD addition and subtraction in the instruction set of the microprocessor
- Stored in unpacked form:
  - unpacked BCD data stored as one digit per byte
  - returned from a keypad or keyboard

- Applications requiring BCD data are point-of-sales terminals.
  - also devices that perform a minimal amount of simple arithmetic
- If a system requires complex arithmetic, BCD data are seldom used.
  - there is no simple and efficient method of performing complex BCD arithmetic

Byte-Sized Data

- Stored as *unsigned* and *signed* integers.
- Difference in these forms is the weight of the leftmost bit position.
  - value 128 for the unsigned integer
  - *minus* 128 for the signed integer
- In signed integer format, the leftmost bit represents the sign bit of the number.
  - also a weight of *minus* 128

Figure 2–9 The unsigned and signed bytes illustrating the weights of each binary-bit position.

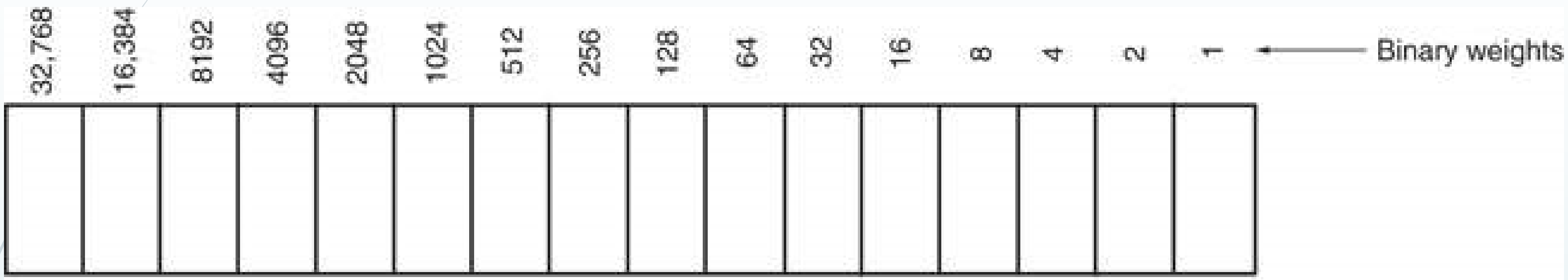


- Unsigned integers range 00H to FFH (0–255)
- Signed integers from –128 to 0 to + 127.
- Negative signed numbers represented in this way are stored in the two's complement form.
- Evaluating a signed number by using weights of each bit position is much easier than the act of two's complementing a number to find its value.
  - especially true in the world of calculators designed for programmers

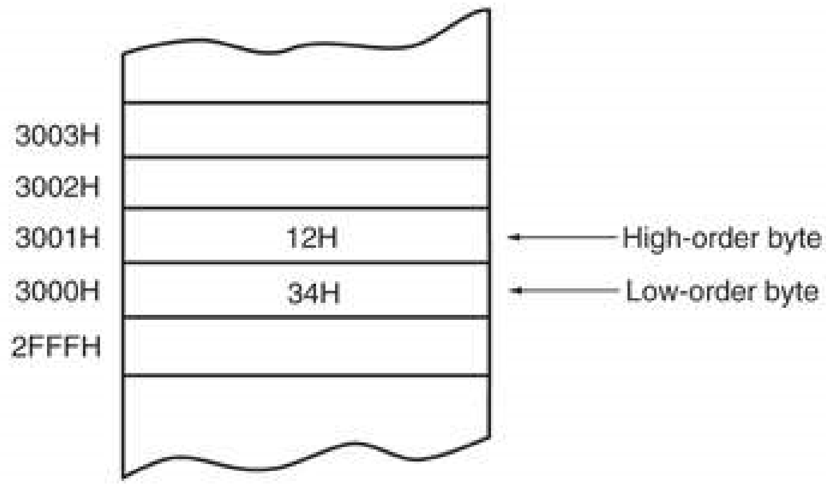
Word-Sized Data

- A word (16-bits) is formed with two bytes of data.
- The least significant byte always stored in the lowest-numbered memory location.
- Most significant byte is stored in the highest.
- This method of storing a number is called the **little endian** format.

Figure 2–10 The storage format for a 16-bit word in (a) a register and (b) two bytes of memory.

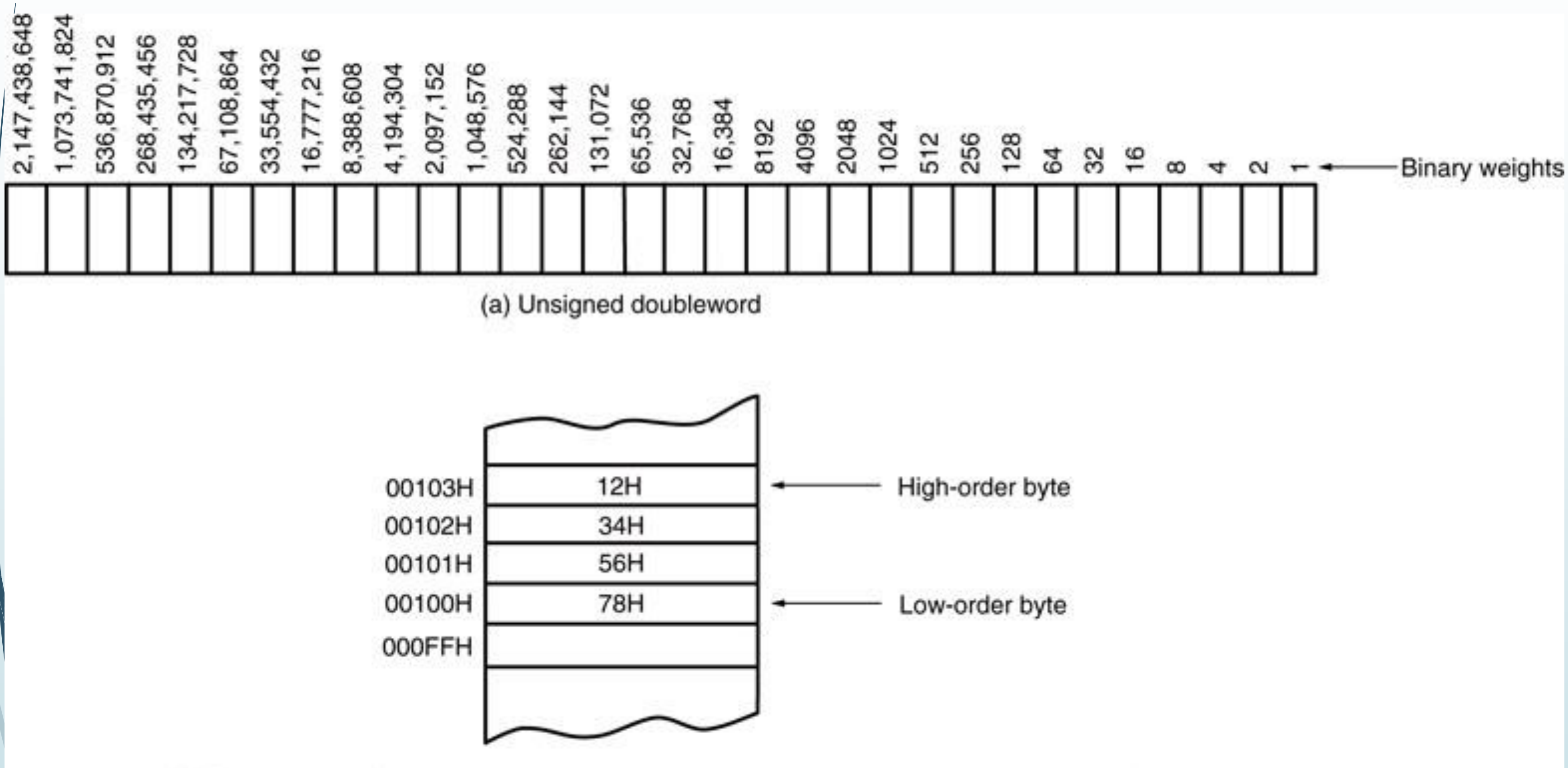


(a) Unsigned word



(b) The contents of memory location 3000H and 3001H are the word 1234H.

Figure 2–11 The storage format for a 32-bit word in (a) a register and (b) 4 bytes of memory.



(b) The contents of memory location 00100H–00103H are the doubleword 12345678H.

- Alternate method is called the **big endian** format.
- Numbers are stored with the lowest location containing the most significant data.
- Not used with Intel microprocessors.
- The big endian format is used with the Motorola family of microprocessors.

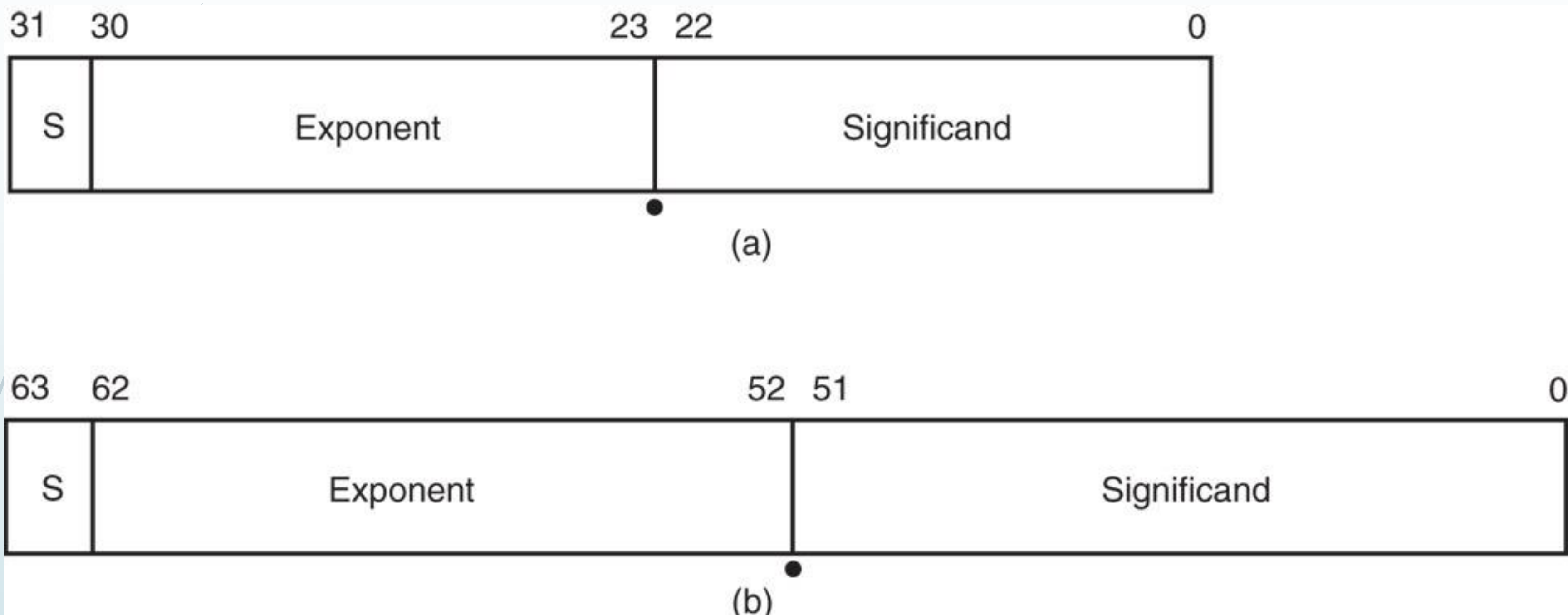
Doubleword-Sized Data

- **Doubleword-sized data** requires four bytes of memory because it is a 32-bit number.
  - appears as a product after a multiplication
  - also as a dividend before a division
- Define using the assembler directive **define doubleword(s)**, or **DD.**
  - also use the DWORD directive in place of DD

Real Numbers

- Since many high-level languages use Intel microprocessors, real numbers are often encountered.
- A real, or a **floating-point number** contains two parts:
  - a mantissa, significant, or fraction
  - an exponent.
- A 4-byte number is called **single-precision.**
- The 8-byte form is called **double-precision.**

Figure 2–12 The floating-point numbers in (a) single-precision using a bias of 7FH and (b) double-precision using a bias of 3FFH.



- The assembler can be used to define real numbers in single- & double-precision forms:
  - use the DD directive for single-precision 32-bit numbers
  - use **define quadword(s)**, or DQ to define 64-bit double-precision real numbers
- Optional directives are REAL4, REAL8, and REAL10.
  - for defining single-, double-, and extended precision real numbers

SUMMARY

- Mechanical computer age began with the advent of the abacus in 500 B.C.
- This first mechanical calculator remained unchanged until 1642, when Blaise Pascal improved it.
- An early mechanical computer system was the Analytical Engine developed by Charles Babbage in 1823.

SUMMARY

cont'd..

- The first electronic calculating machine was developed during World War II by Konrad Zuse, an early pioneer of digital electronics.
- The Z3 was used in aircraft and missile design for the German war effort.
- The first electronic computer, which used vacuum tubes, was placed into operation in 1943 to break secret German military codes.

SUMMARY

cont'd...

- The first electronic computer system, the Colossus, was invented by Alan Turing.
- Its only problem was that the program was fixed and could not be changed.
- The first general-purpose, programmable electronic computer system was developed in 1946 at the University of Pennsylvania.
- This first modern computer was called the ENIAC (Electronics Numerical Integrator and Calculator).

SUMMARY

cont'd...

- The first high-level programming language, called FLOWMATIC.
- Developed for the UNIVAC I computer by Grace Hopper in the early 1950s.
- This led to FORTRAN and other early programming languages such as COBOL.

SUMMARY

cont'd...

- The world's first microprocessor, the Intel 4004, was a 4-bit microprocessor-a programmable controller on a chip-that was meager by today's standards.
- It addressed a mere 4096 4-bit memory locations.
- Its instruction set contained only 45 different instructions.

SUMMARY

cont'd...

- Microprocessors that are common today include the 8086/8088, which were the first 16-bit microprocessors.
- Following these early 16-bit machines were the 80286, 80386, 80486, Pentium, Pentium Pro, Pentium II, Pentium III, Pentium 4, and Core2 processors.
- The architecture has changed from 16 bits to 32 bits and, with the Itanium, to 64 bits.

SUMMARY

cont'd...

- With each newer version, improvements followed that increased the processor's speed and performance.
- From all indications, this process of speed and performance improvement will continue.
- Performance increases may not always come from an increased clock frequency.

SUMMARY

cont'd...

- DOS-based personal computers contain memory systems that include three main areas: TPA (transient program area), system area, and extended memory.
- The TPA hold application programs, the operating system, and drivers.
- The system area contains memory used for video display cards, disk drives, and the BIOS ROM.

SUMMARY

cont'd...

- The extended memory area is only available to the 80286 through the Core2 microprocessor in an AT-style or ATX-style personal computer system.
- The Windows-based personal computers contain memory systems that include two main areas: TPA and systems area.

SUMMARY

cont'd...

- The 8086/8088 address 1M byte of memory from locations 00000H-FFFFFH.
- The 80286 and 80386SX address 16M bytes of memory from 000000H-FFFFFFH.
- The 80386SL addresses 32M bytes of memory from 0000000H-1FFFFFFH.
- The 80386DX through the Core2 address 4G bytes of memory from locations 00000000H-FFFFFFFFH.

SUMMARY

cont'd...

- Pentium Pro through the Core2 can operate with a 36-bit address and access up to 64G bytes of memory from locations 000000000H- FFFFFFFFFH.
- A Pentium 4 or Core2 operating with 64-bit extensions addresses memory from locations 0000000000H- FFFFFFFFFFH for 1T byte of memory.

SUMMARY

cont'd...

- All versions of the 8086 through the Core2 microprocessors address 64K bytes of I/O address space.
- These I/O ports are numbered from 0000H to FFFFH with I/O ports 0000H-03FFH reserved for use by the personal computer system.
- The PCI bus allows ports 0400H-FFFFH.

SUMMARY

cont'd...

- The operating system in early personal computers was either MSDOS (Microsoft disk operating system) or PCDOS (personal computer disk operating system from IBM).
- The operating system performs the task of operating or controlling the computer system, along with its I/O devices.
- Modern computers use Microsoft Windows in place of DOS as an operating system.

SUMMARY

cont'd...

- The microprocessor is the controlling element in a computer system.
- The micro-processor performs data transfers, does simple arithmetic and logic operations, and makes simple decisions.
- The microprocessor executes programs stored in the memory system to perform complex operations in short periods of time.

SUMMARY

cont'd...

- All computer systems contain three buses to control memory and I/O.
- The address bus is used to request a memory location or I/O device.
- The data bus transfers data between the microprocessor and its memory and I/O spaces.
- The control bus controls the memory and I/O, and requests reading or writing of data.

SUMMARY

cont'd...

- Numbers are converted from any number base to decimal by noting the weights of each position.
- The weight of the position to the left of the radix point is always the units position in any number system.
- The position to the left of the units position is always the radix times one.
- Succeeding positions are determined by multiplying by the radix.

SUMMARY

cont'd...

- The weight of the position to the right of the radix point is always deter-mined by dividing by the radix.
- Conversion from a whole decimal number to any other base is accomplished by dividing by the radix.
- Conversion from a fractional decimal number is accomplished by multiplying by the radix.

SUMMARY

cont'd…

- Hexadecimal data are represented in hexadecimal form or in a code called binary-coded hexadecimal (BCH).
- A binary-coded hexadecimal number is one that is written with a 4-bit binary number that represents each hexadecimal digit.
- The ASCII code is used to store alphabetic or numeric data.

SUMMARY

cont'd...

- The ASCII code is a 7-bit code; it can have an eighth bit that is used to extend the character set from 128 codes to 256 codes.
- The carriage return (Enter) code returns the print head or cursor to the left margin.
- The line feed code moves the cursor or print head down one line.
- Most modern applications use Unicode, which contains ASCII at codes 0000H-00FFH.

SUMMARY

cont'd...

- Binary-coded decimal (BCD) data are sometimes used in a computer system to store decimal data.
- These data are stored either in packed (two digits per byte) or unpacked (one digit per byte) form.
- Binary data are stored as a byte (8 bits), word (16 bits), or doubleword (32 bits) in a computer system.
- These data may be unsigned or signed.

SUMMARY

cont'd...

- Signed negative data are always stored in the two's complement form.
- Data that are wider than 8 bits are always stored using the little-endian format.
- In 32-bit Visual C++ these data are represented with char (8 bits), short (16 bits) and int (32 bits).

SUMMARY

- Floating-point data are used in computer systems to store whole, mixed, and fractional numbers.
- A floating-point number is composed of a sign, a mantissa, and an exponent.
- The assembler directives DB or BYTE define bytes, DW or WORD define words, DD or DWORD define doublewords, and DQ or QWORD define quadwords.