MODULE I

Inside PC

Syllabus

Working of PC- Motherboard- Foam factors-Components of Motherboard- Bus Architecture- Chipsets- Expansion Slots- Memory slots and Cache- CPU and Processor socket- BIOS and POST- CMOS and CMOS Battery- Purpose and characteristics of Processors- Characteristics of Memory- types of memory- different types of DRAM verities of RAM- Memory packaging

Working of PC

A **personal computer (PC)** is a computing device made up of many distinct electronic components that all function together in order to accomplish some useful task, such as adding up the numbers in a spreadsheet or helping you write a letter. Figure shows the parts of a computer

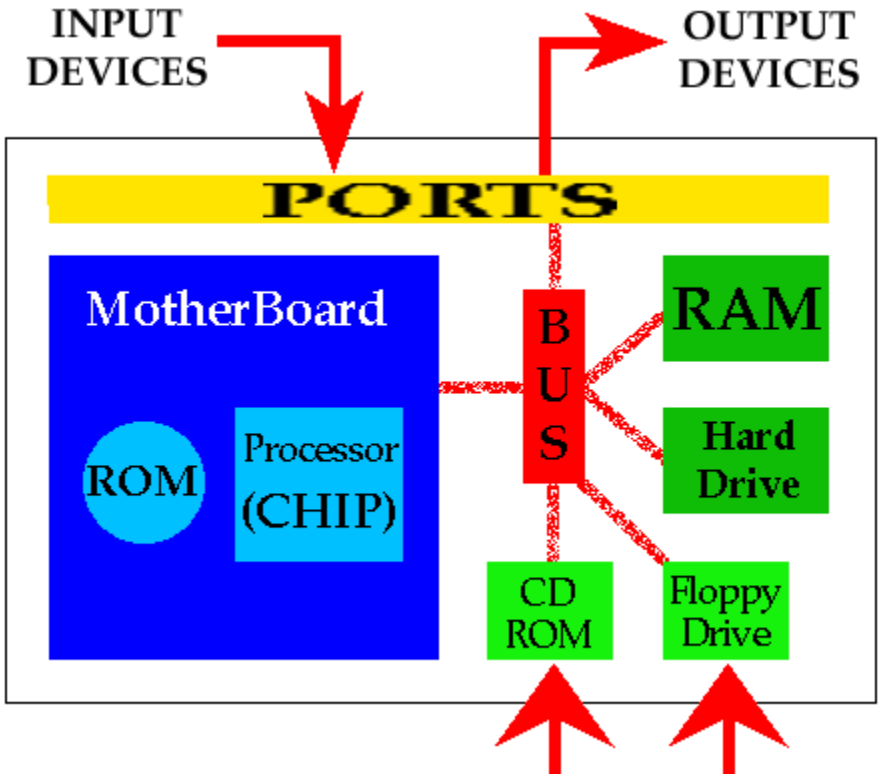


Figure 0: Inside The Computer

- The Central Processing Unit:
  - (CPU),
  - Buses,
  - Ports and controllers,
  - ROM;
- Main Memory (RAM);
- Input Devices;

- Output Devices;
- Secondary Storage;
    - floppy disks,
    - hard disk,
    - CD-ROM

The Central Processing Unit (CPU)

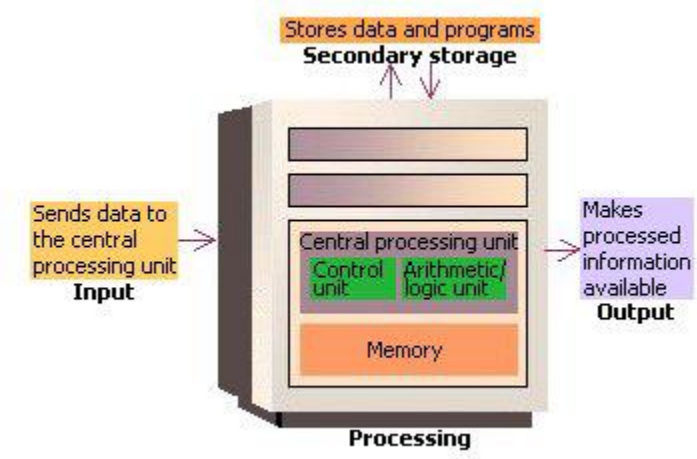


Figure 1: The Central Processing Unit

The computer does its primary work in a part of the machine we cannot see, a control center that converts data input to information output. This control center, called the central processing unit (CPU), is a highly complex, extensive set of electronic circuitry that executes stored program instructions. All computers, large and small, must have a central processing unit. As Figure 1 shows, the central processing unit consists of two parts: The control unit and the arithmetic/logic unit. Each part has a specific function.

Computers use two types of storage: Primary storage and secondary storage. The CPU interacts closely with primary storage, or main memory, referring to it for both instructions and data. Technically, memory is not part of the CPU.

A computer's memory holds data only temporarily, at the time the computer is executing a program. Secondary storage holds permanent or semi-permanent data on some external magnetic or optical medium. The diskettes and CD-ROM disks that you have seen with personal computers are secondary storage devices, as are hard disks.

The Control Unit

The control unit of the CPU contains circuitry that uses electrical signals to direct the entire computer system to carry out, or execute, stored program instructions. The control unit must communicate with both the arithmetic/logic unit and memory.

The Arithmetic/Logic Unit

The arithmetic/logic unit (ALU) contains the electronic circuitry that executes all arithmetic and logical operations. The arithmetic/logic unit can perform four kinds of arithmetic operations, or mathematical calculations: addition, subtraction, multiplication, and division. As its name implies, the arithmetic/logic unit also performs logical operations. A logical operation is usually a comparison. The unit can compare numbers, letters, or special characters. The computer can then take action based on the result of the comparison. This is a very important capability. It is by comparing that a computer is able to tell, for instance, whether there are unfilled seats on airplanes, whether charge- card customers have exceeded their credit limits, and whether one candidate for Congress has more votes than another.

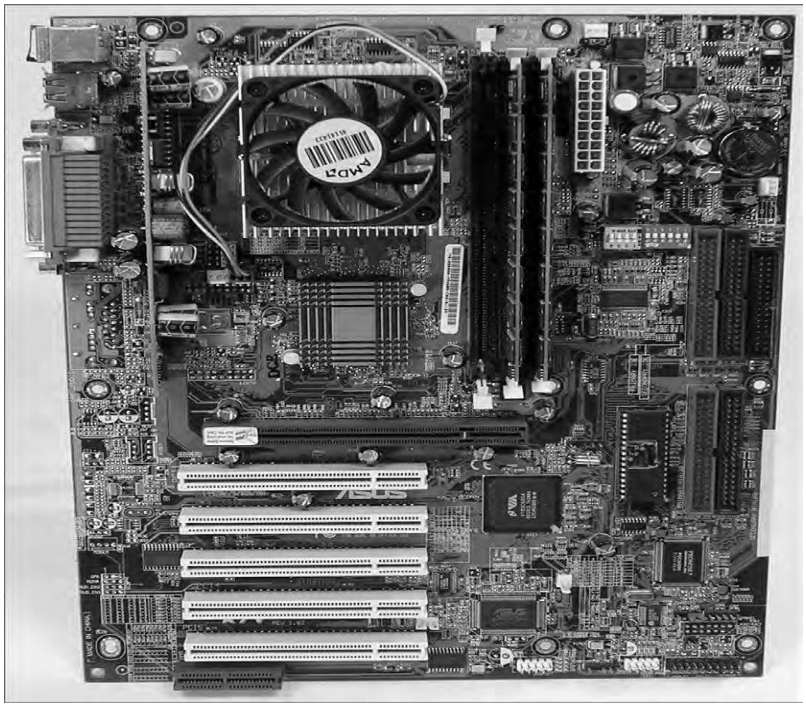
Motherboard

A motherboard (sometimes alternatively known as the mainboard, system board or mobo) is the main printed circuit board (PCB) found in computers and other expandable systems. It holds many of the crucial electronic components of the system, such as the central processing unit (CPU) and memory, and provides connectors for other peripherals. The motherboard contains the buses, or electrical pathways, found in a computer. These buses allow data to travel between the various components that comprise a computer.

The motherboard accommodates the central processing unit (CPU), RAM, expansion slots, heat sink/fan assembly, BIOS chip, chip set, and the embedded wires that interconnect the motherboard components. Sockets, internal and external connectors, and various ports are also placed on the motherboard

Components of Motherboards

The spine of the computer is the *motherboard*, otherwise known as the system board and mainboard. This is the printed circuit board (PCB)—a conductive series of pathways laminated to a nonconductive substrate—that lines the bottom of the computer and is often of a uniform color, such as olive, brown, or blue. It is the most important component in the computer because it connects all the other components together. Figure 1.1 shows a typical PC system board, as seen from above. All other components are attached to this circuit board. On the system board, you will find the central processing unit (CPU), underlying circuitry, expansion slots, video components, random access memory (RAM) slots, and a variety of other chips.



F i g u r e 1.1 A typical system board

Motherboards Form Factor

The **form factor** of motherboards pertains to the size and shape of the board. It also describes the physical layout of the different components and devices on the motherboard. The form factor affects where individual components go and the shape of the computer's case. There are several specific form factors that most PC motherboards use so that they can all fit in standard cases.

Various Form Factors Exist For Motherboards

- System Board FormAT – Advanced Technology
- Baby AT-Advanced Technology
- ATX – Advanced Technology Extended
- Mini-ATX – Smaller footprint of ATX
- Micro-ATX – Smaller footprint of ATX
- Flex-ATX – Smaller footprint of Micro -ATX
- LPX – Low-profile Extended
- NLX – New Low-profile Extended
- BTX – Balanced Technology Extended

i) Advanced Technology Extended

The Advanced Technology Extended (*ATX*) motherboard was developed by Intel in the mid-1990s to improve upon the classic AT-style motherboard architecture that had ruled the PC world for many years. The ATX motherboard has the processor and memory slots at right angles to the expansion cards. This arrangement puts the processor and memory in line with the fan output of the power supply, allowing the processor to run cooler. And because those components are not in line with the expansion cards, you can install full-length expansion cards—adapters that extend the full length of the inside of a standard computer case—in an ATX motherboard machine. ATX (and its derivatives) are the primary motherboards in use today. Standard ATX motherboards measure 12g n 9.6g (305 n 244 mm).

ii) Micro ATX

One form factor that is designed to work in standard ATX cases, as well as its own smaller cases, is known as *micro ATX* (also referred to as µATX). Micro ATX follows the ATX principle of component placement for enhanced cooling over pre-ATX designs but with a smaller footprint. With this smaller form come some trade-offs. For the compact use of space, you must give up quantity: quantity of memory slots, motherboard headers, expansion slots, integrated components. You also have fewer micro ATX chassis

bays, although the same small-scale motherboard can fit into much larger cases if your original peripherals are still a requirement.

Be aware that micro ATX systems tend to be designed with power supplies of lower wattage in order to help keep down power consumption and heat production. This is generally acceptable with the standard, reduced micro ATX suite of components. As more off-board USB ports are added and larger cases are used with additional in-case peripherals, a larger power supply might be required.

Micro ATX motherboards share their width, mounting hole pattern, and rear interface pattern with ATX motherboards but are shallower and square, measuring 9.6 g n 9.6 g (244 n 244 mm). They were designed to be able to fit into full-size ATX cases.

iii) ITX

The *ITX* line of motherboard form factors was developed by VIA as a low-power, small form factor (SFF) board for specialty uses, such as home-theater systems and as embedded components. ITX itself is not an actual form factor but a family of form factors. The family consists of the following form factors:

Mini-ITX—6.7 g n 6.7 g (170 n 170 mm)

Nano-ITX—4.7 g n 4.7 g (120 n 120 mm)

Pico-ITX—3.9g n 2.8g (100 n 72 mm)

Mobile-ITX—2.4g n 2.4g (60 n 60 mm)

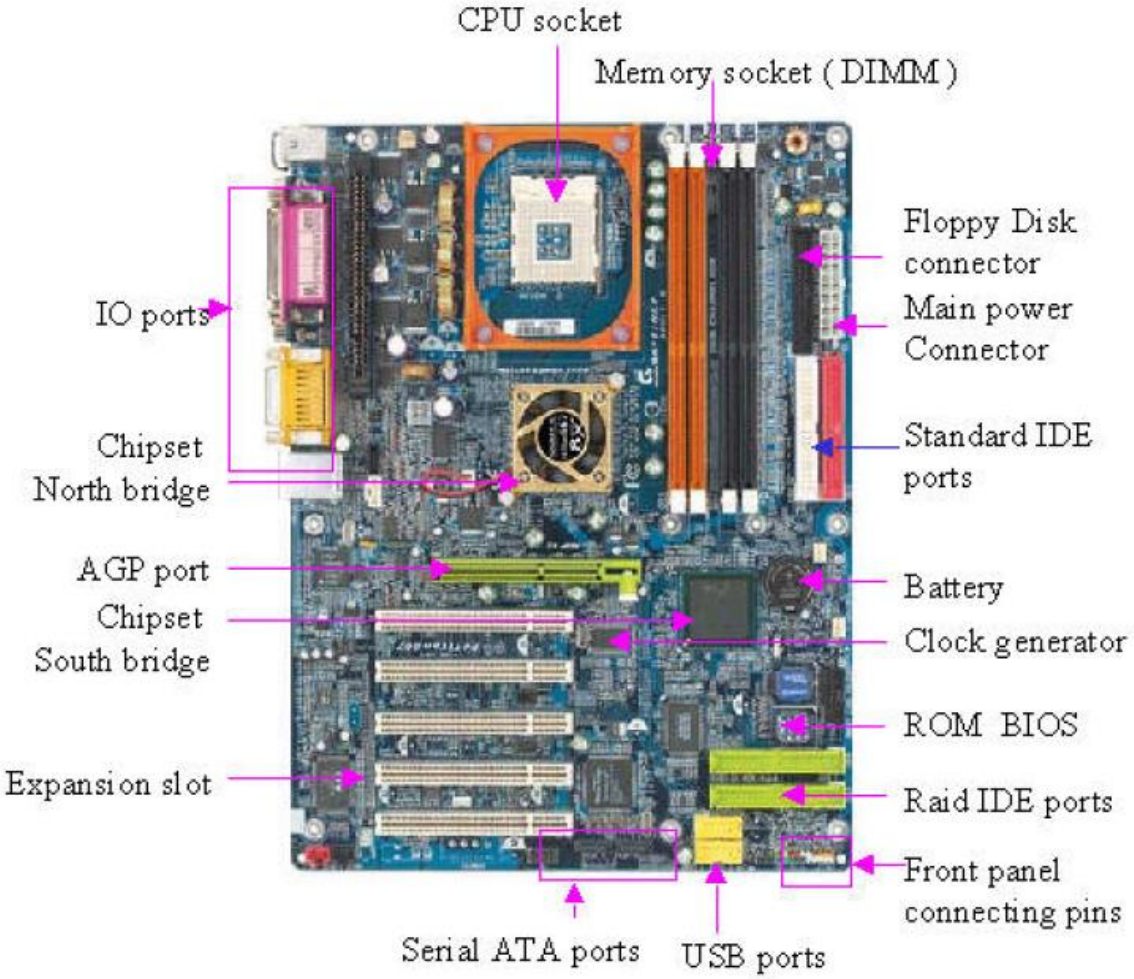
The mini-ITX motherboard has four mounting holes that line up with three or four of the holes in the ATX and micro ATX form factors. In mini-ITX boards, the rear interfaces are placed in the same location as those on the ATX motherboards. These features make mini-ITX boards compatible with ATX chassis. This is where the mounting compatibility ends because despite the PC compatibility of the other ITX form factors, they are used in embedded systems, such as set-top boxes, and lack the requisite mounting and interface specifications.

Components of Motherboard

Many of the following components can be found on a typical motherboard:

- Chipsets
- Expansion slots and buses
- Memory slots and external cache
- CPUs and their sockets

- Power connectors
- Onboard disk drive connectors
- Keyboard connectors
- Integrated peripheral ports and headers
- BIOS/firmware
- CMOS battery
- Jumpers and DIP switches



- Front-panel connectors

Bus Architecture

Parallel computer-system components work on the basis of a bus. A ***bus***, in this sense, is a common collection of signal pathways over which related devices communicate within the computer system. Expansion buses of various architectures, such as PCI and AGP, incorporate slots at certain points in the bus to allow insertion of external devices, or adapters, into the bus, usually with no regard to which adapters are inserted into which slots; insertion is generally arbitrary. Other types of buses exist within the system to allow communication between the CPU and components with which data must be exchanged. Except for CPU slots and sockets and memory slots, there are no insertion points in such closed buses because no adapters exist for such an environment.

The term *bus* is also used in any parallel or bit-serial wiring implementation where multiple devices can be attached at the same time in parallel or in series (daisy-chained). Examples include Small Computer System Interface (SCSI), USB, and Ethernet.

Chipsets

A *chipset* is a collection of chips or circuits that perform interface and peripheral functions for the processor. This collection of chips is usually the circuitry that provides inter-faces for memory, expansion cards, and onboard peripherals and generally dictates how a motherboard will communicate with the installed peripherals.

Chipsets are usually given a name and model number by the original manufacturer. Typically, the manufacturer and model also tell you that your particular chipset has a certain set of features (for example, type of RAM supported, type and brand of onboard video, and so on).

Chipsets can be made up of one or several integrated circuit chips. Intel-based mother-boards, for example, typically use two chips. To know for sure, you must check the manufacturer's documentation, especially because today's chipset chips are frequently obscured by cooling mechanisms, sometimes hindering visual brand and model identification.

The functions of chipsets can be divided into two major functional groups, called Northbridge and Southbridge. Let's take a brief look at these groups and the functions they perform.

Northbridge

The *Northbridge* subset of a motherboard's chipset is the set of circuitry or chips that performs one very important function: management of high-speed peripheral communications. The Northbridge is responsible primarily for communications with integrated video using AGP and PCIe, for instance, and processor-to-memory communications. Therefore, it can be said that much of the true performance of a PC relies on the specifications of the Northbridge component and its communications capability with the peripherals it controls.

The communications between the CPU and memory occur over what is known as the *frontside bus (FSB)*, which is just a set of signal pathways connecting the CPU and main memory, for instance. The

clock signal that drives the FSB is used to drive communications by certain other devices, such as AGP and PCIe slots, making them local-bus technologies. The *backside bus (BSB),* if present, is a set of signal pathways between the CPU and Level 2 or 3 (external) cache memory. The BSB uses the same clock signal that drives the FSB. If no backside bus exists, cache is placed on the frontside bus with the CPU and main memory.

The Northbridge is directly connected to the Southbridge (discussed next). It controls the Southbridge and helps to manage the communications between the Southbridge and the rest of the computer.

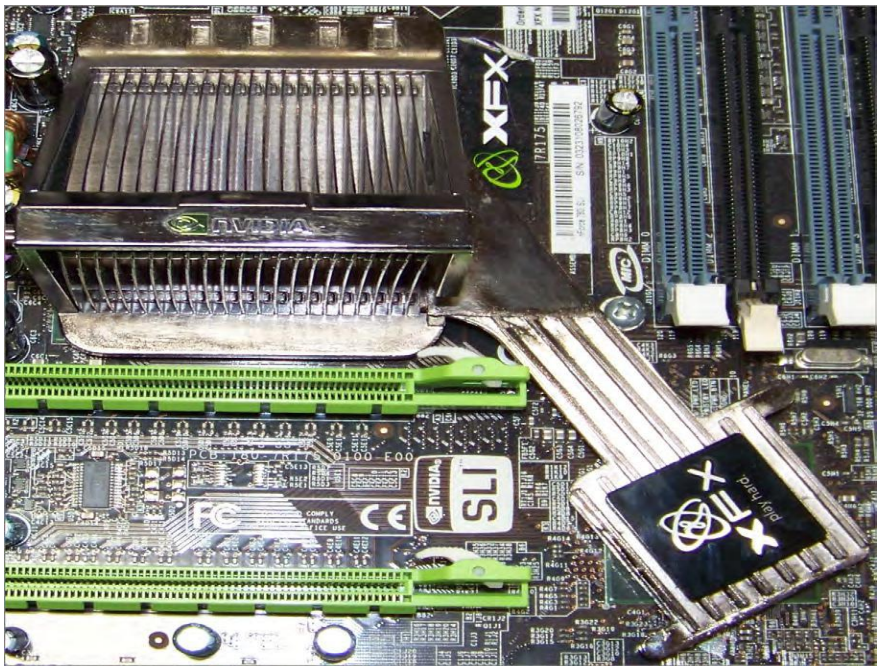
Southbridge

The *Southbridge* subset of the chipset is responsible for providing support to the onboard slower peripherals (PS/2, parallel ports, serial ports, Serial and Parallel ATA, and so on), man-aging their communications with the rest of the computer and the resources given to them. These components do not need to keep up with the external clock of the CPU and do not represent a bottleneck in the overall performance of the system. Any component that would impose such a restriction on the system should eventually be developed for FSB attachment.

In other words, if you're considering any component other than the CPU, memory and cache, AGP slots, or PCIe slots, the Southbridge is in charge. Most motherboards today have integrated PS/2, USB, LAN, analog and digital audio, and FireWire ports for the Southbridge to manage, for example, all of which are discussed in more detail later in this chapter. The Southbridge is also responsible for managing communications with the slower expansion buses, such as PCI, and legacy buses.

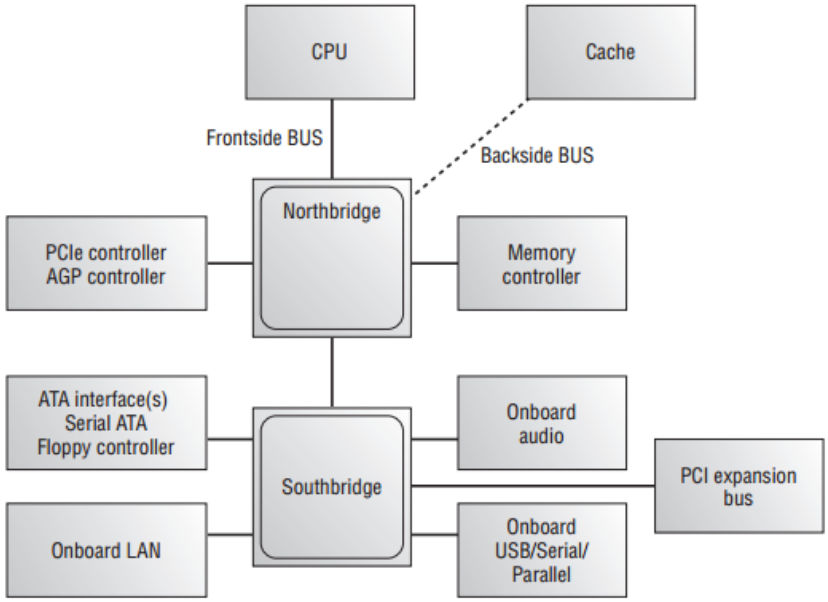
Figure 1.2 is a photo of the chipset of a motherboard, with the heat sink of the Northbridge, at the top left, connected to the heat-spreading cover of the Southbridge, at the bottom right.

Figure 1.3 shows a schematic of a typical motherboard chipset (both Northbridge and Southbridge) and the components they interface with. Notice which components interface with which parts of the chipset.



F i g u r e 1. 2 A modern computer chipset

FIGURE 1.3 A schematic of a typical motherboard chipset



Expansion Slots

The most visible parts of any motherboard are the *expansion slots*. These are small plastic slots, usually from 1 to 6 inches long and approximately 1⁄2 inch wide. As their name suggests, these slots are used to install various devices in the computer to expand its capabilities. Some expansion devices that might be installed in these slots include video, network, sound, and disk interface cards. Following are the main types of expansion slots used in computers today

- PCI
- AGP
- PCIe
- PCI-X
- CNR

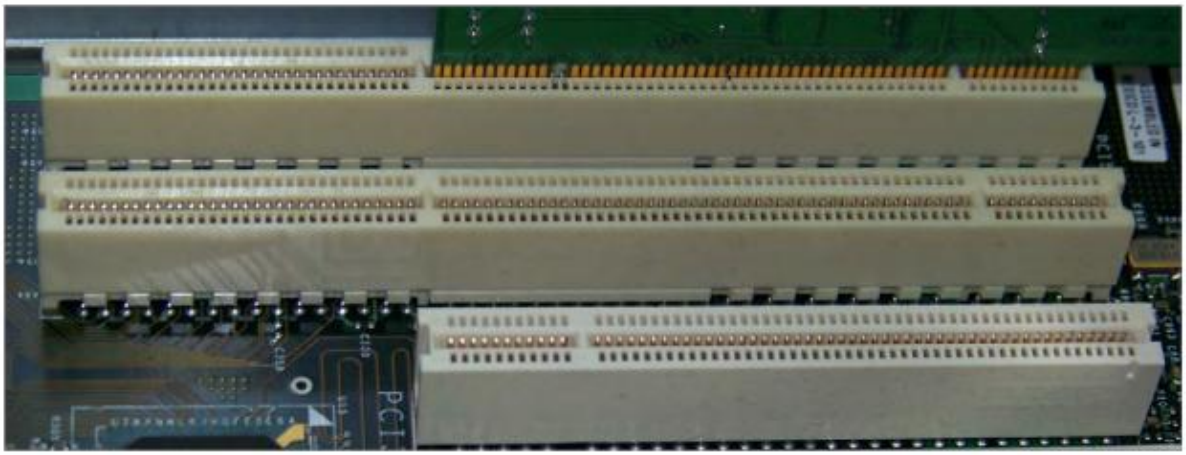
PCI Expansion Slots

Many computers in use today contain 32-bit Peripheral Component Interconnect (*PCI*) slots. PCI expansion buses operate at 33 or 66MHz over a 32-bit (4-byte) channel, resulting in data rates of 133 and

266MBps, respectively, with 133MBps being the most common, server architectures excluded. PCI is a shared-bus topology, however, so mixing 33 and 66MHz adapters in a 66MHz system will slow all adapters to 33MHz.

PCI slots and adapters are manufactured in 3.3 and 5V versions. Universal adapters are keyed to fit in slots based on either of the two voltages. The notch in the card edge of the com-mon 5V slots and adapters is oriented toward the front of the motherboard, and the notch in the 3.3V adapters toward the rear. Figure 1.4 shows several PCI expansion slots. Note the 5V 32-bit slot in the foreground and the 3.3V 64-bit slots. Also notice that a universal 32-bit card, which has notches in both positions, is inserted into and operates fine in the 64-bit 3.3V slot in the background.

FIGURE 1.4 PCI expansion slots



PCI-X Expansion Slots

Visually indistinguishable from 64-bit PCI, because it uses the same slots, PCI-Extended (*PCI-X*) takes the 66MHz maximum frequency of PCI to new heights, to the most com-mon, 133MHz, and the current maximum, 533MHz. With an 8-byte (64-bit) bus, this translates to maximum throughput of 4266MBps, roughly 4.3GBps. Additionally, PCI-X supports a 266MHz bus as well as the only frequency it shares with PCI, 66MHz, making PCI-X slots compatible with PCI adapters.

PCI-X is targeted at server platforms with its speed and support for hot-plugging but is still no match for the speeds available with PCIe, which all but obviates PCI-X today. PCI-X also suffers from the same shared-bus topology as PCI, resulting in all adapters falling back to the frequency of the slowest inserted adapter.

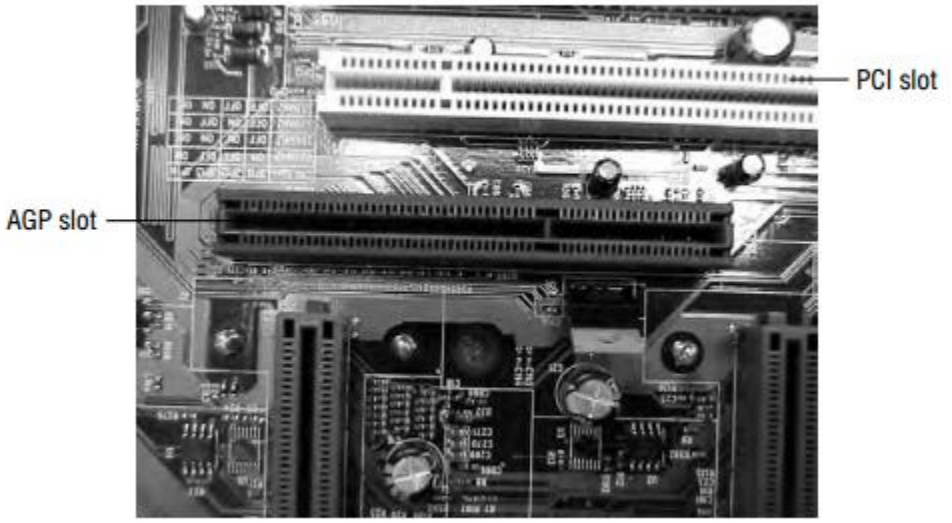
AGP Expansion Slots

Accelerated Graphics Port (*AGP*) slots are known mostly for legacy video card use and have been supplanted in new installations by PCI Express slots and their adapters. Preceding the introduction of AGP, if you wanted a high-speed, accelerated 3D graphics video card, you had to install the card into an existing PCI or ISA slot. AGP slots were designed to be a direct connection between the video circuitry and the PC's memory. They are also easily recognizable because they are usually brown and are located

right next to the PCI slots on the motherboard. AGP slots are slightly shorter than PCI slots and are pushed back from the rear of the motherboard in comparison with the position of the PCI slots.

Figure 1.5 shows an example of an older AGP slot, along with a PCI slot for comparison. Notice the difference in length between the two.

FIGURE 1.5 An AGP slot compared to a PCI slot



AGP performance is based on the original specification, known as AGP 1x. It uses a 32-bit (4-byte) channel and a 66MHz clock, resulting in a data rate of 266MBps. AGP 2x, 4x, and 8x specifications multiply the 66MHz clock they receive to increase throughput lin-early. For instance, AGP 8x uses the 66MHz clock to produce an effective clock frequency of 533MHz, resulting in throughput of 2133MBps over the 4-byte channel. Note that this maximum throughput is only half the maximum of PCI-X.

PCIe Expansion Slots

A newer expansion slot architecture that is being used by motherboards is PCI Express (*PCIe*). It was designed to be a replacement for AGP and PCI. PCIe has the advantage of being faster than AGP while maintaining the flexibility of PCI. PCIe has no plug compatibility with either AGP or PCI. As a result, modern PCIe motherboards still tend to have regular PCI slots for backward compatibility, but AGP slots typically are not also included. The lack of AGP means a legacy AGP video card must be replaced with a PCIe video card, often resulting in an appreciable expense. However, because PCI slots tend to be present as well, in general no other adapter requires replacement unless increased performance is desired.

PCIe is casually referred to as a bus architecture to simplify its comparison with other bus technologies. True expansion *buses* share total bandwidth among all slots, each of which taps into different points along the common bus lines. In contrast, PCIe uses a switching component with point-to-point connections to slots, giving each component full use of the corresponding bandwidth and producing more of a star topology versus a bus. Furthermore, unlike other expansion buses, which have parallel architectures, PCIe is a serial technology, striping data packets across multiple serial paths to achieve higher data rates.

PCIe uses the concept of *lanes*, which are the switched point-to-point signal paths between any two PCIe components. Each lane that the switch interconnects between any two intercommunicating devices comprises a separate pair of wires for both directions of traffic. Each PCIe pairing between cards requires a negotiation for the highest mutually supported number of lanes. The single lane or combined collection of lanes that the switch interconnects between devices is referred to as a *link*.

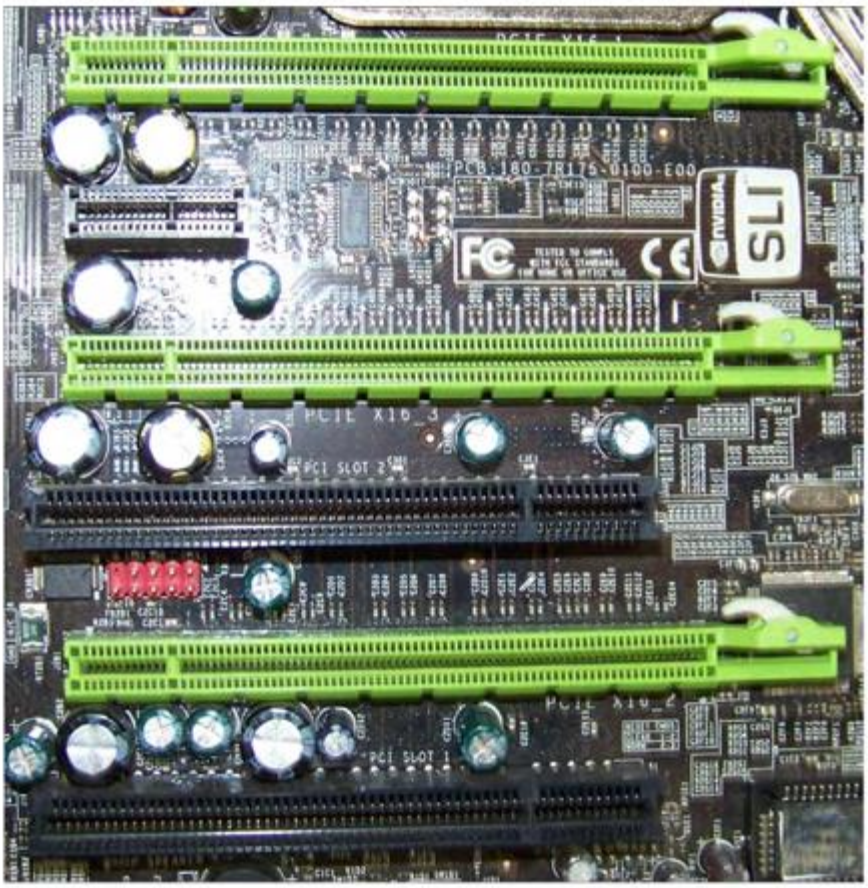
There are seven different link widths supported by PCIe, designated x1 (pronounced "by 1"), x2, x4, x8, x12, x16, and x32, with x1, x4, and x16 being the most common. The x8 link width is less common than these but more common than the others. A slot that supports a particular link width is of a physical size related to that width because the width is based on the number of lanes supported, requiring a related number of wires. As a result, a x8 slot is longer than a x1 slot but shorter than a x16 slot. Every PCIe slot has a 22-pin portion in common toward the rear of the motherboard, which you can see in Figure 1.6, in which the rear of the motherboard is to the left. These 22 pins comprise mostly voltage and ground leads.

There are three major versions of PCIe currently specified: 1.x, 2.x, and 3.0. The beginning of development on version 4.0 was announced in late 2011. During the same period, new motherboards were predominantly produced with PCIe 2.0 slots. For the four versions, a single lane, and hence a x1 slot, operates in each direction (or transmit and receive from either communicating device's perspective), at a data rate of 250MBps (almost twice the rate of the most common PCI slot), 500MBps, 1GBps, and 2GBps respectively.

An associated bidirectional link has a nominal throughput of double these rates. Use the doubled rate when comparing PCIe to other expansion buses because those other rates are for bidirectional communication. This means that the 500MBps bidirectional link of a x1 slot in the first version of PCIe was comparable to PCI's best, a 64-bit slot running at 66MHz and producing throughput of 533MBps.

Combining lanes results in a linear multiplication of these rates. For example, a PCIe 1.1 x16 slot is capable of 4GBps of throughput in each direction, 16 times the 250MBps x1 rate. Bidirectionally, this fairly common slot produces a throughput of 8GBps, qua-drupling the data rate of an AGP 8x slot. Later PCIe specifications increase this through-put even more.

Figure 1.6 PCIe expansion slots



Because of its high data rate, PCIe is the current choice of gaming aficionados. Additionally, technologies similar to NVIDIA's Scalable Link Interface (SLI) allow such users to combine preferably identical graphics adapters in appropriately spaced PCIe x16 slots with a hardware bridge to form a single virtual graphics adapter. The job of the bridge is to provide non-chipset communication among the adapters. The bridge is not a require-ment for SLI to work, but performance suffers without it. SLI-ready motherboards allow two, three, or four PCIe graphics adapters to pool their graphics processing units (GPUs) and memory to feed graphics output to a single monitor attached to the adapter acting as SLI master. SLI implementation results in increased graphics performance over single-PCIe and non-PCIe implementations.

Refer to Figure 1.6, which is a photo of an SLI-ready motherboard with three PCIe x16 slots (every other slot, starting with the top one), one PCIe x1 slot (second slot from the top), and two PCI slots (first and third slots from the bottom). Notice the latch and tab that secures the x16 adapters in place by their hooks. As with later AGP slots, any movement of these high-performance devices can result in temporary failure or poor performance.

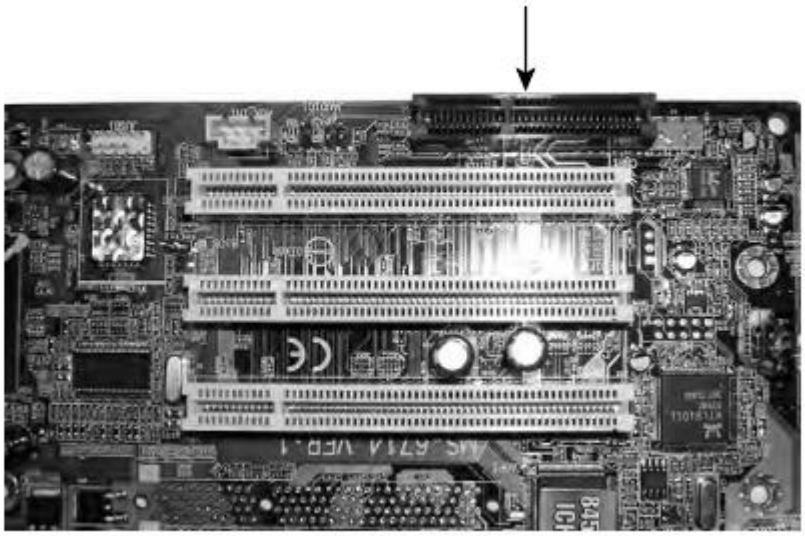
CNR Expansion Slots

As is always the case, Intel and other manufacturers are constantly looking for ways to improve the production process. One lengthy process that would often slow down the pro-duction of motherboards with integrated analog I/O functions was FCC certification. The manufacturers developed a way of separating the analog circuitry (for example, modem and analog audio) onto its own card. This allowed the analog circuitry to be separately certified (it was its own expansion card) from the already certified digital motherboard, thus reducing time for FCC certification. Eventually this became a nonissue and these cards became extinct.

The Communications and Networking Riser (*CNR*) slot that can be found on some older Intel motherboards was a replacement for Intel's even earlier Audio Modem Riser (AMR) slot, each of which appeared in quantities of no more than one per motherboard. One portion of this slot is the same length as one of the portions of the AMR slot, but the other portion of the CNR slot is longer than that of the AMR slot. The cards made for the CNR slot contained circuitry for sound and analog modem (communications) as well as networking.

Essentially, these legacy 60-pin slots allowed motherboard manufacturers to implement a motherboard chipset with certain integrated features. Then, if the built-in features of that chipset need to be enhanced (by adding Dolby Digital Surround to a standard sound chipset, for example), a CNR riser card could be added to enhance the onboard capabili-ties. Additional advantages of CNR over AMR include networking support, Plug and Play compatibility, support for hardware acceleration (as opposed to CPU control only), and the fact that there's no need to lose a competing PCI slot for networking unless the CNR slot is in use. Figure 1.7 shows an example of a CNR slot (indicated by the arrow).

FIGURE 1.7 A CNR slot



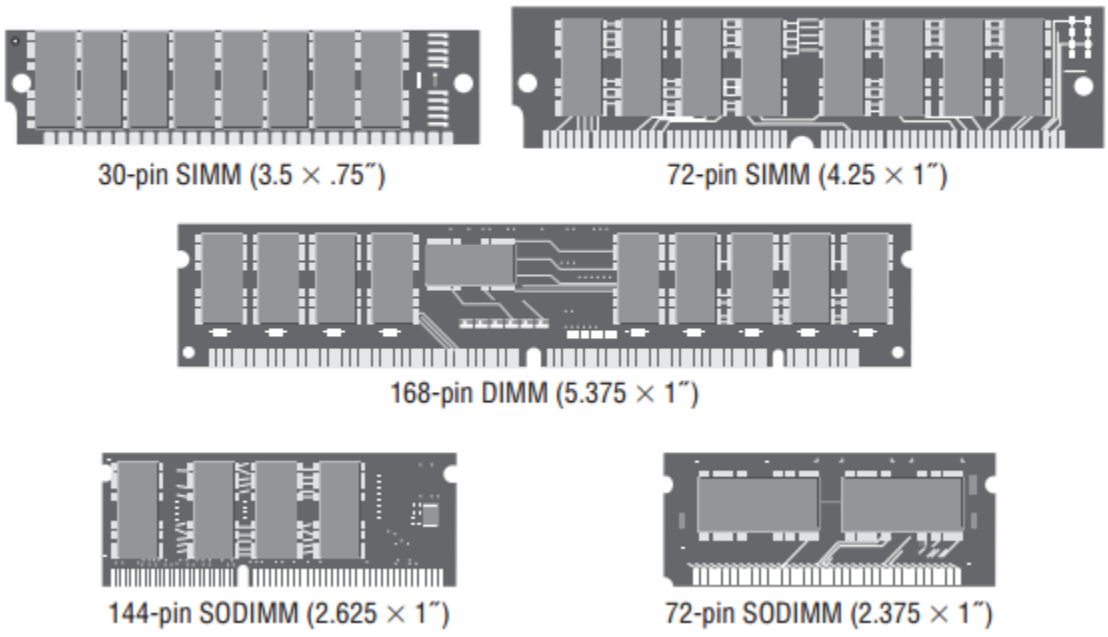
Memory Slots and Cache

Memory or random access memory (RAM) slots are the next most notable slots on a motherboard. These slots are for the modules that hold memory chips that make up primary memory that is used to store currently used data and instructions for the CPU. Many and varied types of memory are available for PCs today.

For the most part, PCs today use memory chips arranged on a small circuit board. A dual inline memory module (*DIMM*) is one type of circuit board. Today's DIMMs differ in the number of conductors, or pins, that each particular physical form factor uses. Some common examples include 168-, 184-, and 240-pin configurations. In addition, laptop memory comes in smaller form factors known as small outline DIMMs (*SODIMM*s) and MicroDIMMs. The single inline memory module (SIMM) is an older memory form factor that began the trend of placing memory chips on modules.

Memory slots are easy to identify on a motherboard. Classic DIMM slots were usually black and, like all memory slots, were placed very close together. DIMM slots with color coding are more common these days, however. The color coding of the slots acts as a guide to the installer of the memory. See the section "Single-, Dual-, and Triple-Channel Memory" later in this chapter for more on the purpose of this color coding. Consult the motherboard's docu-mentation to determine the specific modules allowed as well as their required orientation. The number of memory slots varies from motherboard to motherboard, but the structure of the different slots is similar. Metal pins in the bottom make contact with the metallic pins on each memory module. Small metal or plastic tabs on each side of the slot keep the memory module securely in its slot.

FIGURE 1.8 Different memory module form factors



Cache Memory :

The Cache Memory is the Memory which is very nearest to the CPU , all the Recent Instructions are Stored into the Cache Memory. The Cache Memory is attached for storing the input which is given by the user and which is necessary for the CPU to Perform a Task. But the Capacity of the Cache Memory is too low in compare to Memory and Hard Disk.

Only the most recently used data and code or that which is expected to be used next is stored in cache. Cache on the motherboard is known as external cache because it is external to the processor; it's also referred to as Level 2 cache (*L2 cache*). Level 1 cache (*L1 cache*), by comparison, is internal cache because it is built into the processor's silicon wafer, or *die*. The word *core* is often interchangeable with the word *die*.

It is now common for chip makers to use extra space in the processor's packaging to bring the L2 cache from the motherboard closer to the CPU. When L2 cache is present in the pro-cessor's packaging, but not on-die, the cache on the motherboard is referred to as Level 3 cache (*L3 cache*). Unfortunately, due to the de facto naming of cache levels, the term *L2 cache* alone is not a definitive description of where the cache is located. The terms *L1 cache* and *L3 cache* do not vary in their meaning, however.

The typical increasing order of capacity and distance from the processor die is L1 cache, L2 cache, L3 cache, RAM, HDD/SSD (hard disk drive and solid-state drive—more on these in Chapter 2, "Storage Devices and Power Supplies"). This is also the typical decreasing order of speed. The following list includes representative capacities of these memory types. The cache capacities are for each core of the original Intel Core i7 processor. The other capacities are simply modern examples.

- L1 cache—64KB (32KB each for data and instructions)
- L2 cache—256KB
- L3 cache—4MB–12MB
- RAM—4–16GB
- HDD/SSD—100s–1000s of GB

Central Processing Unit and Processor Socket

The "brain" of any computer is the central processing unit (CPU). There's no computer without a CPU. Typically, in today's computers, the processor is the easiest component to identify on the motherboard. It is usually the component that has either a fan or a heat sink (usually both) attached to it (as shown in Figure 1.9). These devices are used to draw away and disperse the heat a processor generates. This is done because heat is the enemy of micro-electronics. Theoretically, a Pentium (or higher) processor generates enough heat that without the heat sink it would permanently damage itself and the motherboard in a matter of hours or even minutes.

CPU sockets are almost as varied as the processors they hold. Sockets are basically flat and have several columns and rows of holes or pins arranged in a square, as shown in Figure 1.10. The top socket is known as Socket A or Socket 462, made for earlier AMD processors such as the Athlon, and has holes to receive the pins on the CPU. This is known as a pin grid array (PGA) arrangement for a CPU socket. The

holes and pins are in a row column orientation, an array of pins. The bottom socket is known as Socket T or Socket LGA 775, and there are spring-loaded pins in the socket and a grid of lands on the CPU. The land grid array (LGA) is a newer technology that places the delicate pins (yet more sturdy than those on chips) on the cheaper motherboard instead of on the more expensive CPU, opposite to the way the aging PGA does. The device with the pins has to be replaced if the pins become too damaged to function. The PGA and LGA are mentioned again later in this chapter in the section "Identifying Purposes and Characteristics of Processors."

FIGURE 1.9 Two heat sinks, one with a fan

Modern CPU sockets have a mechanism in place that reduces the need to apply the consid-erable force to the CPU that was necessary in the early days of personal computing to install a processor. Given the extra surface area on today's processors, excessive pressure applied in the wrong manner could damage the CPU packaging, its pins, or the motherboard itself. For CPUs based on the PGA concept, zero insertion force (*ZIF*) sockets are exceedingly popular. ZIF sockets use a plastic or metal lever on one of the two lateral edges to lock or release the mechanism that secures the CPU's pins in the socket. The CPU rides on the mobile top por-tion of the socket, and the socket's contacts that mate with the CPU's pins are in the fixed bottom portion of the socket. The Socket 462 image in Figure 1.10 shows the ZIF locking mechanism at the edge of the socket along the bottom of the photo.

For processors based on the LGA concept, a socket with a different locking mechanism is used. Because there are no receptacles in either the motherboard or the CPU, there is no opportunity for a locking mechanism that holds the component with the pins in place. LGA-compatible sockets, as they're called despite the misnomer, have a lid that closes over the CPU and is locked in place by an L-shaped arm that borders two of the socket's edges. The nonlocking leg of the arm has a bend in the middle that latches the lid closed when the other leg of the arm is secured. The bottom image in Figure 1.10 shows an LGA socket with no CPU installed and the locking arm secured over the lid's tab (right-hand edge in the photo).

FIGURE 1.10 CPU socket examples

Table 1.1 lists some common socket/CPU relationships.

TABLE 1.1: Socket types and the processors they support

Socket	Processors
LGA 775 (Socket T)	Intel only: Pentium 4, Pentium 4 Extreme Edition (single core), Pentium D, Celeron D, Pentium Extreme Edition (dual core), Core 2 Duo, Core 2 Extreme, Core 2 Quad, Xeon, Celeron (4xx, Exxxx series)
LGA 1155 (Socket H2)	Intel only: Replacement for LGA 1156 to support CPUs based on the Sandy Bridge (such as Celeron G4xx and G5xx) and eventual Ivy Bridge architectures
LGA 1156 (Socket H)	Intel only: Celeron (G1xxx series), Core i3, Core i5, Core i7 (8xx series), Pentium (G6xxx series), Xeon (34xx series)
LGA 1366 (Socket B)	Intel only: Core i7 (9xx series), Xeon (35xx, 36xx, 55xx, 56xx series), Intel Celeron P1053
Socket 940	AMD only: Athlon 64 FX (FX-51, -53), Opteron
Socket AM2	AMD only: Athlon 64, Athlon 64 X2, Athlon 64 FX, Opteron, Sempron, Phenom
Socket AM2+	AMD only: Often backward compatible with AM2 CPUs as well as Athlon II and Phenom II and forward compatible with AM3 CPUs
Socket AM3	AMD only: DDR3 capable CPUs only (thus not compatible with AM2+ CPUs), such as Phenom II, Athlon II, Sempron, Opteron 138x, and has the potential to accept AM3+ CPUs
Socket AM3+	AMD only: Specified for CPUs based on the Bulldozer microarchitecture and designed to accept AM3 CPUs
Socket FM1	AMD only: Designed to accept AMD Fusion APUs that incorporate CPUs and GPUs, such as the E2-3200 and the A Series
Socket F (LGA)	AMD only: Opteron (2xxx, 8xxx series), Athlon 64 FX (FX-7x series), and replaced by Sockets C32 and G34

Firmware

Firmware is the name given to any software that is encoded in hardware, usually a read-only memory (ROM) chip, and can be run without extra instructions from the operating system. Most computers and large printers use firmware in some sense. The best example of firmware is a computer's Basic Input/Output System (BIOS) routine, which is burned in to a chip. Also, some expansion cards, such as SCSI cards and graphics adapters, use their own firmware utilities for setting up peripherals.

BIOS and POST

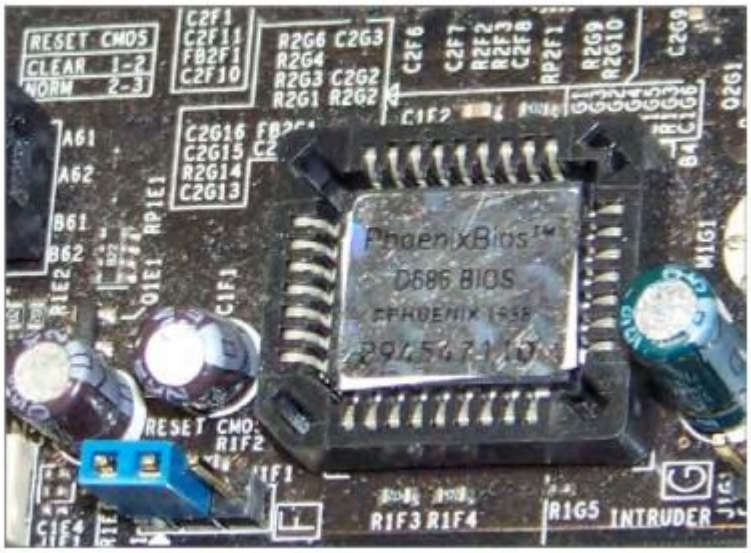
Aside from the processor, the most important chip on the motherboard is the Basic Input/ Output System (*BIOS*) chip, also referred to as the ROM BIOS chip. This special memory chip contains the BIOS system software that boots the system and allows the operating system to interact with certain hardware in the computer in lieu of requiring a more com-plex device driver to do so. The BIOS chip is easily identified: If you have a brand-name computer, this chip might have on it the name of the manufacturer and usually the word *BIOS*. For clones, the chip usually has a sticker or printing on it from

one of the major BIOS manufacturers (AMI, Phoenix/Award, Winbond, and so on). On later motherboards, the BIOS might be difficult to identify or even be integrated into the Southbridge, but the functionality remains, regardless of how it's implemented.

BIOS

Figure 1.12 gives you an idea of what a modern BIOS might look like. Despite the 1998 copy-right on the label, which refers only to the oldest code present on the chip, this particular chip can be found on motherboards produced as late as 2009. Notice also the Reset CMOS jumper at lower left and its configuration silkscreen at upper left. You might use this jumper to clear the CMOS memory, discussed next, when an unknown password, for example, is keeping you out of the BIOS configuration utility. The jumper in the photo is in the clear posi-tion, not the normal operating position. System boot-up is typically not possible in this state.

FIGURE 1.12 A BIOS chip on a motherboard



Most BIOS setup utilities have more to offer than a simple interface for making selections and saving the results. As always, you can enter the utility to check to see if the clock appears to be losing time, possibly due to a dying battery. Today, these utilities also offer diagnostic routines that you can use to have the BIOS analyze the state and quality of the same compo-nents it inspects during boot-up, but at a much deeper level.

There is often also a page within the utility that gives you access to such bits of information as current live readings of the temperature of the CPU and the ambient temperature of the inte-rior of the system unit. In such a page, you can set the temperature at which the BIOS sounds a warning tone and the temperature at which the BIOS shuts the system down to protect it. You can also monitor the instantaneous fan speeds, bus speeds, and voltage levels of the CPU and other vital landmarks to make sure they are all within acceptable ranges. You might also be able to set a lower fan-speed threshold at which the system

warns you. In many cases, some of these levels can be altered to achieve such phenomena as overclocking or undervolting.

Some BIOS firmware can monitor the status of a contact on the motherboard for intrusion detection. If the feature in the BIOS is enabled and the sensor on the chassis is connected to the contact on the motherboard, the removal of the cover will be detected and logged by the BIOS. This can occur even if the system is off, thanks to the CMOS battery. At the next boot-up, the BIOS will notify you of the intrusion. No notification occurs over subsequent boots unless additional intrusion is detected.

POST

A major function of the BIOS is to perform a process known as a power-on self-test (POST). POST is a series of system checks performed by the system BIOS and other high-end components, such as the SCSI BIOS and the video BIOS. Among other things, the POST routine verifies the integrity of the BIOS itself. It also verifies and confirms the size of primary memory. During POST, the BIOS also analyzes and catalogs other forms of hardware, such as buses and boot devices, as well as manages the passing of control to the specialized BIOS routines mentioned earlier. The BIOS is responsible for offering the user a key sequence to enter the configuration routine as POST is beginning. Finally, once POST has completed successfully, the BIOS selects the boot device highest in the config-ured boot order and executes the master boot record (MBR) or similar construct on that device so that the MBR can call its associated operating system's boot loader and continue booting up.

The POST process can end with a beep code or displayed code that indicates the issue discovered. Each BIOS publisher has its own series of codes that can be generated. Figure 1.13 shows a simplified POST display during the initial boot sequence of a computer.

CMOS and CMOS Battery

Your PC has to keep certain settings when it's turned off and its power cord is unplugged:

- Date
- Time
- Hard drive/optical drive configuration
- Memory
- CPU settings, such as overclocking
- Integrated ports (settings as well as enable/disable)
- Boot sequence
- Power management
- Virtualization support
- Security (passwords, trusted platform module settings, LoJack)

Your PC keeps these settings in a special memory chip called the complementary metal oxide semiconductor (*CMOS*) memory chip. Actually, CMOS (usually pronounced *see-moss*) is a

manufacturing technology for integrated circuits. The first commonly used chip made from CMOS technology was a type of memory chip, the memory for the BIOS. As a result, the term *CMOS* stuck and is the accepted name for this memory chip.

The BIOS starts with its own default information and then reads information from the CMOS, such as which hard drive types are configured for this computer to use, which drive(s) it should search for boot sectors, and so on. Any overlapping information read from the CMOS overrides the default information from the BIOS. A lack of corresponding information in the CMOS does not delete information that the BIOS knows natively. This process is a merge, not a write-over. CMOS memory is usually *not* upgradable in terms of its capacity and might be integrated into the BIOS chip or the Southbridge.

To keep its settings, integrated circuit-based memory must have power constantly. When you shut off a computer, anything that is left in this type of memory is lost forever. The CMOS manufacturing technology produces chips with very low power requirements. As a result, today's electronic circuitry is more susceptible to damage from electrostatic discharge (ESD). Another ramification is that it doesn't take much of a power source to keep CMOS chips from losing their contents.

To prevent CMOS from losing its rather important information, motherboard manufac-turers include a small battery called the *CMOS battery* to power the CMOS memory. The batteries come in different shapes and sizes, but they all perform the same function. Most CMOS batteries look like large watch batteries or small cylindrical batteries. Today's CMOS batteries are most often of a long-life, nonrechargeable lithium chemistry.

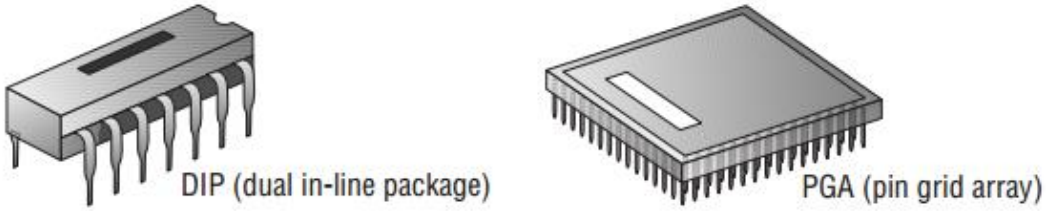
Identifying Purposes and Characteristics of Processors

The most important component on the motherboard is CPU. The role of the CPU, or central processing unit, is to control and direct all the activities of the computer using both external and internal buses. It is a processor chip consisting of an array of *millions* of transistors.Intel and Advanced Micro Devices (AMD) are the two largest PC-compatible CPU manufacturers.

Older CPUs are generally square, with contacts arranged in a pin grid array (PGA). Prior to 1981, chips were found in a rectangle with two rows of 20 pins known as a dual inline package (DIP); see Figure 1.16. There are still integrated circuits that use the DIP form factor. However, the DIP form factor is no longer used for PC CPUs. Most CPUs use either the PGA/SPGA or LGA form factor.

Intel and AMD both make extensive use of an inverted socket/processor combination of sorts. As mentioned earlier, the land grid array (LGA) packaging calls for the pins to be placed on the motherboard, while the mates for these pins are on the processor packaging. As with PGA, LGA is named for the landmarks on the processor, not the ones on the mother board. As a result, the grid of metallic contact points, called lands, on the bottom of the CPU gives this format its name.

FIGURE 1.16 DIP and PGA



You can easily identify which component inside the computer is the CPU because it is a large square lying flat on the motherboard with a very large heat sink and fan (as shown earlier in Figure 1.9). Figure 1.17 shows the location of the CPU in relation to the other components on a typical ATX motherboard. Notice how prominent the CPU is.

FIGURE 1.17 The location of a CPU inside a typical computer

Hyperthreading This term refers to Intel's Hyper-Threading Technology (HTT). HTT is a form of simultaneous multithreading (SMT). SMT takes advantage of a modern CPU's superscalar architecture. Superscalar processors are able to have multiple instructions oper-ating on separate data in parallel.

HTT-capable processors appear to the operating system to be two processors. As a result, the operating system can schedule two processes at the same time, as in the case of symmetric multiprocessing (SMP), where two or more processors use the same system resources. In fact, the operating system must support

SMP in order to take advantage of HTT. If the current process stalls because of missing data caused by, say, cache or branch prediction issues, the execution resources of the processor can be reallocated for a different process that is ready to go, reducing processor downtime.

Multicore A processor that exhibits a multicore architecture has multiple completely sepa-rate processor dies in the same package. The operating system and applications see multiple processors in the same way that they see multiple processors in separate sockets. As with HTT, the operating system must support SMP to benefit from the separate processors. In addition, SMP is not a benefit if the applications run on the SMP system are not written for parallel processing. Dual-core and quad-core processors are common specific examples of the multicore technology.

Don't be confused by Intel's Core 2 labeling. The numeric component does not imply there are two cores. There was a Core series of 32-bit mobile processors that featured one (Solo) or two (Duo) processing cores on a single die (silicon wafer). The same dual-core die was used for both classes of Core CPU. The second core was disabled for Core Solo processors.

The 64-bit Core 2 product line can be thought of as a second generation of the Core series. Core 2, by the way, reunited Intel mobile and desktop computing—the Pentium 4 family had a separate Pentium M for mobile computing. Intel describes and markets the microcode of cer-tain processors as "Core microarchitecture." As confusing as it may sound, the Core 2 proces-sors are based on the Core microarchitecture; the Core processors are not. Core 2 processors come in Solo (mobile only), Duo, and four-core (Quad) implementations. Solo and Duo pro-cessors have a single die; Quad processors have two Duo dies. A more capable Extreme version exists for the Duo and Quad models.

Processors, such as certain models of AMD's Phenom series, can contain an odd number of multiple cores as well. The triple-core processor, which obviously contains three cores, is the most common implementation of multiple odd cores.

Throttling CPU throttling allows reducing the operating frequency of the CPU during times of less demand or during battery operation. CPU throttling is very common in pro-cessors for mobile devices, where heat generation and system-battery drain are key issues of full power usage. You might discover throttling in action when you use a utility that reports a lower CPU clock frequency than expected. If the load on the system does not require full-throttle operation, there is no need to push such a limit.

Speed The speed of the processor is generally described in clock frequency (MHz or GHz). Since the dawn of the personal computer industry, motherboards have included oscillators, quartz crystals shaved down to a specific geometry so that engineers know exactly how they will react when a current is run through them. The phenomenon of a quartz crystal vibrating when exposed to a current is known as the *piezoelectric effect*. The crystal (XTL) known as the system clock keeps the time for the flow of data on the motherboard. How the clock is used by the frontside bus leads to an *effective* clock rate known as the FSB speed. As shown in the section "Types of Memory" later in this chapter, the FSB speed is computed differently for different types of RAM (DDR, DDR2, etc.). From here, the CPU multiplies the FSB speed to produce its own internal clock rate, producing the third *speed* mentioned thus far.

As a result of the foregoing tricks of physics and mathematics, there can be a discrepancy between the frontside bus frequency and the internal frequency that the CPU uses to latch data and instructions through its pipelines. This disagreement between the numbers comes from the fact that the CPU is capable of splitting the clock signal it receives from the exter-nal oscillator that drives the frontside bus into multiple regular signals for its own internal use. In fact, you might be able to purchase a number of processors rated for different (inter-nal) speeds that are all compatible with a single motherboard that has a frontside bus rated, for instance, at 1333MHz. Furthermore, you might be able to adjust the internal clock rate of the CPU you purchased through settings in the BIOS. The successful technician needs to be familiar with more basic information than this, however. The sidebar titled "Matching System Components" explains these basics.

32- and 64-bit processors The set of data lines between the CPU and the primary memory of the system can be 32 or 64 bits wide, among other widths. The wider the bus, the more data that can be processed per unit of time, and hence, more work can be performed. Internal registers in the CPU might be only 32 bits wide, but with a 64-bit system bus, two separate pipelines can receive information simultaneously. For true 64-bit CPUs, which have 64-bit internal registers and can run x64 versions of Microsoft operating systems, the external sys-tem data bus will always be 64 bits wide or some larger multiple thereof.

Virtualization support Many of today's CPUs support virtualization in hardware, which eases the burden on the system that software-based virtualization imposes. For more infor-mation on virtualization, see Chapter 12. Unlike AMD's AMD-V (*V* for virtualization) technology, which is widely inclusive of AMD's CPUs, Intel's Virtualization Technology (VT) is used by Intel to segment its market for CPUs made concurrently. For example, you can find Intel VT on the Core 2 Duo processor in the E6000 series and most of the E8000 series but not in the E7000 series. In some cases, you must also first enable the virtualiza-tion support in the BIOS before it can be used. If you have an Intel processor and would like to check its support of VT, visit

downloadcenter.intel.com/Detail_Desc.aspx?ProductID=1881&DwnldID=7838

to download the Intel Processor Identification utility. As shown in Figure 1.18, the CPU Technologies tab of this utility tells you if your CPU supports Intel VT.

Integrated GPU Intel and AMD both have a line of low-power CPUs aimed at the net-book and embedded markets that have built-in graphics processing units (GPUs). Building in specialized functionality to CPUs is nothing new, but before now, math coprocessors were some of the most complex features added on to the die of CPUs. A GPU, then, which is normally a large chip on your graphics adapter, is quite a bit more complex than any-thing heretofore integrated into the CPU. Integrated GPUs take much of the burden off of the CPU itself in addition to minimizing the amount of off-package communication that must occur, which improves overall system performance. As if that were not enough, the CPUs in this class are quite a bit smaller than standard CPUs. The Intel Atom and AMD Fusion lines of CPUs have built-in GPUs and open the door for other complex systems to be built into future processors.

Identifying Purposes and Characteristics of Memory

. As the computer's CPU works, it stores data and instructions in the computer's memory. Contrary to what you might expect from an inexpensive solution, memory upgrades tend to afford the greatest performance increase as well, up to a point. Motherboards have memory limits; operating systems have memory limits; CPUs have memory limits.

To visually identify memory within a computer, look for several thin rows of small cir-cuit boards sitting vertically, potentially packed tightly together near the processor. In situ-ations where only one memory stick is installed, it will be that stick and a few empty slots that are tightly packed together. Figure 1.19 shows where memory is located in a system.

Figure 1.19 Location of memory within a system

As the computer's CPU works, it stores data and instructions in the computer's memory. To visually identify memory within a computer, look for several thin rows of small circuit boards sitting vertically, potentially packed tightly together near the processor.

There are a few technical terms and phrases that you need to understand with regard to memory and its function:

- ☐ Parity checking
- ☐ Error-correcting code (ECC)
- ☐ Single- and double-sided memory

- Single-, dual-, and triple-channel memory

Parity Checking and Memory Banks

Parity checking is a rudimentary error-checking scheme that offers no error correction. Parity checking works most often on a byte, or 8 bits, of data. A ninth bit is added at the transmitting end and removed at the receiving end so that it does not affect the actual data transmitted. The four most common parity schemes affecting this extra bit are known as even, odd, mark, and space. Even and odd parity are used in systems that actually compute parity. Mark (a term for a 1 bit) and space (a term for a 0 bit) parity are used in systems that do not compute parity but expect to see a fixed bit value stored in the parity location. Systems that do not support or reserve the location required for the parity bit are said to implement non-parity memory.

The most basic model for implementing memory in a computer system uses eight memory chips to form a set. Each memory chip holds millions or billions of bits of information, each in its own cell. For every byte in memory, one bit is stored in each of the eight chips. A ninth chip is added to the set to support the parity bit in systems that require it. One or more of these sets, implemented as individual chips or as chips mounted on a memory module, form memory bank.

A bank of memory is required for the computer system to electrically recognize that the minimum number of memory components or the proper number of additional memory com-ponents has been installed. The width of the system data bus, the external bus of the proces-sor, dictates how many memory chips or modules are required to satisfy a bank. For example, one 32-bit, 72-pin SIMM (single inline memory module) satisfies a bank for an old 32-bit CPU, such as a 386 or 486 processor. Two such modules are required to satisfy a bank for a 64-bit processor, a Pentium, for instance. However, only a single 64-bit, 168-pin DIMM is required to satisfy the same Pentium processor. For those modules that have fewer than eight or nine chips mounted on them, more than one bit for every byte is being handled by some of the chips. For example, if you see three chips mounted, the two larger chips probably handle four bits, a nybble, from each byte stored, and the third, smaller chip probably handles the single parity bit for each byte.

Even and odd parity schemes operate on each byte in the set of memory chips. In each case, the number of bits set to a value of 1 is counted up. If there are an even number of 1 bits in the byte (0, 2, 4, 6, or 8), even parity stores a 0 in the ninth bit, the parity bit; other-wise, it stores a 1 to even up the count. Odd parity does just the opposite, storing a 1 in the parity bit to make an even number of 1s odd and a 0 to keep an odd number of 1s odd. You can see that this is effective only for determining if there was a blatant error in the set of bits received, but there is no indication as to where the error is and how to fix it. Furthermore, the total 1-bit count is not important, only whether it’s even or odd. Therefore, in either the even or odd scheme, if an even number of bits is altered in the same byte during transmis-sion, the error goes undetected because flipping two, four, six, or all eight bits results in an even number of 1s remaining even and an odd number of 1s remaining odd.

Mark and space parity are used in systems that want to see nine bits for every byte trans-mitted but don't compute the parity bit's value based on the bits in the byte. Mark parity always uses a 1 in the parity bit, and space parity always uses a 0. These schemes offer less error detection capability than the even and odd schemes because only changes in the parity bit can be detected. Again, parity checking is not error correction; it's error detection only, and not the best form of error detection at that. Nevertheless, finding an error can lock up the entire system and display a memory parity error. Enough of these errors and you need to replace the memory.

In the early days of personal computing, almost all memory was parity based. As quality has increased over the years, parity checking in the RAM subsystem has become rarer. As noted earlier, if parity checking is not supported, there will generally be fewer chips per mod-ule, usually one less per column of RAM.

Error Checking and Correction

The next step in the evolution of memory error detection is known as error-correcting code (*ECC*). If memory supports ECC, check bits are generated and stored with the data. An algorithm is performed on the data and its check bits whenever the memory is accessed. If the result of the algorithm is all zeros, then the data is deemed valid and processing contin-ues. ECC can detect single- and double-bit errors and actually correct single-bit errors.

Single- and Double-Sided Memory

Commonly speaking, the terms *single-sided memory* and *double-sided memory* refer to how some memory modules have chips on one side while others have chips on both sides. Double-sided memory is essentially treated by the system as two separate memory modules. Motherboards that support such memory have memory controllers that must switch between the two "sides" of the modules and, at any particular moment, can access only the side they have switched to. Double-sided memory allows more memory to be inserted into a computer using half the physical space of single-sided memory, which requires no switching by the memory controller.

Single-, Dual-, and Triple-Channel Memory

Standard memory controllers manage access to memory in chunks of the same size as the system bus's data width. This is considered communicating over a single channel. Most modern processors have a 64-bit system data bus. This means a standard memory controller can transfer exactly 64 bits of information at a time. Communicating over a single channel is a bottleneck in an environment where the CPU and memory can both operate faster than the conduit between them. Up to a point, every channel added in parallel between the CPU and RAM serves to ease this constriction.

Memory controllers that support dual- and triple-channel memory implementation were developed in an effort to alleviate the bottleneck between the CPU and RAM. *Dual-channel memory* is the memory controller's coordination of two memory banks to work as a synchro-nized set during communication with the CPU, doubling the specified system bus width from the memory's perspective. *Triple-channel memory*, then, demands the coordination of three memory modules at a time.

The major difference between dual- and triple-channel architectures is that triple-channel memory employs a form of interleaving that reduces the amount of information transferred by each module. Nevertheless, there is an overall performance increase over that of dual-channel memory because of the ability to access more information per unit of time with triple-channel memory.

Because today's processors largely have 64-bit external data buses, and because one stick of memory satisfies this bus width, there is a 1:1 ratio between banks and modules. This means that implementing dual- and triple-channel memory in today's most popular computer systems requires that pairs or triads of memory modules be installed at a time. Note, however, that it's the motherboard, not the memory, that implements dual- and triple-channel memory (more on this in a moment). *Single-channel memory*, in contrast, is the classic memory model that dictates only that a complete bank be satisfied when-ever memory is initially installed or added. One bank supplies only half the width of the effective bus created by dual-channel support, for instance, which by definition pairs two banks at a time.

In almost all cases, multichannel implementations support single-channel installation, but poorer performance should be expected. The same loss of performance occurs when only two modules are installed in a triple-channel motherboard. Multichannel motherboards include slots of different colors, usually one of each color per set of slots. To use only a single channel, you populate slots of the same color, skipping neighboring slots to do so. Filling neighboring slots in a dual-channel motherboard takes advantage of its dual-channel capability.

Because of the special tricks that are played with memory subsystems to improve over-all system performance, care must be taken during the installation of disparate memory modules. In the worst case, the computer will cease to function when modules of different speeds, different capacities, or different numbers of sides are placed together in slots of the same channel. If all of these parameters are identical, there should be no problem with pair-ing modules. Nevertheless, problems could still occur when modules from two different manufacturers or certain unsupported manufacturers are installed, all other parameters being the same. Technical support or documentation from the manufacturer of your moth-erboard should be able to help with such issues.

Although it's not the make-up of the memory that leads to dual-channel support but instead the technology on which the motherboard is based, some memory manufacturers still package and sell pairs and triplets of memory modules in an effort to give you peace of mind when you're buying memory for a system that implements dual- or triple-channel memory architecture. Keep in mind, the motherboard memory slots have the distinctive color coding, not the memory modules.

Types of Memory

Memory comes in many formats. Each one has a particular set of features and characteris-tics, making it best suited for a particular application. Some decisions about the application of the memory type are based on suitability; others are based on affordability to consumers or marketability to computer manufacturers. The following list gives you an idea of the vast array of memory types and subtypes:

DRAM

- Asynchronous DRAM
- FPM DRAM
- EDO DRAM
- BEDO DRAM
- Synchronous DRAM
- SDR SDRAM
- DDR SDRAM
- DDR2 SDRAM NN
- DDR3 SDRAM
- DRDRAM

SRAM

- ROM

DRAM

DRAM is dynamic random access memory. (This is what most people are talking about when they mention RAM.) When you expand the memory in a computer, you are adding DRAM chips. You use DRAM to expand the memory in the computer because it's a cheaper type of memory. Dynamic RAM chips are cheaper to manufacture than most other types because they are less complex. *Dynamic* refers to the memory chips' need for a constant update signal (also called a refresh signal) in order to keep the information that is written there. If this signal is not received every so often, the information will bleed off and cease to exist. Currently, the most popular implementations of DRAM are based on synchronous DRAM and include SDR SDRAM, DDR, DDR2, DDR3, and DRDRAM. Before discussing these technologies, let's take a quick look at the legacy asynchronous memory types, none of which should appear on modern exams.

Asynchronous DRAM

Asynchronous DRAM (ADRAM) is characterized by its independence from the CPU's external clock. Asynchronous DRAM chips have codes on them that end in a numeri-cal value that is related to (often 1⁄10 of the actual value of) the access time of the memory. Access time is essentially the difference between the time when the information is requested from memory and the time when the data is returned. Common access times attributed to asynchronous DRAM were in the 40- to 120-nanosecond (ns) vicinity. A lower access time is obviously better for overall performance.

Because ADRAM is not synchronized to the frontside bus, you would often have to insert wait states through the BIOS setup for a faster CPU to be able to use the same mem-ory as a slower CPU. These wait states represented intervals that the CPU had to mark time and do nothing while waiting for the memory subsystem to become ready again for subse-quent access.

Common asynchronous DRAM technologies included Fast Page Mode (FPM), Extended Data Out (EDO), and Burst EDO (BEDO). Feel free to investigate the details of these par-ticular technologies, but a thorough discussion of these memory types is not necessary here. The A+ technician should be concerned with synchronous forms of RAM, which are the only types of memory being installed in mainstream computer systems today.

Synchronous DRAM

Synchronous DRAM (*SDRAM*) shares a common clock signal with the computer's system-bus clock, which provides the common signal that all local-bus components use for each step that they perform. This characteristic ties SDRAM to the speed of the FSB and hence the processor, eliminating the need to configure the CPU to wait for the memory to catch up.

Originally, *SDRAM* was the term used to refer to the only form of synchronous DRAM on the market. As the technology progressed, and more was being done with each clock signal on the FSB, various forms of SDRAM were developed. What was once called simply SDRAM needed a new name retroactively. Today, we use the term *single data rate SDRAM* (SDR SDRAM) to refer to this original type of SDRAM.

SDR SDRAM

With SDR SDRAM, every time the system clock ticks, 1 bit of data can be transmitted per data pin, limiting the bit rate per pin of SDRAM to the corresponding numerical value of the clock's frequency. With today's processors interfacing with memory using a parallel data-bus width of 8 bytes (hence the term *64-bit processor*), a 100MHz clock signal produces 800MBps. That's mega*bytes* per second, not mega*bits*. Such memory modules are referred to as PC100, named for the true FSB clock rate they rely on. PC100 was preceded by PC66 and succeeded by PC133, which used a 133MHz clock to produce 1066MBps of throughput.

Note that throughput in megabytes per second is easily computed as eight times the rating in the name. This trick works for the more advanced forms of SDRAM as well. The common thread is the 8-byte system data bus. Incidentally, you can double throughput results when implementing dual-channel memory.

DDR SDRAM

Double data rate (*DDR*) SDRAM earns its name by doubling the transfer rate of ordinary SDRAM; it does so by double-pumping the data, which means transferring a bit per pin on both the rising and falling edges of the clock signal. This obtains twice the transfer rate at the same FSB clock frequency. It's the increasing clock frequency that generates heating issues with newer components, so keeping the clock the same is an advantage. The same 100MHz clock gives a DDR SDRAM system the impression of a 200MHz clock in comparison to an SDR SDRAM system. For marketing purposes and to aid in the comparison of disparate products (DDR vs. SDR, for example), the industry has settled on the practice of using this effective clock rate as the speed of the FSB.

Because the actual system clock speed is rarely mentioned in marketing literature, on packaging, or on store shelves for DDR and higher, you can use this advertised FSB frequency in your computations for

DDR throughput. For example, with a 100MHz clock and two operations per cycle, motherboard makers will market their boards as having an FSB of 200MHz. Multiplying this effective rate by 8 bytes transferred per cycle, the data rate is 1600MBps. Because DDR made throughput a bit trickier to compute, the industry began using this final throughput figure to name the memory modules instead of the actual frequency, which was used when naming SDR modules. This makes the result seem many times better (and much more marketable), while it's really only twice (or so) as good, or close to it.

In this example, the module is referred to as PC1600, based on a throughput of 1600MBps. The chips that go into making PC1600 modules are named DDR200 for the effective FSB frequency of 200MHz. Stated differently, the industry uses DDR200 mem-ory chips to manufacture PC1600 memory modules. Let's make sure you grasp the relationship between the speed of the FSB and the name for the related chips as well as the relationship between the name of the chips (or the speed of the FSB) and the name of the modules. Consider an FSB of 400MHz, meaning an actual clock signal of 200MHz, by the way—the FSB is double the actual clock for DDR, remember. It should be clear that this motherboard requires modules populated with DDR400 chips and that you'll find such modules marketed and sold as PC3200. Let's try another. What do you need for a motherboard that features a 333MHz FSB (actual clock is 166MHz)? Well, just using the 8:1 rule mentioned earlier, you might be on the lookout for a PC2667 module. However, note that sometimes the numbers have to be played with a bit to come up with the industry's marketing terms. You'll have an easier time finding PC2700 modules that are designed specifically for a motherboard like yours, with an FSB of 333MHz. The label isn't always technically accurate, but round numbers sell better, perhaps. The important concept here is that if you find PC2700 modules and PC2667 modules, there's absolutely no difference; they both have a 2667MBps throughput rate. Go for the best deal; just make sure the memory manufacturer is reputable.

DDR2 SDRAM

Think of the *2* in *DDR2* as yet another multiplier of 2 in the SDRAM technology, using a lower peak voltage to keep power consumption down (1.8V vs. the 2.5V of DDR). Still double-pumping, DDR2, like DDR, uses both sweeps of the clock signal for data transfer. Internally, DDR2 further splits each clock pulse in two, doubling the number of operations it can perform per FSB clock cycle. Through enhancements in the electrical interface and buffers, as well as through adding off-chip drivers, DDR2 nominally produces four times the throughput that SDR is capable of producing.

Continuing the DDR example, DDR2, using a 100MHz actual clock, transfers data in four operations per cycle (effective 400MHz FSB) and still 8 bytes per operation, for a total of 3200MBps. Just like DDR, DDR2 names its chips based on the perceived frequency. In this case, you would be using DDR2-400 chips. DDR2 carries on the effective-FSB frequency method for naming modules but cannot simply call them PC3200 modules because those already exist in the DDR world. DDR2 calls these modules PC2-3200 (note the dash to keep the numeric components separate).

As another example, it should make sense that PC2-5300 modules are populated with DDR2-667 chips. Recall that you might have to play with the numbers a bit. If you multiply the well-known FSB speed of 667MHz by 8 to figure out what modules you need, you might go searching for PC2-5333 modules. You might find someone advertising such modules, but most compatible modules will be labeled PC2-5300 for the same marketability mentioned earlier. They both support 5333MBps of throughput.

DDR3 SDRAM

The next generation of memory devices was designed to roughly double the performance of DDR2 products. Based on the functionality and characteristics of DDR2's proposed successor, most informed consumers and some members of the industry surely assumed the forthcoming name would be DDR4. This

was not to be, however, and DDR3 was born. This naming convention proved that the *2* in DDR2 was not meant to be a multi-plier but instead a revision mark of sorts. Well, if DDR2 was the second version of DDR, then DDR3 is the third. *DDR3* is a memory type that was designed to be twice as fast as the DDR2 memory that operates with the same system clock speed. Just as DDR2 was required to lower power consumption to make up for higher frequencies, DDR3 must do the same. In fact, the peak voltage for DDR3 is only 1.5V.

The most commonly found range of actual clock speeds for DDR3 tends to be from 133MHz at the low end to less than 300MHz. Because double-pumping continues with DDR3 and because four operations occur at each wave crest (eight operations per cycle), this frequency range translates to common FSB implementations from 1066MHz to more than 2000MHz in DDR3 systems. These memory devices are named following the conventions established earlier. Therefore, if you buy a motherboard with a 1600MHz FSB, you know immediately that you need a memory module populated with DDR3-1600 chips because the chips are always named for the FSB speed. Using the 8:1 module-to-chip/FSB naming rule, the modules you need would be called PC3-12800, supporting a 12800MBps throughput.

The earliest DDR3 chips, however, were based on a 100MHz actual clock signal, so we can build on our earlier example, which was also based on an actual clock rate of 100MHz. With eight operations per cycle, the FSB on DDR3 motherboards is rated at 800MHz, quite a lot of efficiency while still not needing to change the original clock our examples began with. Applying the 8:1 rule again, the resulting RAM modules for this motherboard are called PC3-6400 and support a throughput of 6400MBps, carrying chips called DDR3-800, again named for the FSB speed.

DRDRAM

Direct Rambus DRAM (DRDRAM), named for *Rambus*, the company that designed it, is a legacy proprietary SDRAM technology, sometimes called RDRAM, dropping *direct*, and most often associated with server platforms. Although other specifications preceded it, the first motherboard DRDRAM model was known as PC800. As with non-DRDRAM speci-fications that use this naming convention, PC800 specifies that, using a faster 400MHz actual clock signal and double-pumping like DDR SDRAM, an effective frequency and FSB speed of 800MHz is created.

This original naming of DRDRAM modules was based on FSB speed, in a dissimilar fash-ion to other forms of SDRAM after SDR, which are named for their throughput in MBps. You might recall, for those memory types, that the FSB speed was used to name the actual chips on

the modules, not the modules themselves. PC800 DRDRAM, then, features a double-pumped 800MHz FSB. Newer modules, however, such as the 32-bit RIMM 6400, are named for their actual throughput, 6400MBps, in this case. The section “RIMM” later in this chapter details the physical details of the modules.

There are only 16 data pins per channel with DRDRAM, versus 64 bits per channel in other SDRAM implementations. This fact results in a 16-bit (2-byte) channel. A 2-byte packet, therefore, is exchanged during each read/write cycle, bringing the overall transfer rate of PC800 DRDRAM to 1600MBps per channel. DRDRAM chipsets require two 16-bit channels to communicate simultaneously for the same read/write request, creating a mandatory 32-bit dual-channel mode. Two PC800 DRDRAM modules in a dual-channel configuration produce transfer rates of 3200MBps. In motherboards that support 32-bit modules, you would use a single RIMM 3200 to achieve this same 3200MBps of throughput, using the same actual 400MHz clock and 800MHz FSB and transferring 4 bytes (32 bits) at a time.

Despite DRDRAM's performance advantages, it has some drawbacks that kept it from taking over the market in its day. Increased latency, heat output, complexity in the manufacturing process, and cost are the primary shortcomings. The additional heat that individual DRDRAM chips put out led to the requirement for heat sinks on all modules. High manufacturing costs and high licensing fees led to tripling the cost to consumers over SDR. Soon, other SDRAM technologies obviated the need to specialize with DRDRAM. A dual-channel platform using standard PC3200 DDR modules transfers 16 bytes (128 bits) per read/write request, producing the same throughput rate of 6400MBps as high-end RIMM 6400 modules. As a result, and because of the eventual advent of DDR2 and DDR3, DRDRAM no longer held any performance advantage.

To put each of the SDRAM types into perspective, consult Table 1.2, which summarizes how each technology in the SDRAM arena would achieve a transfer rate of 3200MBps, even if only theoretically. For example, PC400 doesn't exist in the SDR SDRAM world.

TABLE 1.2: How some memory types transfer 3200MBps per channel

Memory Type	Actual/Effective (FSB) Clock Frequency (MHz)	Bytes per Transfer
SDR SDRAM PC400*	400/400	8
DDR SDRAM PC3200	200/400	8
DDR2 SDRAM PC2-3200	100/400	8
DDR3 SDRAM PC3-3200**	50/400	8
DRDRAM PC800	400/800	4***

* SDR SDRAM PC400 does not exist.

**PC3-3200 does not exist and is too slow for DDR3.

***Assuming requisite 32-bit dual-channel mode.

SRAM

Static random access memory (SRAM) doesn't require a refresh signal like DRAM does. The chips are more complex and are thus more expensive. However, they are considerably faster. DRAM access times come in at 40 nanoseconds (ns) or more; SRAM has access times faster than 10ns. SRAM is classically used for cache memory.

ROM

ROM stands for read-only memory. It is called read-only because the original form of this memory could not be written to. Once information had been etched on a silicon chip and manufactured into the ROM package, the information couldn't be changed. If you ran out of use for the information or code on the ROM, you added little eyes and some cute fuzzy extras and you had a bug that sat on your desk and looked back at you. Some form of ROM is normally used to store the computer's BIOS because this information normally does not change very often.

The system ROM in the original IBM PC contained the power-on self-test (POST), BIOS, and cassette BASIC. Later IBM computers and compatibles include everything but the cassette BASIC. The system ROM enables the computer to "pull itself up by its boot-straps," or *boot* (find and start the operating system).

Through the years, different forms of ROM were developed that could be altered, later ones more easily than earlier ones. The first generation was the programmable ROM (PROM), which could be written to for the first time in the field using a special programming device, but then no more. You had a new bug to keep the ROM bug company. Liken this to the burn-ing of a CD-R. Don't need it any longer? You've got a handy coaster. Following the PROM came erasable PROM (EPROM), which was able to be erased using ultraviolet light and subse-quently reprogrammed using the original programming device. These days, our flash memory is a form of electronically erasable PROM (EEPROM), which does not require UV light to erase its contents but rather a slightly higher than normal electrical pulse.

Memory Packaging

First of all, it should be noted that each motherboard supports memory based on the speed of the frontside bus (or the CPU's QPI) and the memory's form factor. For example, if the motherboard's FSB is rated at a maximum speed of 1333MHz and you install memory that is rated at 1066MHz, the memory will operate at only 1066MHz, if it works at all, thus making the computer operate slower than it could. In their documentation, most mother-board manufacturers list which type(s) of memory they support as well as its maximum speeds and required pairings.

The memory slots on a motherboard are designed for particular module form factors or styles. RAM historically evolved from form factors no longer seen for such applications, such as dual inline package (DIP), single inline memory module (SIMM), and single inline pin package (SIPP). The most popular form factors for primary memory modules today are as follows:

DIMM

RIMM

SODIMM NN MicroDIMM

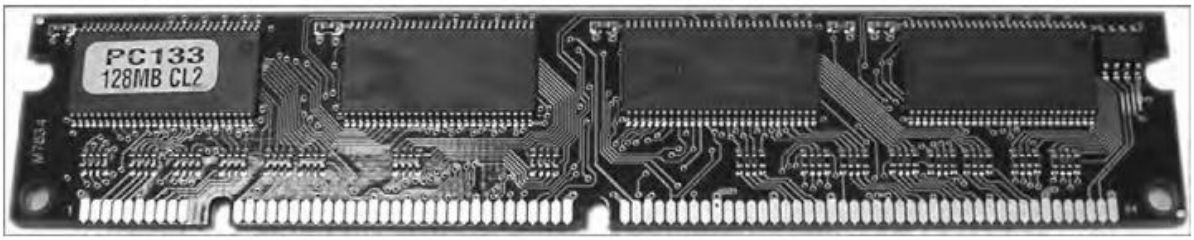
Note also that the various CPUs on the market tend to support only one form of physical memory packaging due to the memory controller in the Northbridge or CPU itself (QPI). For example, the Intel Pentium 4 class of processors was always paired with DIMMs, while certain early Intel Xeon processors

mated only with RIMMs. Laptops and smaller devices require SODIMMs or smaller memory packaging. So, in addition to coordinating the speed of the components, their form factor is an issue that must be addressed.

DIMM

One type of memory package is known as a DIMM. As mentioned earlier in this chapter, DIMM stands for dual inline memory module. DIMMs are 64-bit memory modules that are used as a package for the SDRAM family: SDR, DDR, DDR2, and DDR3. The term *dual* refers to the fact that, unlike their SIMM predecessors, DIMMs differentiate the functionality of the pins on one side of the module from the corresponding pins on the other side. With 84 pins per side, this makes 168 independent pins on each standard SDR module, as shown with its two keying notches as well as the last pin labeled 84 on the side shown in Figure 1.20.

FIGURE 1.20 An SDR dual inline memory module (DIMM)



The DIMM used for DDR memory has a total of 184 pins and a single keying notch, while the DIMM used for DDR2 has a total of 240 pins, one keying notch, and possibly an aluminum cover for both sides, called a *heat spreader* and designed like a heat sink to dissipate heat away from the memory chips and prevent overheating. The DDR3 DIMM is similar to that of DDR2. It has 240 pins and a single keying notch, but the notch is in a different location to avoid cross insertion. Not only is the DDR3 DIMM physically incom-patible with DDR2 DIMM slots, it's also electrically incompatible.

Figure 1.21 is a photo of a DDR2 module. A matched pair of DDR3 modules with heat spreaders, suitable for dual-channel use in a high-end graphics adapter or motherboard, is shown in Figure 1.22.

FIGURE 1.21 A DDR2 SDRAM module

FIGURE 1.22 A pair of DDR3 SDRAM modules

RIMM

Assumed to stand for Rambus inline memory module but not really an acronym, RIMM is a trademark of Rambus Inc. and perhaps a clever play on the acronym DIMM, a competing form factor and by definition, what a RIMM actually is. A RIMM is a custom memory mod-ule that carries DRDRAM and varies in physical specification, based on whether it is a 16-bit or 32-bit module. The 16-bit modules have 184 pins and two keying notches, while 32-bit modules have 232 pins and only one keying notch, reminiscent of the trend in SDRAM-to-DDR evolution. Figure 1.23 shows a RIMM module, including the aluminum heat spreaders.

FIGURE 1.23 A Rambus RIMM module

As mentioned earlier, DRDRAM is based on a 16-bit channel. However, dual-channel implementation is not optional with DRDRAM; it's required. The dual-channel architecture can be implemented utilizing two separate 16-bit RIMMs (leading to the generally held view that RIMMs must always be installed in pairs) or the newer 32-bit single-module design.

Typically, motherboards with the 16-bit single- or dual-channel implementation provide four RIMM slots that must be filled in pairs, while the 32-bit versions provide two RIMM slots that can be filled one at a time. A 32-bit RIMM essentially has two 16-bit modules built in (possibly contributing to the persistence of the belief in the "pair" requirement) and requires only a single motherboard slot, albeit a physically different slot. So you must be sure of the module your motherboard accepts before upgrading.

Unique to the use of RIMM modules, a computer must have every RIMM slot occupied. Even one vacant slot will cause the computer not to boot. Any slot not populated with live memory requires an inexpensive blank of sorts called a continuity RIMM, or C-RIMM, for its role of keeping electrical continuity in the DRDRAM channel until the signal can terminate on the motherboard. Think of it like a fusible link in a string of holiday lights.

It seems to do nothing, but no light works without it. However, 32-bit modules terminate themselves and do not rely on the motherboard circuitry for termination, so vacant 32-bit slots require a module known as a continuity and termination RIMM (CT-RIMM).

SODIMM

Notebook computers and other computers that require much smaller components don't use standard RAM packages, such as the DIMM. Instead, they call for a much smaller memory form factor, such as a small outline DIMM. SODIMMs are available in many physical imple-mentations, including the older 32-bit (72- and 100-pin) configuration and newer 64-bit (144-pin SDR SDRAM, 200-pin DDR/DDR2, and 204-pin DDR3) configurations.

All 64-bit modules have a single keying notch. The 144-pin module's notch is slightly off-center. Note that although the 200-pin SODIMMs for DDR and DDR2 have slightly differ-ent keying, it's not so different that you don't need to pay close attention to differentiate the two. They are not, however, interchangeable. Figure 1.24 shows an example of a 144-pin, 64-bit SDR module. Figure 1.25 is a photo of a 200-pin DDR2 SODIMM.

FIGURE 1.24 144-pin SODIMM

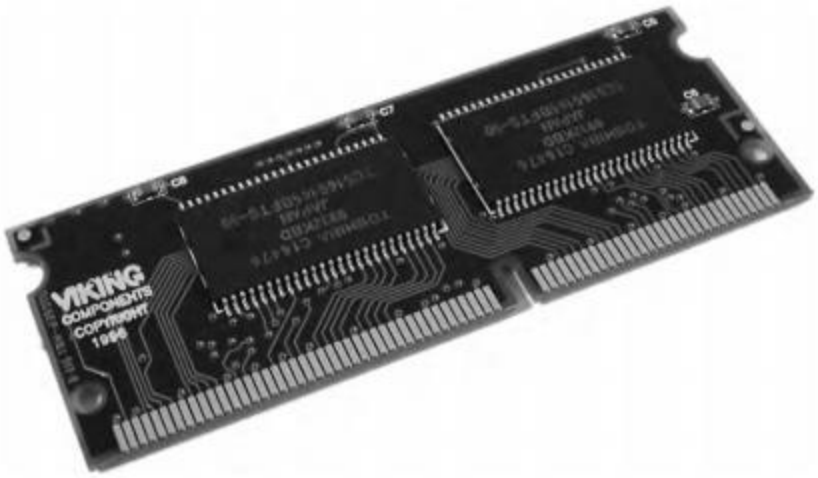


FIGURE 1.25 200-pin DDR2 SODIMM

MicroDIMM

A newer, smaller, and rarer RAM form factor is the MicroDIMM. The MicroDIMM is an extremely small RAM form factor. In fact, it is over 50 percent smaller than a SODIMM, only 45.5 millimeters (about 1.75 inches) long and 30 millimeters (about 1.2 inches—a bit bigger than a quarter) wide. It was designed for the ultralight and portable subnotebook style of computer. Standard versions of these modules have 144 pins for SDR SDRAM,

172 pins for DDR DRAM, and 214 pins for DDR2 SDRAM. MicroDIMMs are similar to a DIMM in that they use a 64-bit data bus. The insertion keying of the MicroDIMM for card-edge versions is reminiscent of the SIMM; only one notch and on one of the two inser-tion corners of the module instead of somewhere in the middle. Figure 1.26 shows an art-ist's rendering of a MicroDIMM module. Often employed in laptop computers, SODIMMs and MicroDIMMs are mentioned in Chapter 9 as well.

FIGURE 1.26 172-pin MicroDIMM