# Computer Architecture

# SYSTEM BUS AND BUS ARBITRATION

PROFESSOR SAMIR ABOU EL-SEOUD

# System bus

- In computer architecture, a bus is a collection of wires through which data is transmitted from one part of a computer to another. This is a bus that connects all the internal computer components to the CPU and main memory.
- The size of a bus, known as its *width,* is important because it determines how much data can be transmitted at one time. For example, a 16-bit bus can transmit 16 bits of data, whereas a 32-bit bus can transmit 32 bits of data.
- Every bus has a clock speed measured in MHz. A fast bus allows data to be transferred faster, which makes applications run faster. On PCs, the old ISA bus is being replaced by faster buses such as PCI.

# System Bus

In computer architecture, a bus is a collection of wires through which data is transmitted from one part of a computer to another. You can think of a bus as a highway on which data travels within a computer. When used in reference to personal computers, the term *bus* usually refers to *internal bus or system bus*. This is a bus that connects all the internal computer components to the CPU and main memory. There's also an expansion bus that enables expansion boards to access the CPU and memory.

The size of a bus, known as its *width,* is important because it determines how much data can be transmitted at one time. For example, a 16-bit buses can transmit 16 bits of data, whereas a 32-bit buses can transmit 32 bits of data.

Every bus has a clock speed measured in MHz. A fast bus allows data to be transferred faster, which makes applications run faster. On PCs, the old ISA bus is being replaced by faster buses such as PCI

Nearly all PCs made today include a local bus for data that requires especially fast transfer speeds, such as video data. The local bus is a high-speed pathway that connects directly to the processor. Several different types of buses are used on Apple Macintosh computers. Older Macs use a bus called NuBus, but newer ones use PCI.

A system bus is a single computer bus that connects the major components of a computer system.

# System bus

- All system buses consist of three parts – a data bus, an address bus, and a control bus. The data bus transfers actual data whereas the address bus transfers information about where the data should go. The control bus carries commands from the CPU and returns status signals from the devices.

# Data Bus

A data bus to carry data between the CPU and its random-access memory. It is an internal pathway across which data are transferred to and from the processor or to and from memory. The width and clock rate (the fundamental rate in cycles per second at which a computer performs its most basic operations such as adding two numbers or transferring a value from one register to another) of the data bus determine its data rate (the number of bytes per second it can carry), which is one of the main factors determining the processing power of a computer. Most current processor designs use a 32-bit bus, meaning that 32 bits of data can be transferred at once. Some processors have an internal data bus which is wider than their external bus in order to make external connections cheaper while retaining some of the benefits in processing power of a wider bus.

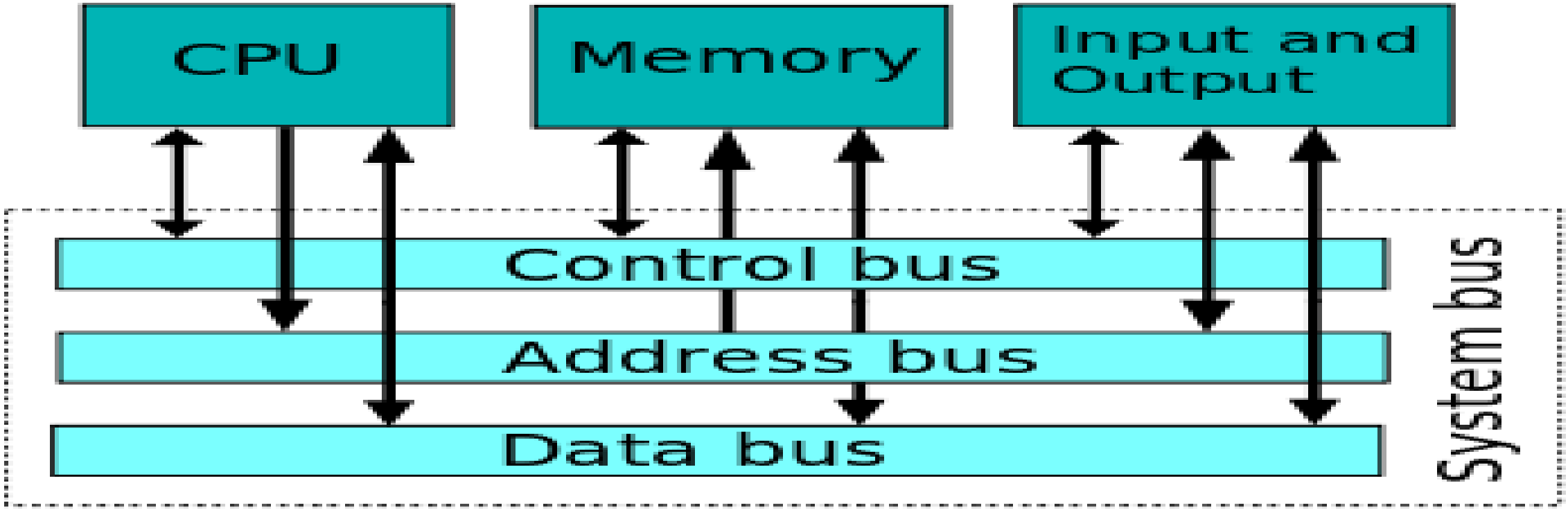
# Control Bus

A control bus to determine its operation. A control bus is used by CPUs for communicating with other devices within the computer. While the address bus carries the information on which device the CPU is communicating with and the data bus carries the actual data being processed, the control bus carries commands from the CPU and returns status signals from the devices, for example if the data is being read or written to the device the appropriate line (read or write) will be active (logic zero).

# Address Bus

An address bus to determine where it should be sent. An address bus is a series of lines connecting two or more devices that is used to specify a physical address. When a processor needs to read or write to a memory location, it specifies that memory location on the address bus (the value to be read or written is sent on the data bus). The width of the address bus determines the amount of memory a system can address. For example, a system with a 32-bit address bus can address 232 (4,294,967,296) memory locations. If each memory address holds one byte, the addressable memory space is 4 GB.

# System bus


CPU
Memory
Input and Output
Control bus
Address bus
Data bus
System bus

Generally speaking, in computer architecture, a bus is a subsystem that transfers data between computer components inside a computer or between computers. Unlike a point-to-point connection, a bus can logically connect several peripherals over the same set of wires. Each bus defines its set of connectors to physically plug devices, cards or cables together.

# Bus Arbitration

In single bus architecture when more than one device requests the bus, a controller called bus arbiter decides who gets the bus, this is called the bus arbitration. Arbitration is mostly done in favor of a master micro-processor with the highest priority. Briefly, bus arbitration regulates the traffic on the bus to prevent two devices from trying to use it at the same time.

It is common to build computer systems with multiple active devices attached to the same memory bus. The most obvious example may be a machine with multiple central processors, but it is equally reasonable to speak in terms of secondary processors such as direct-memory-access (DMA) peripheral devices. Say, we have two CPU and two DMA devices.

Each of these may independently access RAM. The CPU initiate DMA transfers to or from disk, but once initiated, the disk controller transfers the data between the disk and RAM on its own, without involving the CPU in each transfer. The DMA video controller operates continuously, reading a new pixel from RAM every microsecond and converting it to voltages on the red, green and blue signals sent to a color CRT.

We can view this system as having the simplest form of multiport memory, one where there is in actuality only one memory port that is time multiplexed between the various clients.

The problem this poses is how these different devices can share the same bus connecting them to memory. Up to this point, we have spoken of memory busses with the assumption that a single device, the bus master, provided the memory address and memory control signals. Now, we assume bus where any of several devices may control the bus, and they must share it in an orderly way!

We solve this problem with bus arbitration logic. This logic may be centralized or distributed, and it may operate synchronously or asynchronously. Distributed asynchronous bus arbitration is very hard, while centralized synchronous arbitration is fairly easy.