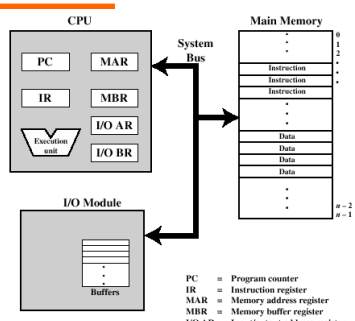
Computer Systems Organization

Von Neumann Architecture

- **Data and instructions are stored in a single read-write memory**
- **The content of this memory are addressable by location, without regard to the type of data contained in it.**
- **Execution occurs in a sequential fashion (unless explicitly modified) from one instruction to the next.**

Computer Components

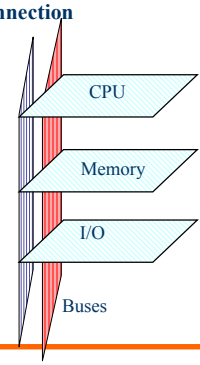
- A computer consists of a set of modules of three basic types:
- They communicate with each other
- Needs connection paths for connecting the modules



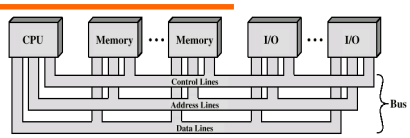
PC = Program counter
IR = Instruction register
MAR = Memory address register
MBR = Memory buffer register
I/O AR = Input/output address register
I/O BR = Input/output buffer register

Bus Interconnection

- A bus is a communication path that connects two or more devices
- A bus is a shared transmission medium.
- A signal transmitted by one device is available for reception by all other devices attached to the bus
- If two devices transmit during the same period, their signal will overlap and become garbled.
- Only one device can successfully transmit at any one time.

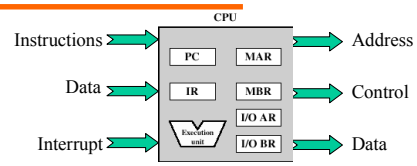


Bus Interconnection



- Data Lines (Data Bus, DB)
- Address Lines (Address Bus, AB)
- Control Lines (Control Bus, CB): Control the access to and the use of DB and AB. (remember AB and DB shared by all devices)
- CB send out both command and timing signals
  - Command: specify the type of operation (R/W)
  - Timing: Indicate the validity of data on DB and AB
- Typical control lines include: Memory R/W, I/O R/W, Bus request, Bus grant, Interrupt request, Interrupt grant, Transfer ACK

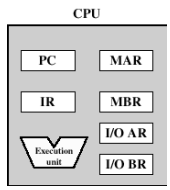
Computer Components: CPU module



- Read instructions and data
- Write out data after processing
- Use control signal to control the overall operation of the computer
- Receive interrupt signal

Computer Components: CPU module

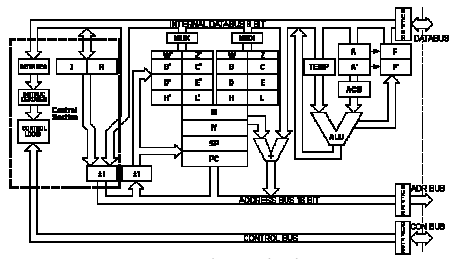
- Registers
  - Permanent storage locations within the CPU
  - Each used for a particular, defined purpose
- Accumulator
  - General purpose register
- Registers in the Control Unit
  - PC - program counter register
  - IR - Instruction register
  - MAR - memory address register
  - MBR - memory buffer register
  - I/O AR - I/O address register
  - I/O BR - I/O buffer register
  - Status register



Move data

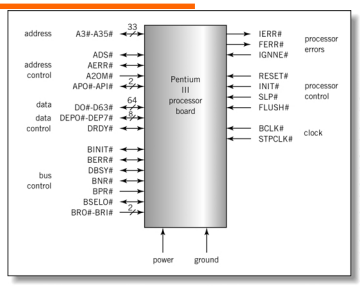
Manipulate data

Computer Components: CPU module



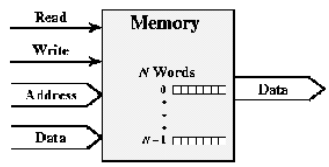
Z80 Internal Organization

Computer Components: CPU module



Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition
Chapter 7, Figure 07-13

Computer Components: Memory module



- N words of equal length
- Each word with a unique address (0, 1, …, N-1)
- A word of data can be read from or write into the memory
- The nature of the the operation (R/W) is indicated by read and write control signals
- The location for operation is specified by an address

Memory System Architecture

- Primary memory is a collection of independent storage units. Each unit stores a single multi-bit value.
- The number of bits in a storage unit is a constant for all storage units in the memory system, and this constant is called the memory width.
- Addresses are used to access the storage units in the memory system. Each storage unit has a unique address.

Memory System Architecture

A graphical representation of a memory with 128 storage units and width 8.

The memory is called 128 x 8 memory.

Address	Memory Content
1	10101110
2	11110101
3	11011000
•	•
•	•
•	•
128	10101111

Memory System Architecture

Manufacturers produce a number of different types of memory devices having a variety of technologies.

The technology affects not only the operating characteristics, such as power consumption, size, and speed, but also the manufacturing cost.

Thus in the selection of memory chips for a particular application, designers must weigh the trade-offs between cost and performance.

Memory System Architecture

Read-only memory

Read-only memories (ROMs) are memory devices that the CPU can read but cannot write.

Many ROMs are factory programmed and there is no way to alter their contents (the term programming here means writing values into a ROM). These devices are denser and cheaper to manufacture than other type of ROM.

Programmable ROMs (*PROMs*): This type of ROM can be programmed by using special high current device to destroy (burn) the fuse that were manufactured into the device. The result of burning a PROM is that certain bits are always 0 and the rest are always 1. These values cannot be altered once written.

Erasable PROMs (*EPROMs*): This type of ROM is alterable, although not during ordinary use. A technician can program an EPROM off line, later completely erase its contents by using ultraviolet light, and then reprogram it.

Memory System Architecture

Read/Write Memory

Read/Write memories refer to memory devices can be read from and write into with equal ease. Two main types of read/write memory devices are **static** random access memories (SRAMs) and **dynamic** random access memories (DRAMs).

SRAMs: In SRAMs, the individual memory contents, once written, do not need to be further addressed or manipulated to hold their values. These devices are composed of flip-flops that use a small current to maintain their contents. SRAMs are used mostly in CPU registers and other high speed storage devices. Some computers use them for **cache** and main memory. SRAMs are currently the fastest and most expensive of semiconductor memory circuit.

DRAMs: These are semiconductor memory devices in which the stored data will not remain permanently stored, even with power applied, unless the data are periodically rewritten into the memory. The latter operation is called the *refresh* operation. Although much cheaper than SRAMs, DRAMs are also slower and used mostly for main memory

Memory System Architecture

General Memory Operation

Although each type of memory is different in its internal operation, there are certain basic operating principles that are the same for all memory systems. Every memory system requires several different types of input and output lines to perform the following function:

Select the address in memory that is being accessed for READ or WRITE operation

Select either READ or WRITE operation to be performed

Supply the input data to be stored in memory during write operation

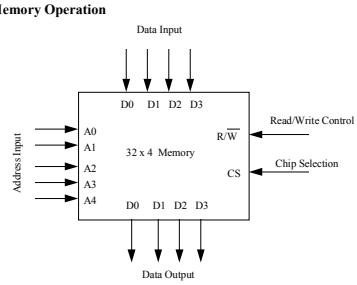
Hold the output data coming from memory during a read operation

Enable (or disable) the memory so that it will (or will not) respond to the address inputs and read/write command.

Memory System Architecture

General Memory Operation

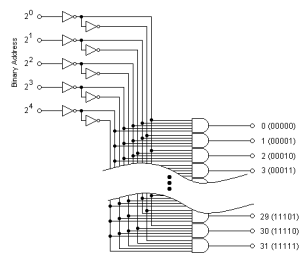
.



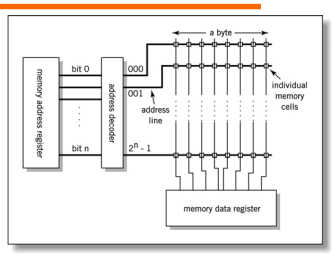
Memory System Architecture

Address Decoder

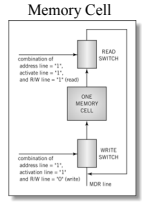
.



Computer Components: Memory module



Memory Cell



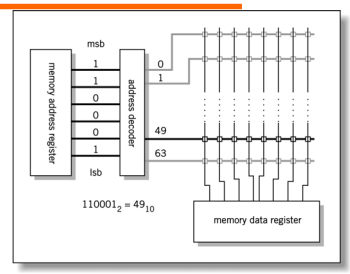
Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition Chapter 7, Figure 07-07

Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition Chapter 7, Figure 07-04

Address Decoder

The relationship between the MDR, the MAR, and memory

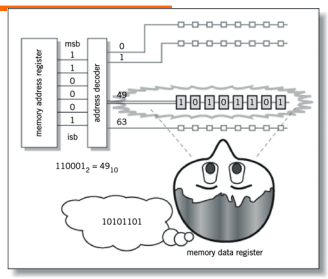
Computer Components: Memory module



Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition Chapter 7, Figure 07-05

MAR-MDR example

Computer Components: Memory module



Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition Chapter 7, Figure 07-06

Computer Components: Memory module

Memory Address Space and Memory Map

The total amount of memory contained in any system is limited by the size of the address bus.

Example: *A 6502 processor has 16-bit address bus, what is the maximum amount of memory which a system can utilised?*

Computer Components: Memory module

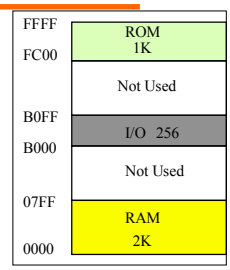
Memory Map

The microcomputer designer has to allocate this address space among the RAM, ROM, and I/O devices that are to be part of the system.

The manner in which the total address space is apportioned among these devices depends, to certain extent, on characteristics of the processor.

A **memory map** is a simple diagram which identifies the size and location of any memory block in the total address space.

Computer Components: Memory module



A typical memory map for a microprocessor system with 16-bit address bus

Computer Components: Memory module

IBM PC Main Memory Map

Main memory, also called conventional memory, refers to the storage locations that the CPU can reference during an ordinary memory-read or memory-write bus cycle without special hardware.

The amount of main memory a PC could directly address to is 1MB (2^{20} bytes).

The PC architects divided the address space of conventional memory into a number of blocks, which they allocated for various software components.

They allocated the largest block, with addresses ranging from 0K to 640K, to program memory, which they implemented with DRAM chips.

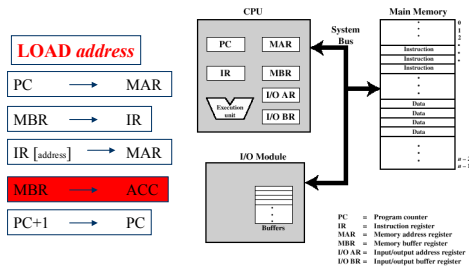
They reserved the remaining block, with address ranging from 640K and 1024K, for ROM BIOS and other system components. The following table summarises the allocation of addresses within conventional memory of a PC.

Computer Components: Memory module

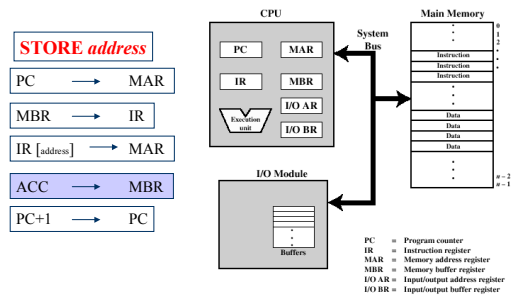
Address	PC Usage
960K - 1024K	ROM BIOS
880K - 960K	Unused
848K - 880K	LIM data area
816K - 848K	LIM data area
800K - 816K	Hard disk ROM
784K - 800K	Unused
768K - 784K	EGA ROM
752K - 768K	Unused
736K - 752K	CGA
720K - 736K	Unused
704K - 720K	MDA
640K - 704K	EGA or VGA

1536 - 640K	User RAM
1152 - 1535	BASIC, Special system RAM
0 - 1023	Interrupt-vector table

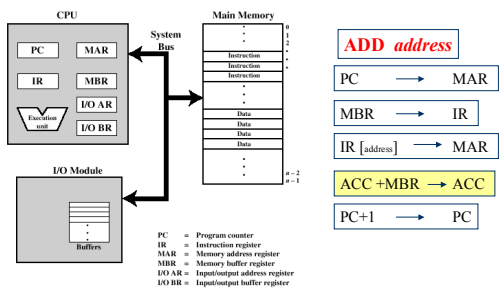
The Fetch-Execute Instruction Cycle



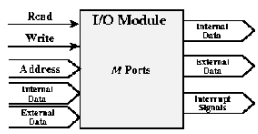
The Fetch-Execute Instruction Cycle



The Fetch-Execute Instruction Cycle



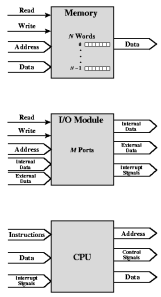
Computer Components: I/O module



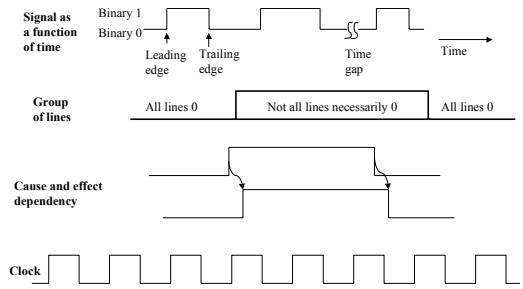
- From an internal (to the computer system) point of view, I/O is functionally similar to memory
- There two operations read and write
- I/O module may control more than one devices.
- The interfaces to each external devices is referred to as port, and each port is given a unique address
- External data path for input output data with external devices
- I/O module may be able to send interrupt signals to the CPU

Computer Components: Data Exchange

- Memory to processor
- Processor to memory
- I/O to processor
- Processor to I/O
- I/O to and from Memory (DMA)

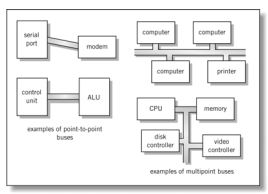


Timing Diagram

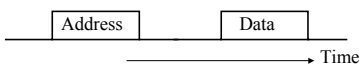


Bus Types

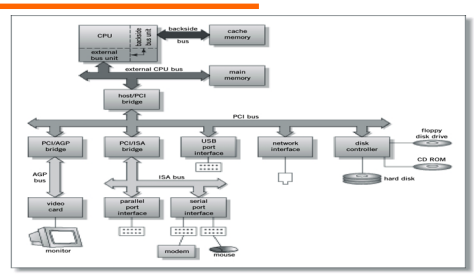
- Dedicated
  - Physical:
    - Connected to a subset of modules
  - Functional:
    - Data bus, Address Bus
- Multiplexed:
  - Time multiplexing



Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition Chapter 7, Figure 07-08



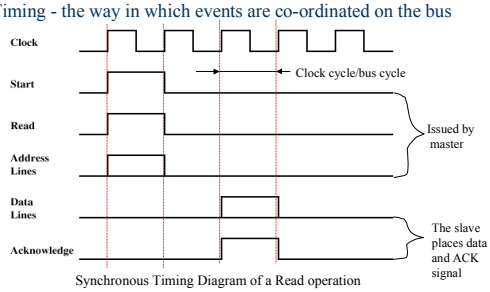
Bus Configuration Examples



Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition Chapter 7, Figure 07-09

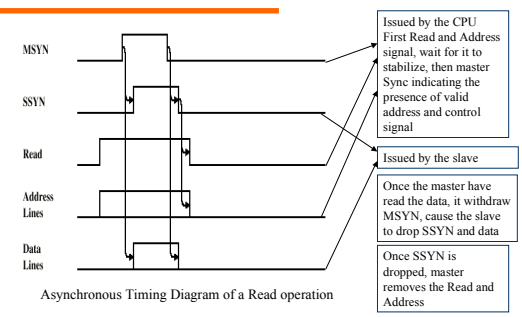
Timing: Synchronous

Timing - the way in which events are co-ordinated on the bus



Synchronous Timing Diagram of a Read operation

Timing: Asynchronous



Asynchronous Timing Diagram of a Read operation

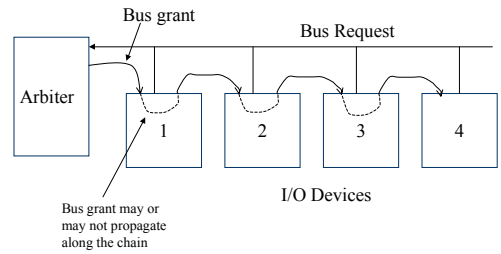
Bus Arbitration

- More than one module (e.g. CPU and DMA controller) may need control of the bus
- Only one module may control bus at one time
- Needs some form of arbitration

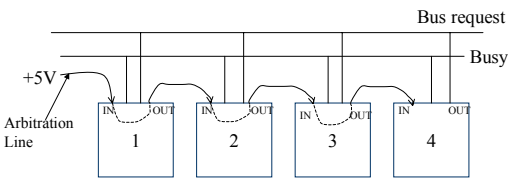
- Centralised Arbitration
  - Single hardware device controlling bus access (Bus Controller or Arbiter)
  - May be a separate module or part of CPU or separate
- Distributed Arbitration
  - Each module may claim the bus
  - Control logic on all modules

One device is designated as master, which may initiate a data transfer with some other device (slave)

Bus Arbitration : Centralized



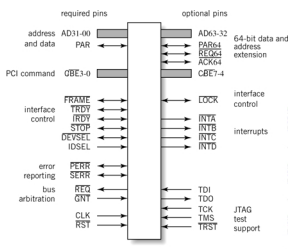
Bus Arbitration : Decentralized



PCI Bus

- Peripheral component interconnect
- Start development 1990
- Became standard 1995
- Used in
  - Sun Workstations
  - Apple Macintosh
  - Wintel PCs
  - Compaq Alpha Server
- The same peripheral I/O cards may be plugged into many different computers

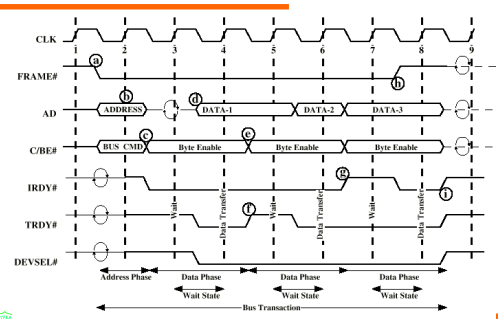
PCI Bus



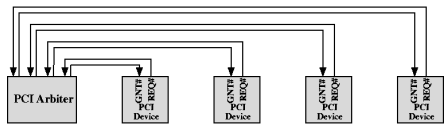
Englander: The Architecture of Computer Hardware and Systems Software, 2nd edition
Chapter 7, Figure 07-12

PCI bus connections ***Source:*** **Copyright © PCI Pin List/PCI Special Interest Group, 1999.**

PCI Bus: Operation Example - Read



PCI Bus: Arbitration



Centralized synchronous arbitration scheme