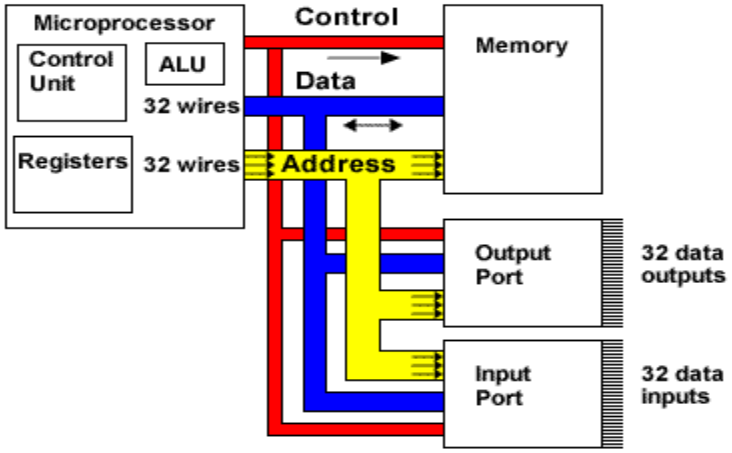
System Buses



Lecture 7

Buses

- There are a number of possible interconnection systems
- Single and multiple BUS structures are most common
- e.g. Control/Address/Data bus (PC)

What is a Bus?

- A communication pathway connecting two or more devices
- Usually broadcast
- Often grouped
  - A number of channels in one bus
  - e.g. 32 bit data bus is 32 separate single bit channels

A **system bus** is a single **computer bus** that connects the major components of a computer system. The technique was developed to reduce costs and improve modularity. It combines the functions of a **data bus** to carry information, an **address bus** to determine where it should be sent, and a **control bus** to determine its operation.

Data Bus

- Carries data
  - Remember that there is no difference between "data" and "instruction" at this level
- Width is a key determinant of performance
  - 8, 16, 32, 64 bit

Address bus

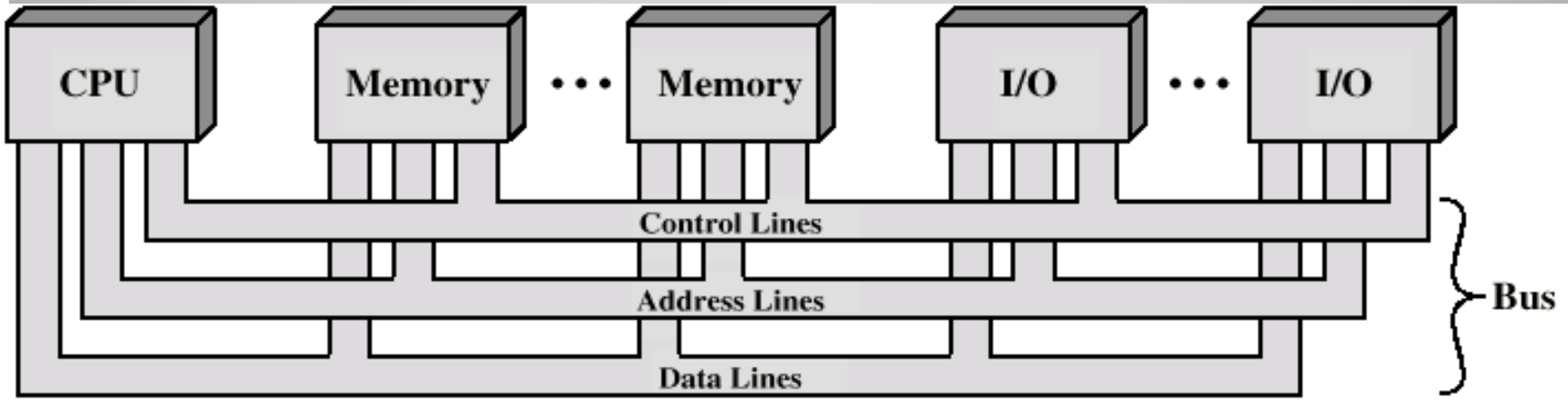
- Identify the source or destination of data
- e.g. CPU needs to read an instruction (data) from a given location in memory
- Bus width determines maximum memory capacity of system
  - e.g. 8080 has 16 bit address bus giving 64k address space

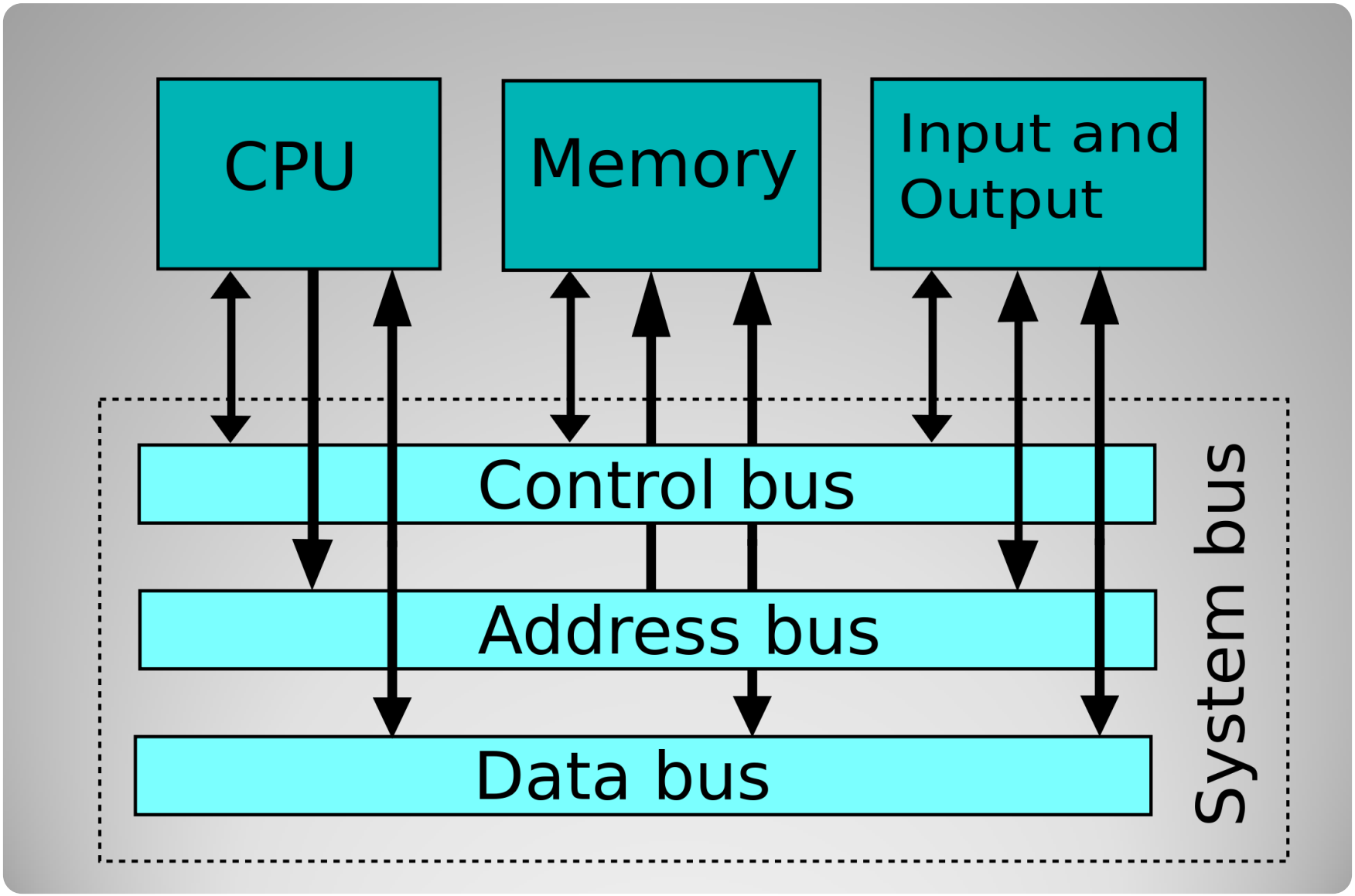
Control Bus

- Control and timing information
  - Memory read/write signal
  - Interrupt request
  - Clock signals

An **interrupt** is a signal from a device attached to a computer or from a program within the computer that causes the main program that operates the computer to stop and figure out what to do next.

Bus Interconnection Scheme




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

Big and Yellow?

- What do buses look like?
  - Parallel lines on circuit boards
  - Ribbon cables
  - Strip connectors on mother boards
    - e.g. PCI
  - Sets of wires

Single Bus Problems

- Lots of devices on one bus leads to:
  - Propagation delays
- Most systems use multiple buses to overcome these problems

Bus Types

- Dedicated
  - Separate data & address lines
- Multiplexed
  - Shared lines
  - Address valid or data valid control line
  - Advantage - fewer lines
  - Disadvantages
    - More complex control

Timing

- Coordination of events on bus
- Synchronous
  - Events determined by clock signals
  - Control Bus includes clock line
  - A single 1-0 is a bus cycle
  - All devices can read clock line
  - Usually a single cycle for an event

PCI Bus

- Peripheral Component Interconnection
- Intel released to public domain
- 32 or 64 bit
- 50 lines

PCI Bus Lines

- Systems lines
  - Including clock and reset
- Address & Data
  - 32 time mux lines for address/data
  - Interrupt & validate lines
- Interface Control
- Error lines

PCI Bus Lines (Optional)

- Interrupt lines
  - Not shared
- Cache support
- 64-bit Bus Extension
  - Additional 32 lines
  - Time multiplexed
  - 2 lines to enable devices to agree to use 64-bit transfer
- JTAG/Boundary Scan
  - For testing procedures

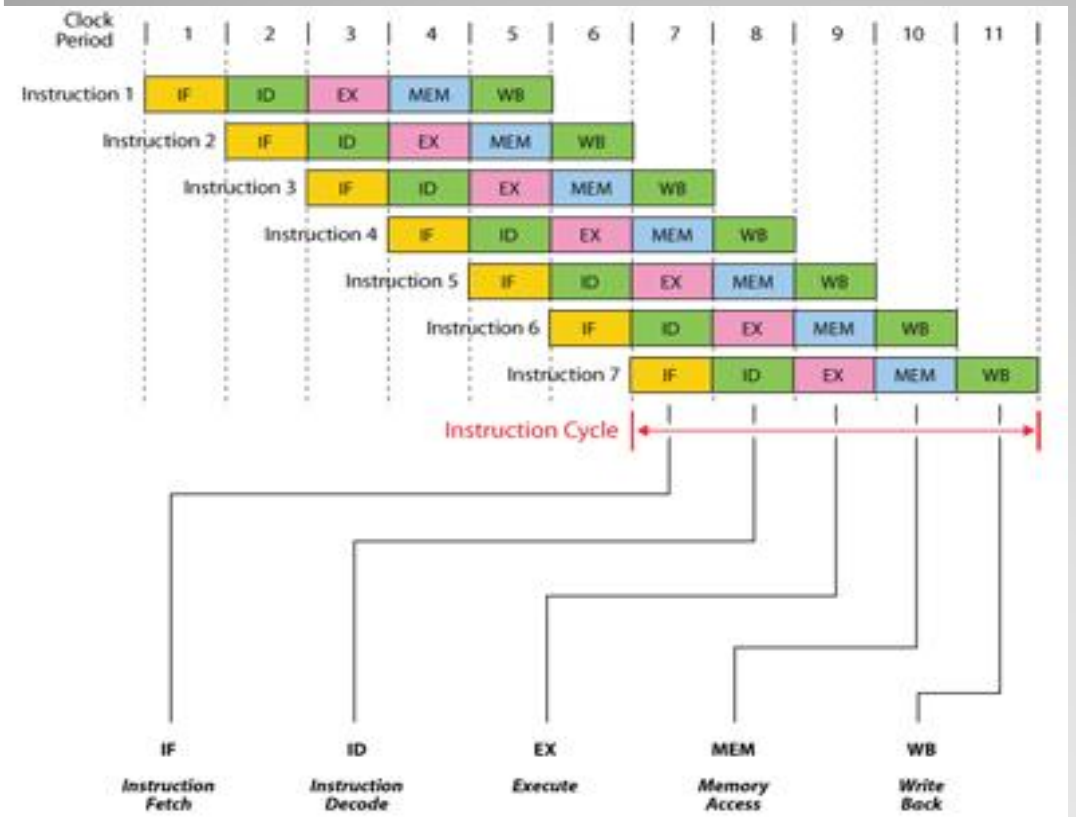
PCI Commands

- Transaction between initiator (master) and target
- Master claims bus
- Determine type of transaction
  - e.g. I/O read/write
- Address phase
- One or more data phases

Von Neumann and Harvard architectures

- **Von Neumann**
  - Allows instructions and data to be mixed and stored in the same memory module
  - More flexible and easier to implement
  - Suitable for most of the general purpose processors
- **Harvard:**
  - Uses separate memory modules for instructions and for data
  - It is easier to pipeline and there are no memory alignment problems
  - Higher memory throughput
  - Suitable for DSP (Digital Signal Processors)

Pipeline



Pipeline Stages

1. IF "Instruction Fetch"
2. ID "Instruction Decode"
3. EX "Execute"
4. MEM "Memory Access"
5. WB "Write Back"

Instr. No.	Pipeline Stage						
1	IF	ID	EX	MEM	WB		
2		IF	ID	EX	MEM	WB	
3			IF	ID	EX	MEM	WB
4				IF	ID	EX	MEM
5					IF	ID	EX
Clock Cycle	1	2	3	4	5	6	7

Pipelining is an implementation technique where multiple instructions are overlapped in execution. The computer pipeline is divided in **stages**. Each stage completes a part of an instruction in parallel.