Recap from last class

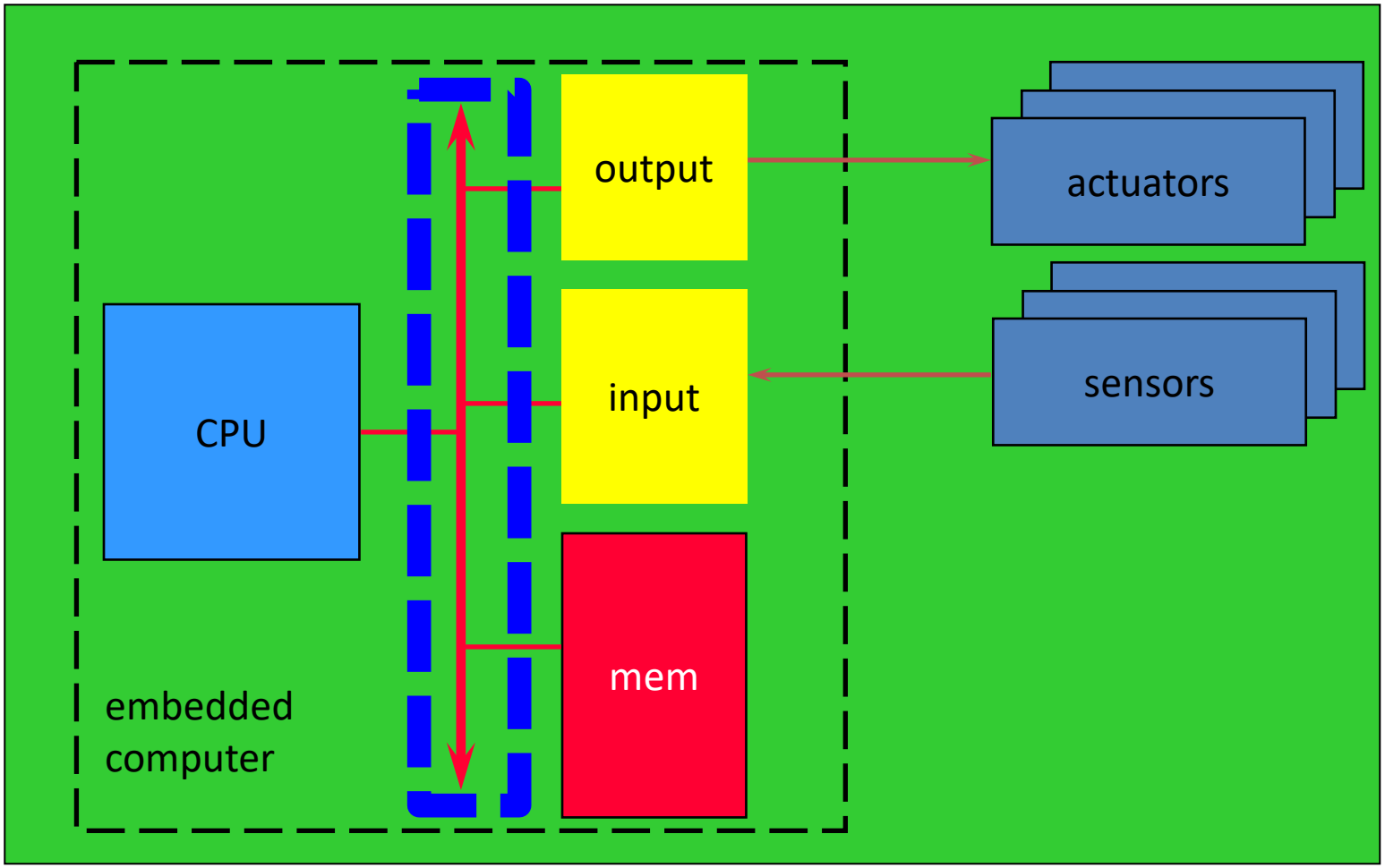
- I/O programming
  - Memory-mapped I/O vs. special-purpose I/O instructions
  - Busy-wait is simplest but very inefficient
    - Devices are usually slower than CPU
- Interrupts
  - Using buffer to allow input/output at different rates
  - Priorities and vectors allow to handle multiple interrupts
- Practical I/O interfaces
  - I2C, SPI, USB and GPIO

ECE 1175
Embedded Systems Design

The Bus System

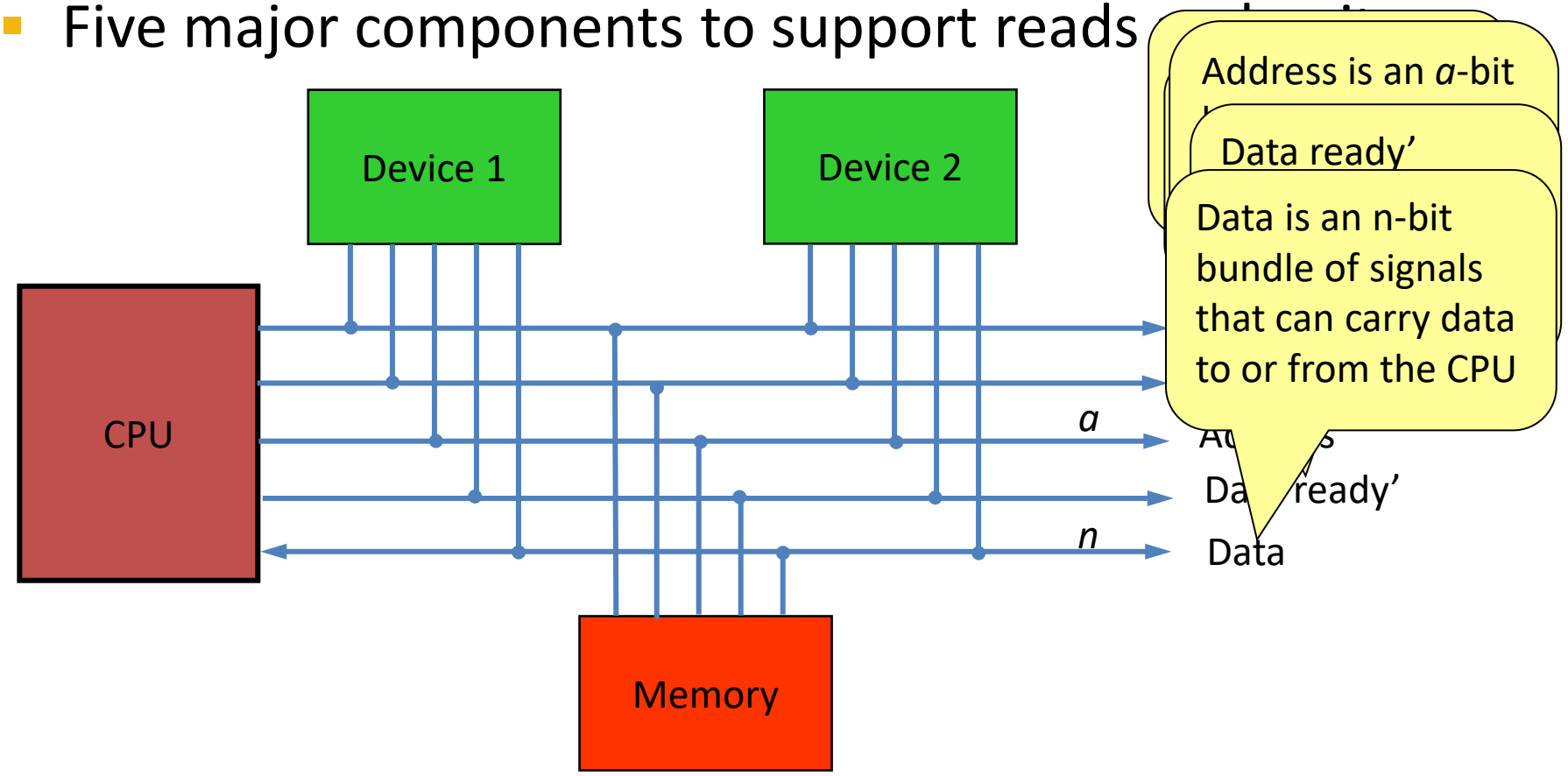
Wei Gao

Embedding A Computer: The CPU Bus



Typical Microprocessor Bus

- Bus is a set of wires and a protocol for the CPU to communicate with memory and devices
- Five major components to support reads



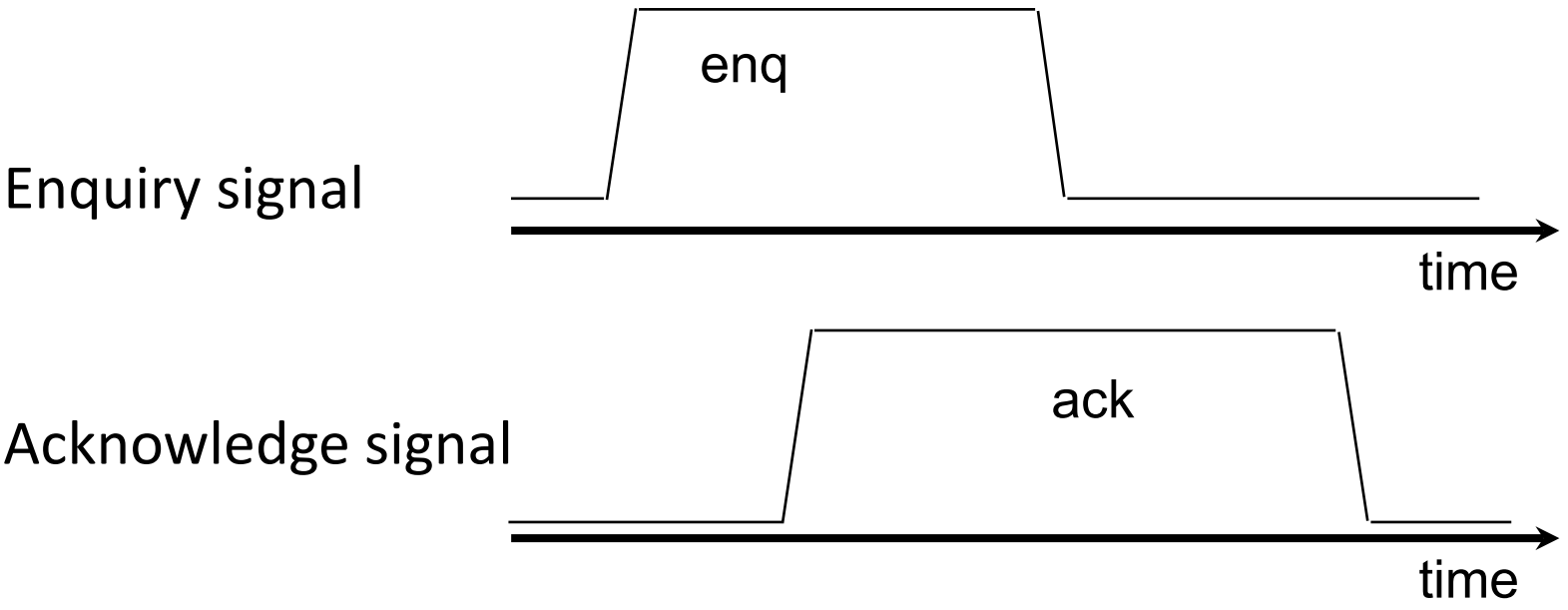
Types of Buses

- PC systems
  - Northbridge vs. Southbridge
  - PCI-e vs. ISA

- Embedded systems
  - I2C, SPI and GPIO

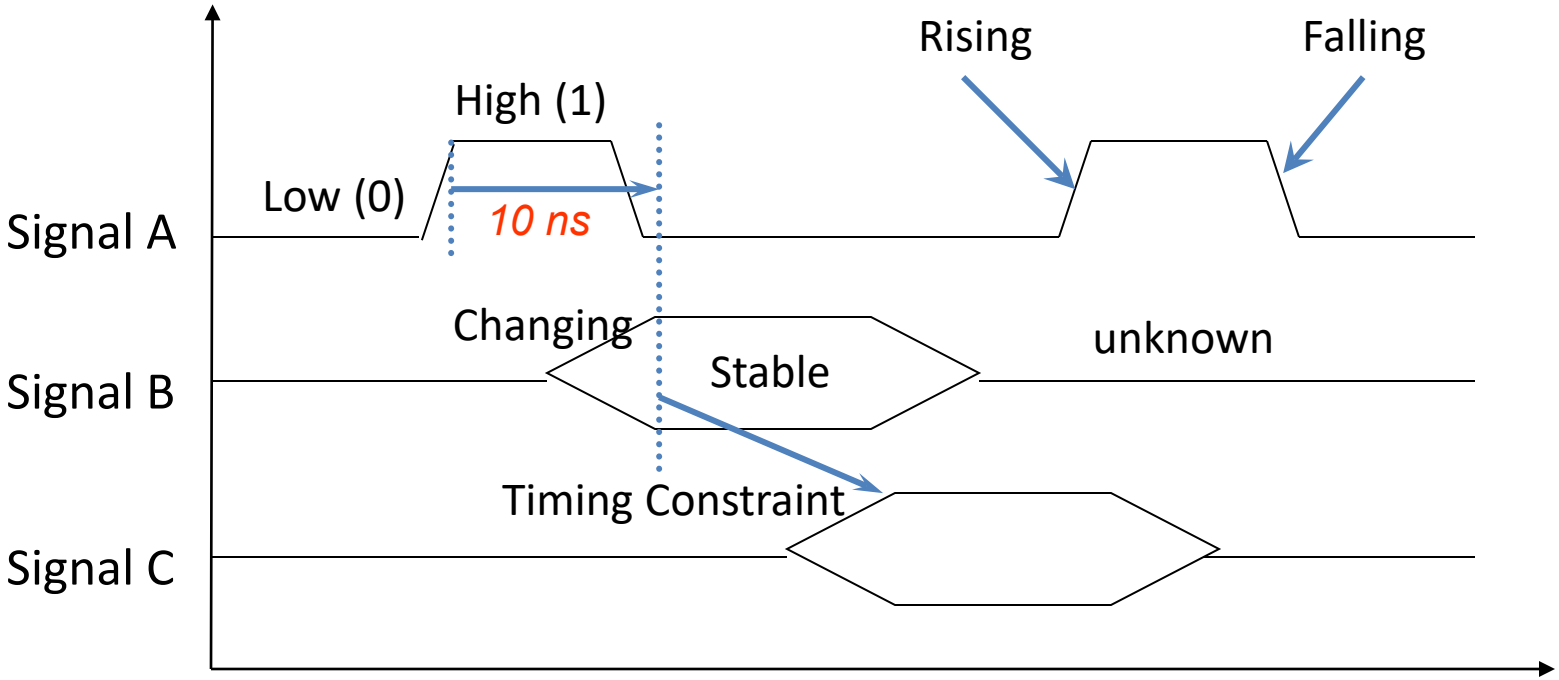
Timing Diagrams

- The behavior of a bus is specified as a timing diagram
- A timing diagram shows how signals on a bus vary over time.
  - Generally used for asynchronous machines with timing constraints.



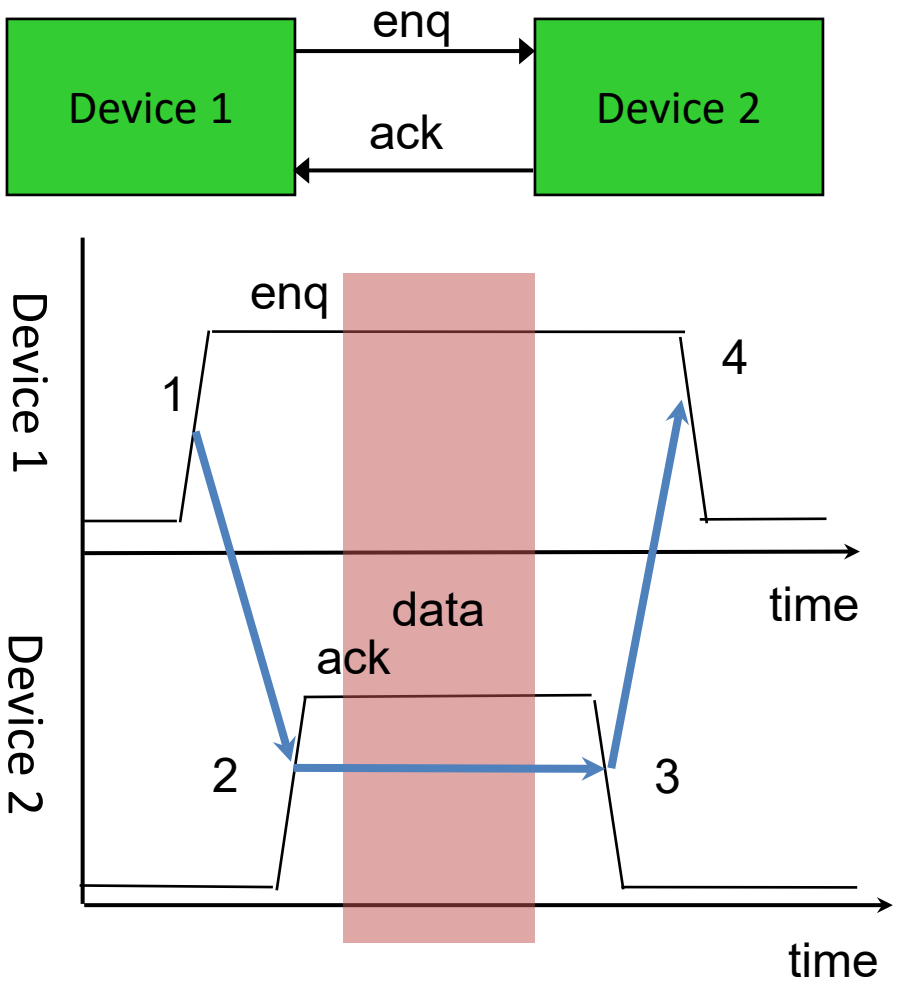
Timing Diagram Notation

- Timing diagram syntax:
  - Constant value (0/1), stable, changing, unknown.
- Timing constraints: minimum time between two events



Basic Block of Most Bus Protocols: Four-Cycle Handshake

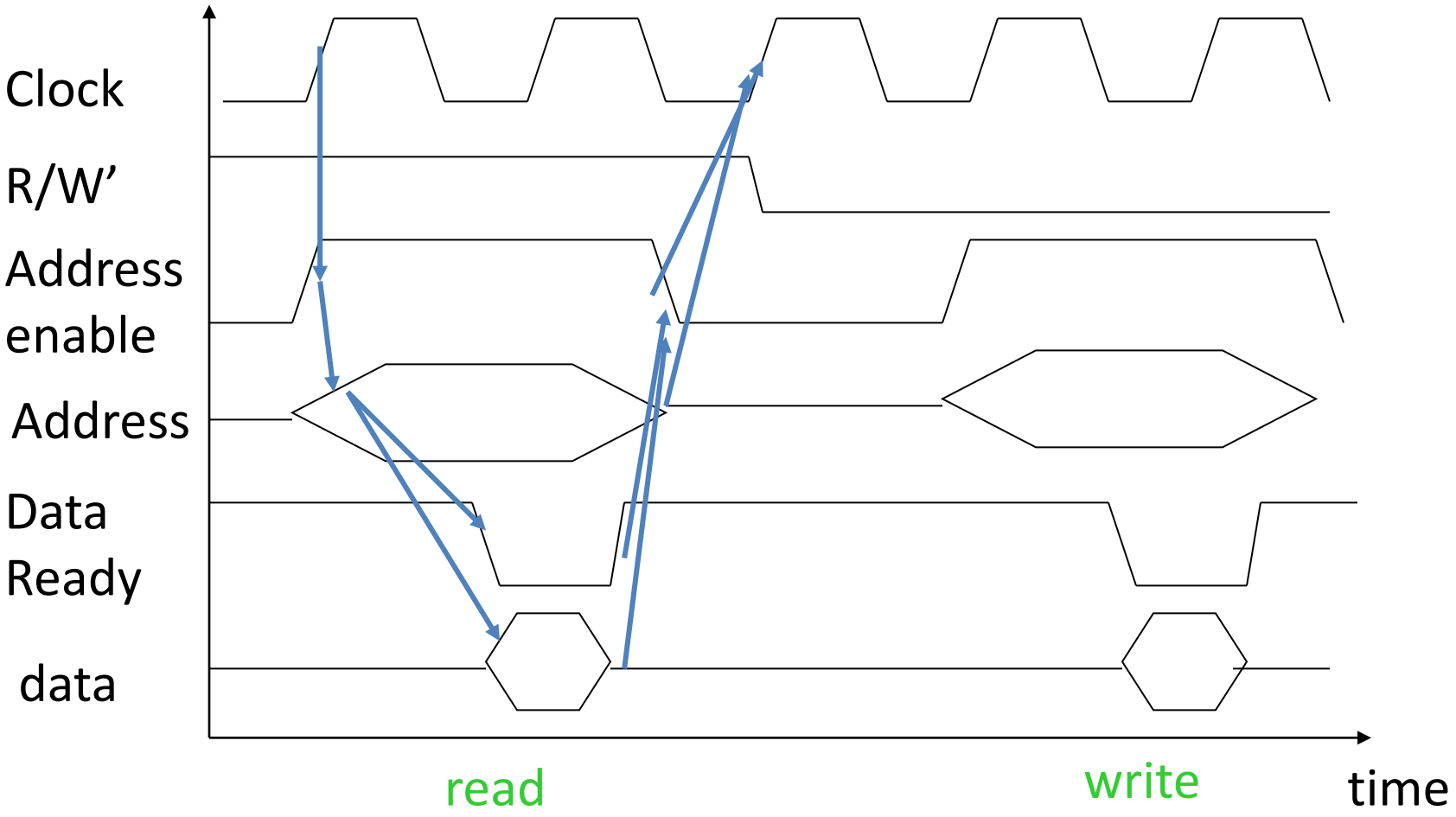
1. Device 1 raises its output to signal an enquiry
2. Device 2 raises output to signal an acknowledge
3. Device 1 and 2 can start to transmit data
4. Once transfer is complete, device 2 lowers output, signaling it finishes the data transmission
5. Device 1 lowers output



When Should You Handshake?

- When response time cannot be guaranteed in advance:
  - Data-dependent delay.
  - Component variations.

Typical Bus Access



Bus Design

- Bus signals are usually tri-stated.
  - Low, high, stable
- Address and data lines may be shared.
- Bus mastership
  - Bus master controls operations on the bus.
  - CPU is default bus master.
  - Other devices may request bus mastership.
    - Separate set of handshaking lines.
    - CPU can’t use bus when it is not master

Summary

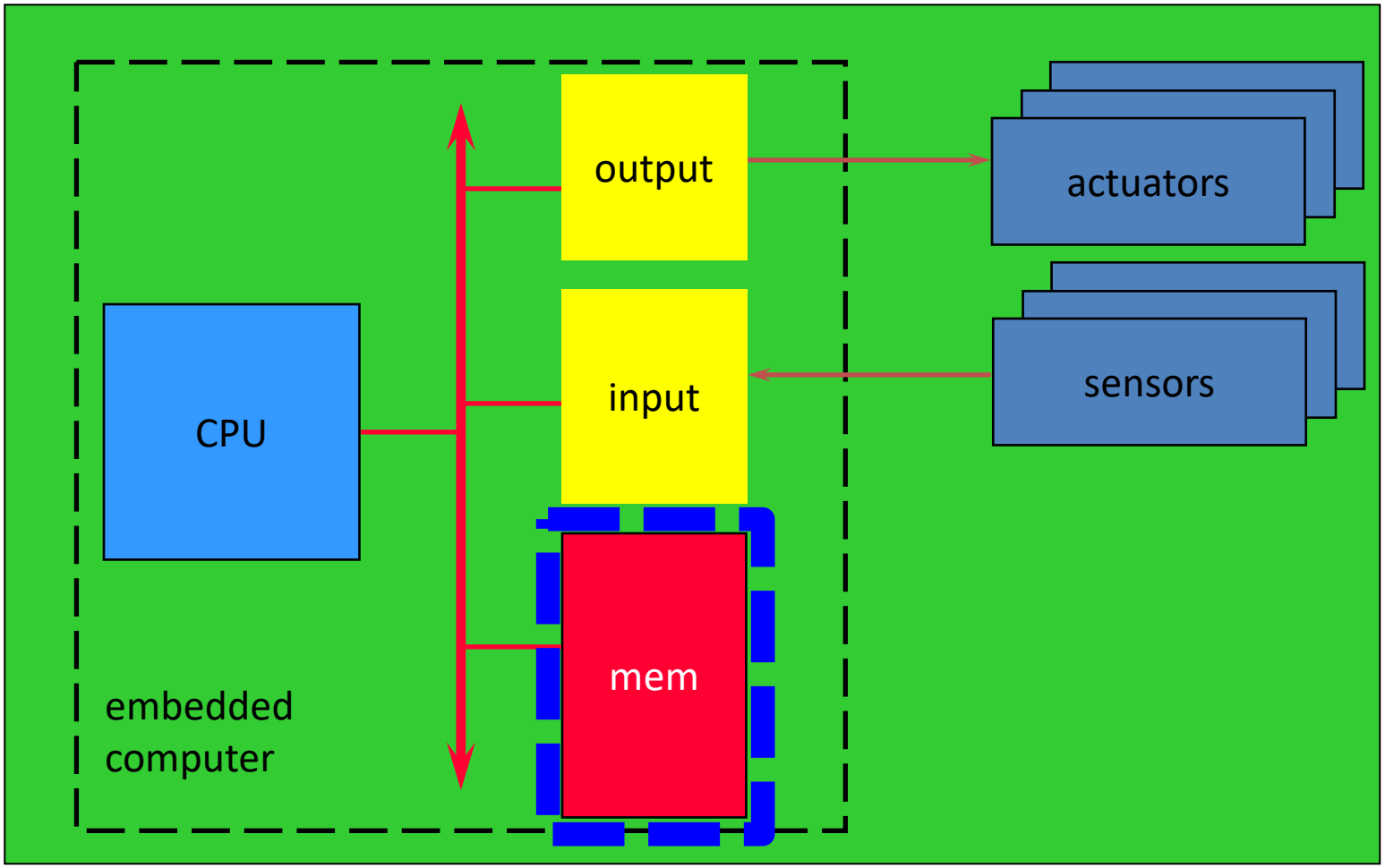
- The CPU Bus
  - A set of wires and protocols for CPU to communicate with memory and I/O devices.
  - Four-cycle handshake protocol
  - Timing diagram for typical bus access

ECE 1175
Embedded Systems Design

The Cache System

Wei Gao

Embedding A Computer

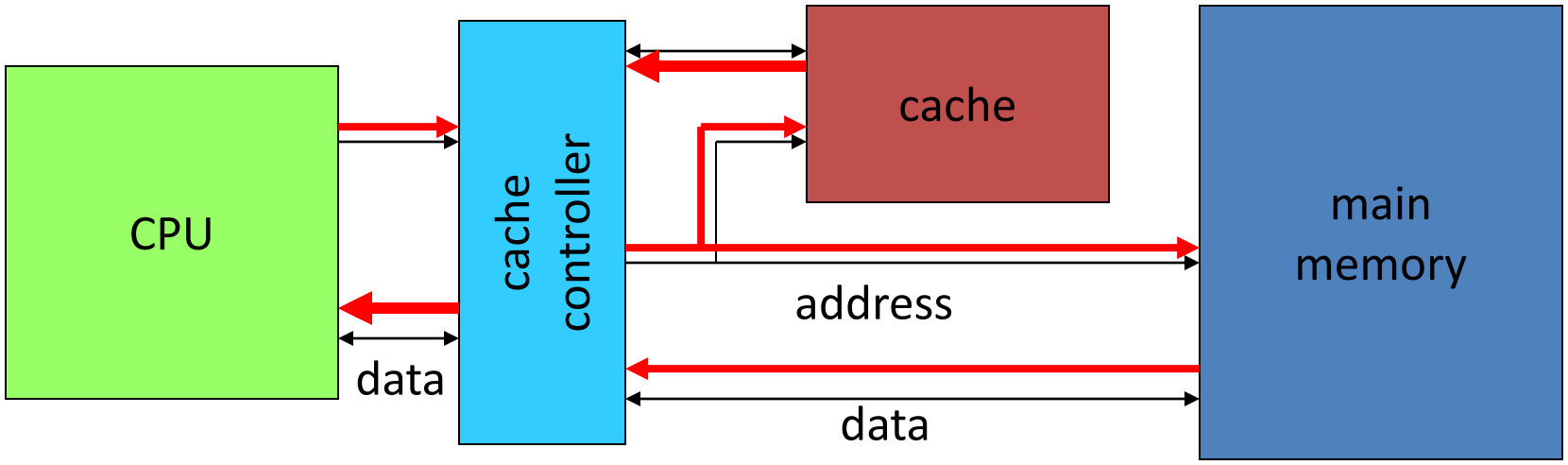


Memory System and Caches

- Memory is slower than CPU
  - CPU clock rates increase faster than memory
- Caches are used to speed up memory
  - Cache is a small but fast memory that holds copies of the contents of main memory
  - More expensive than main memory, but faster

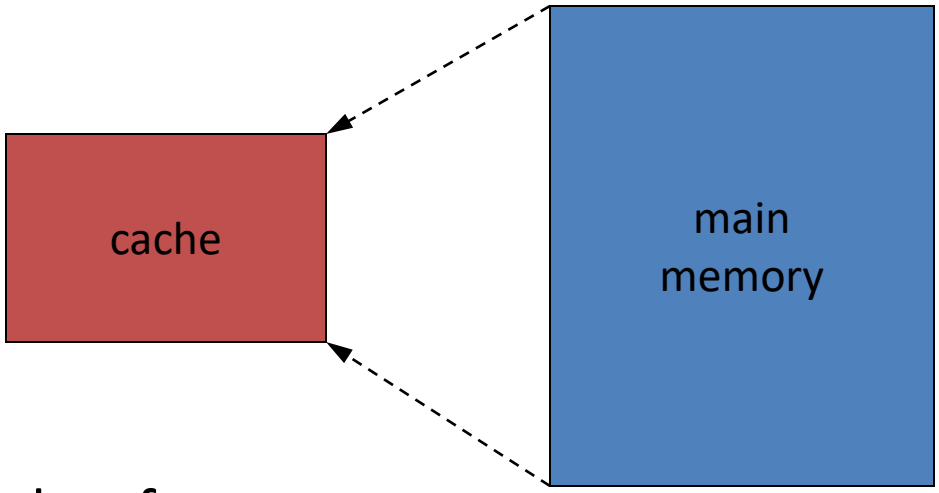
Cache in the Memory System

- Cache controller mediates between CPU and memory system
- Sends a memory request to both cache and main memory
- If requested location is in cache, request to main memory is aborted



Cache Operation

- Many main memory locations are mapped onto one cache entry.



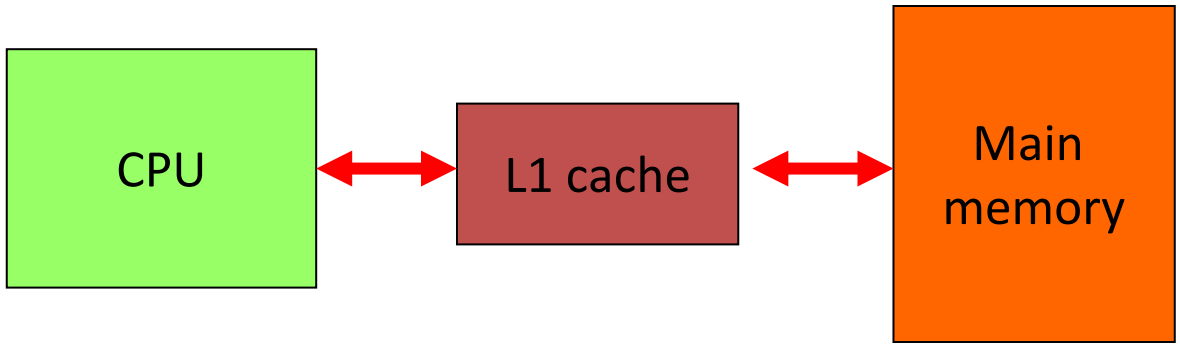
- May have caches for:
  - instructions;
  - data;
  - data + instructions (unified).
- Memory access time is no longer deterministic.

Terms

- Cache hit: required location is in cache.
- Working set: set of memory locations used by program in a time interval.
- Cache miss: required location is not in cache.
  - Compulsory (cold) miss: location has never been accessed.
  - Capacity miss: working set is too large.
  - Conflict miss: multiple locations in working set map to same cache entry.

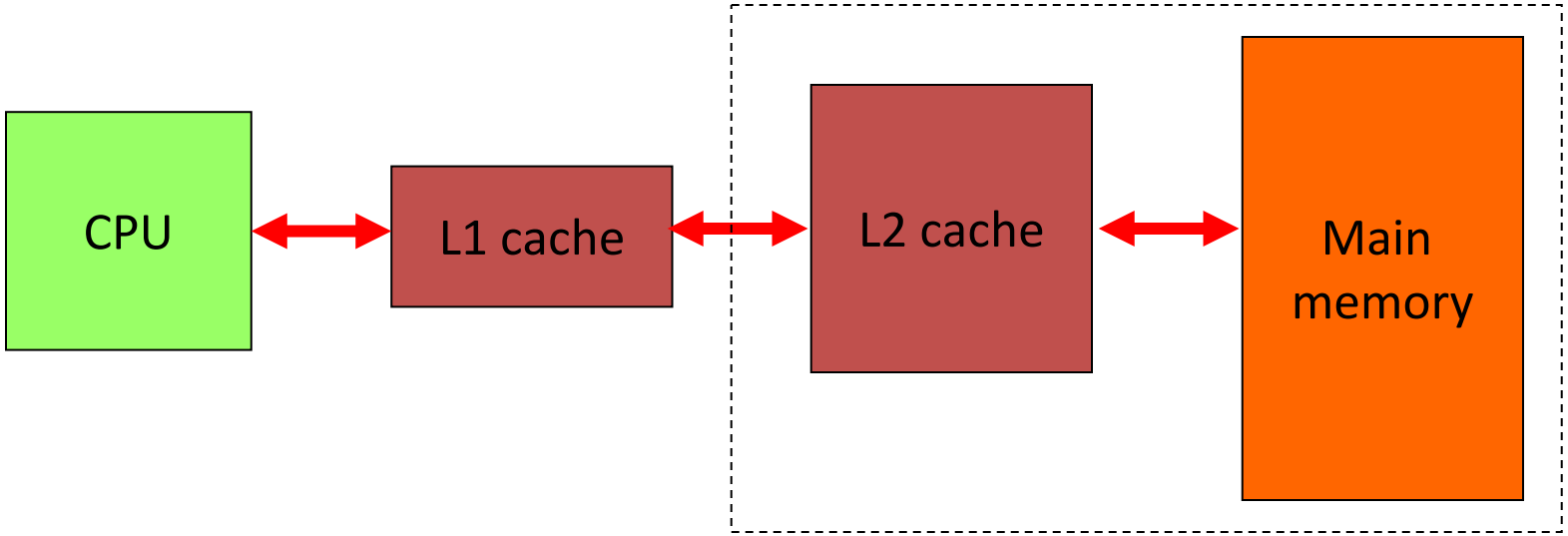
Memory System Performance

- h = cache hit rate: the percentage of cache hits
- t_{cache} = cache access time,
- t_{main} = main memory access time.
- Average memory access time:
  - $t_{av} = ht_{cache} + (1-h)t_{main}$
- Example: t_{cache} = 10ns, t_{main} = 100ns, h = 97%
  - t_{av} = 97%*10ns + (1-97%)*100ns = 12.7ns



Multi-Level Cache Access Time

- h_1 = cache hit rate for L1
- h_2 = cache hit rate for L2
- Average memory access time:
  - $t_{av} = h_1 t_{L1} + (1-h_1)(h_2 t_{L2} + (1-h_2)t_{main})$



Cache Performance Improvement

- To maximize cache hit rate
  - Keep most frequently-accessed memory items in fast cache.
- It is impossible to put everything in small cache
  - Need a good policy to decide which items should be in cache
  - e.g. who should be your favorite 5 people?
    - Nationwide unlimited calls by T-Mobile