Networking for Embedded Systems

- Why we use networks.

- Network abstractions.

- Example networks.
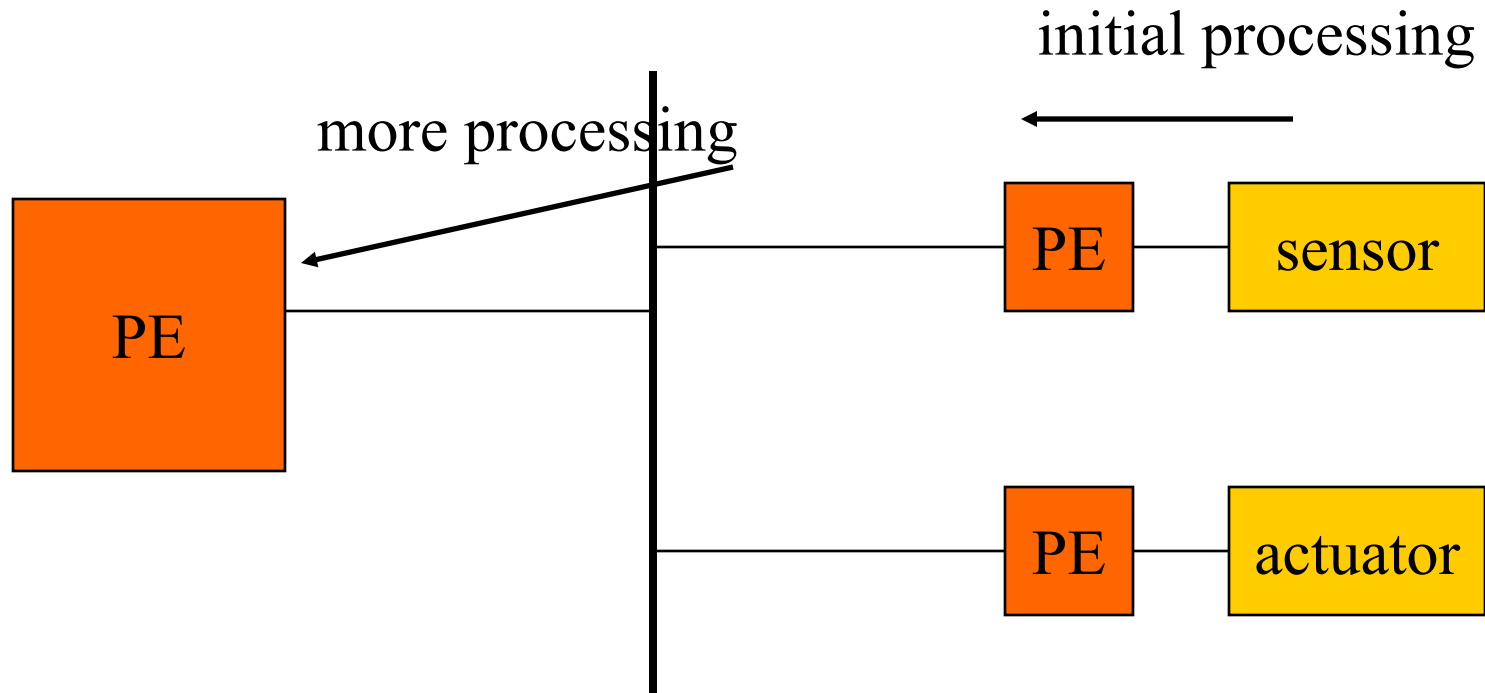
Overheads for *Computers as Components* 2nd *ed.*

Network elements

distributed computing platform:
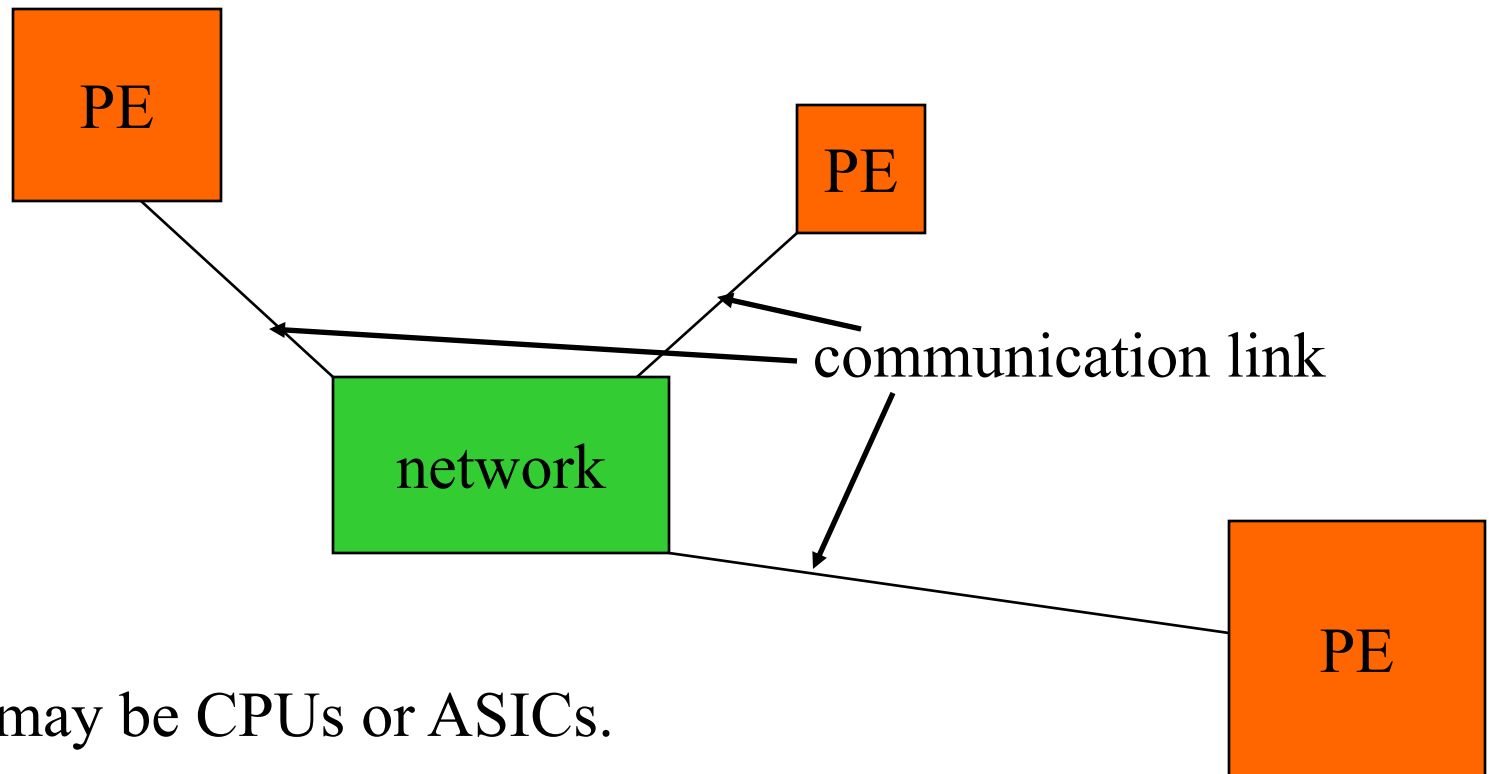
PEs may be CPUs or ASICs.

Networks in embedded systems

initial processing

more processing

PE

PE — sensor

PE — actuator

Overheads for *Computers as Components 2nd ed.*

Why distributed?

- Higher performance at lower cost.

- Physically distributed activities---time constants may not allow transmission to central site.

- Improved debugging---use one CPU in network to debug others.

- May buy subsystems that have embedded processors.

Overheads for *Computers as Components* 2nd ed.

Network abstractions

- International Standards Organization (ISO) developed the Open Systems Interconnection (OSI) model to describe networks:
  - 7-layer model.
- Provides a standard way to classify network components and operations.

OSI model

Layer	Description
application	end-use interface
presentation	data format
session	application dialog control
transport	connections
network	end-to-end service
data link	reliable data transport
physical	mechanical, electrical

OSI layers

- **Physical**: connectors, bit formats, etc.
- **Data link**: error detection and control across a single link (single hop).
- **Network**: end-to-end multi-hop data communication.
- **Transport**: provides connections; may optimize network resources.

OSI layers, cont'd.

▌ Session: services for end-user applications: data grouping, checkpointing, etc.

▌ Presentation: data formats, transformation services.

▌ Application: interface between network and end-user programs.

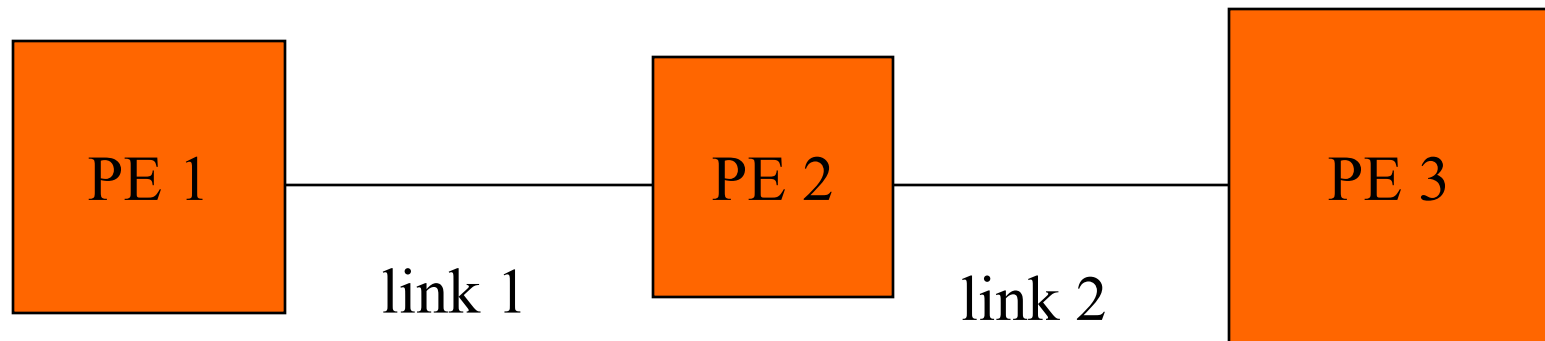# Hardware architectures

- Many different types of networks:
    - topology;
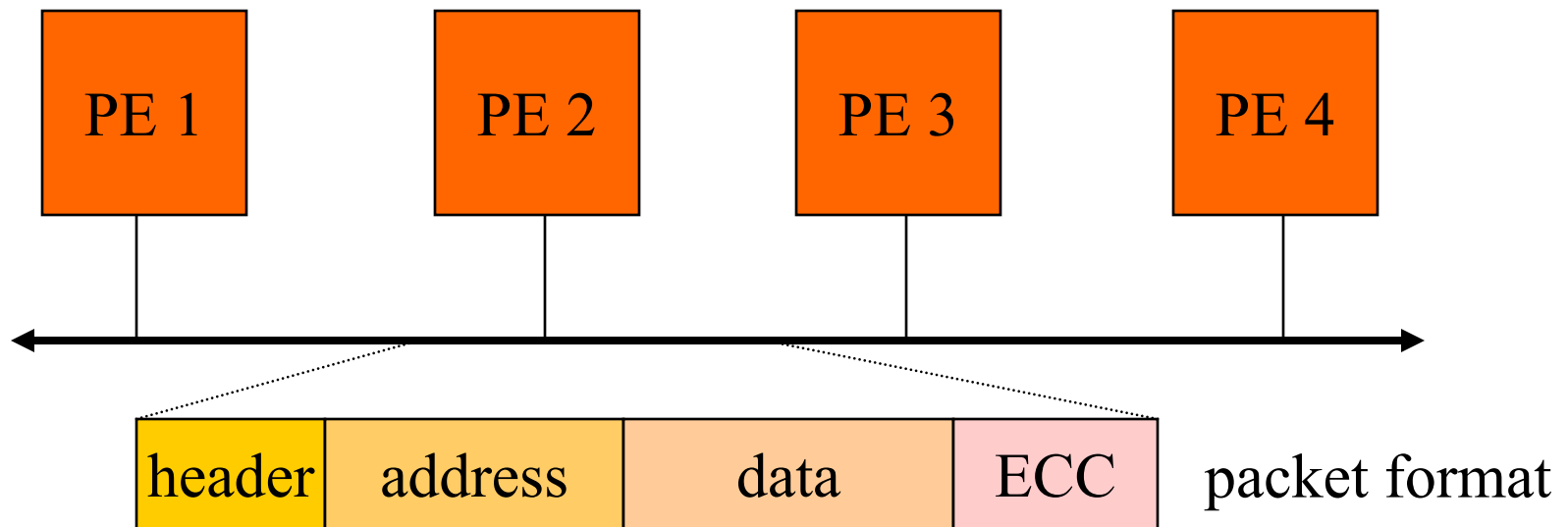   - scheduling of communication;
    - routing.

Overheads for *Computers as Components* 2nd ed.

Point-to-point networks

- One source, one or more destinations, no data switching (serial port):

PE 1 — link 1 — PE 2 — link 2 — PE 3

Bus networks

Common physical connection:

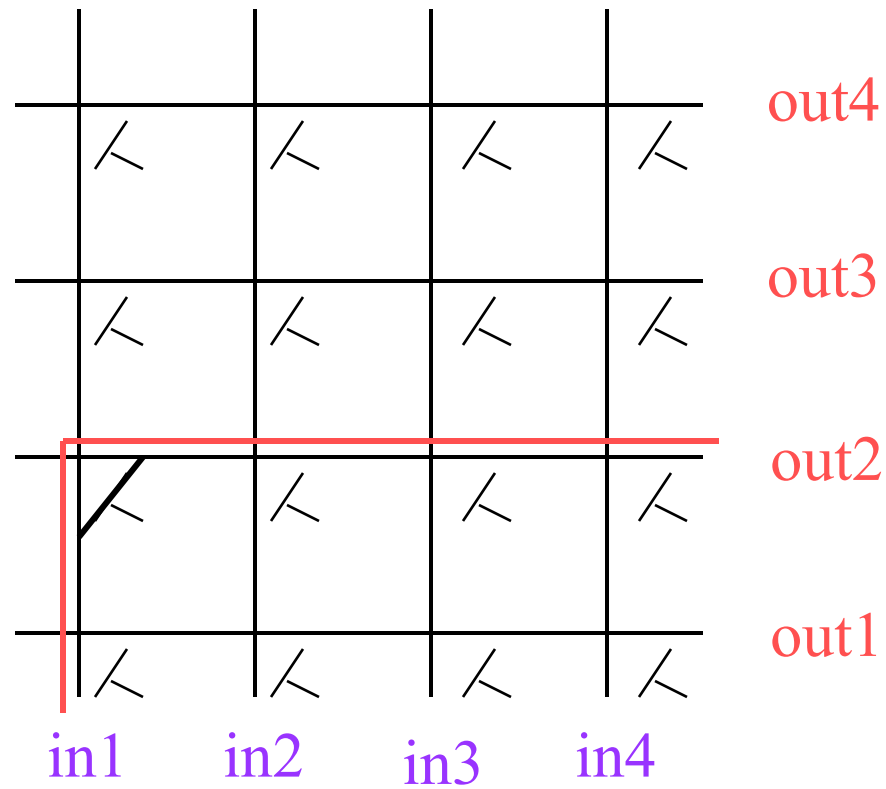| header | address | data | ECC | packet format |

Bus arbitration

- Fixed: Same order of resolution every time.

- Fair: every PE has same access over long periods.

  - round-robin: rotate top priority among Pes.

fixed

| A | B | C | | A | B | C |

round-robin

| A | B | C | | B | C | A |

A,B,C   A,B,C

Overheads for *Computers as Components 2nd ed.*

Crossbar

out4

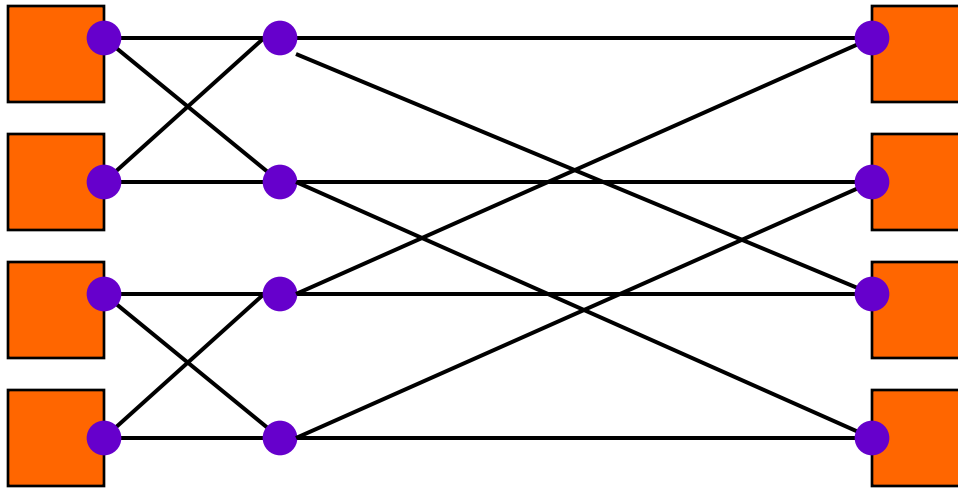out3

out2

out1

in1   in2   in3   in4
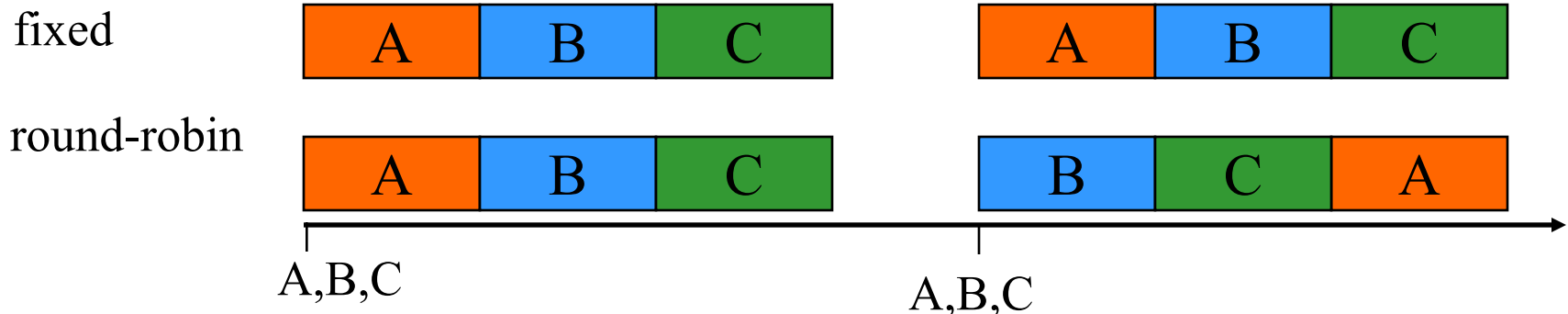
Crossbar characteristics

- Non-blocking.

- Can handle arbitrary multi-cast combinations.

- Size proportional to n^2.

Multi-stage networks

- Use several stages of switching elements.
- Often blocking.
- Often smaller than crossbar.

Overheads for *Computers as Components* 2*nd ed.*

Message-based programming

- Transport layer provides message-based programming interface:
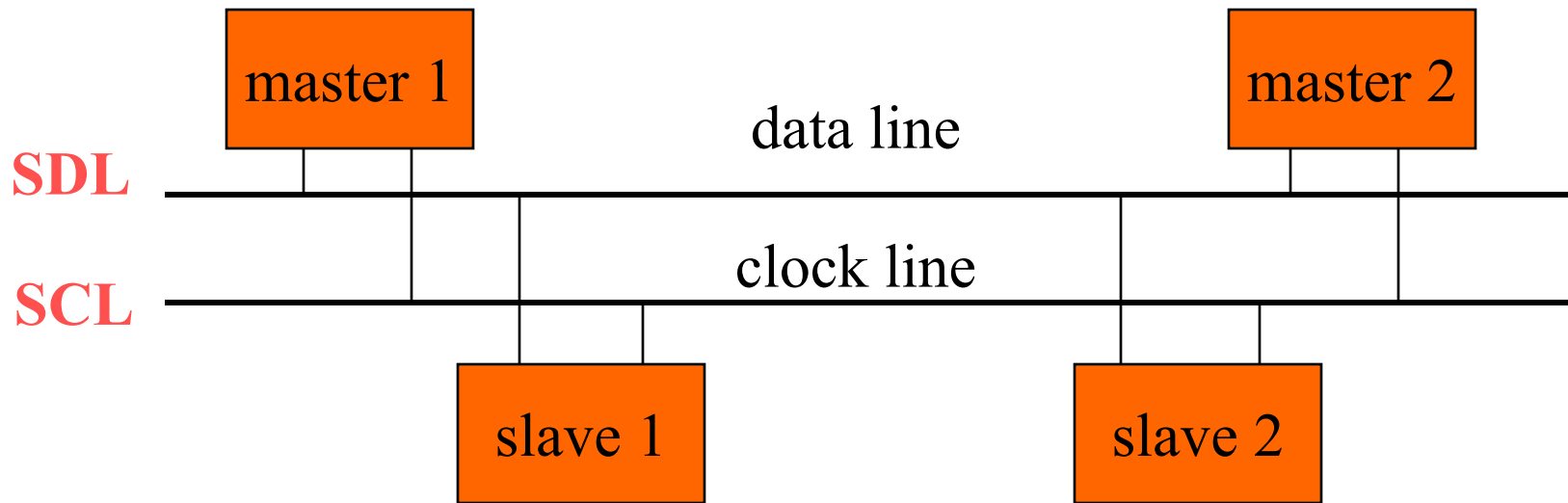
  ```
  send_msg(adrs,data1);
  ```

- Data must be broken into packets at source, reassembled at destination.

- Data-push programming: make things happen in network based on data transfers.
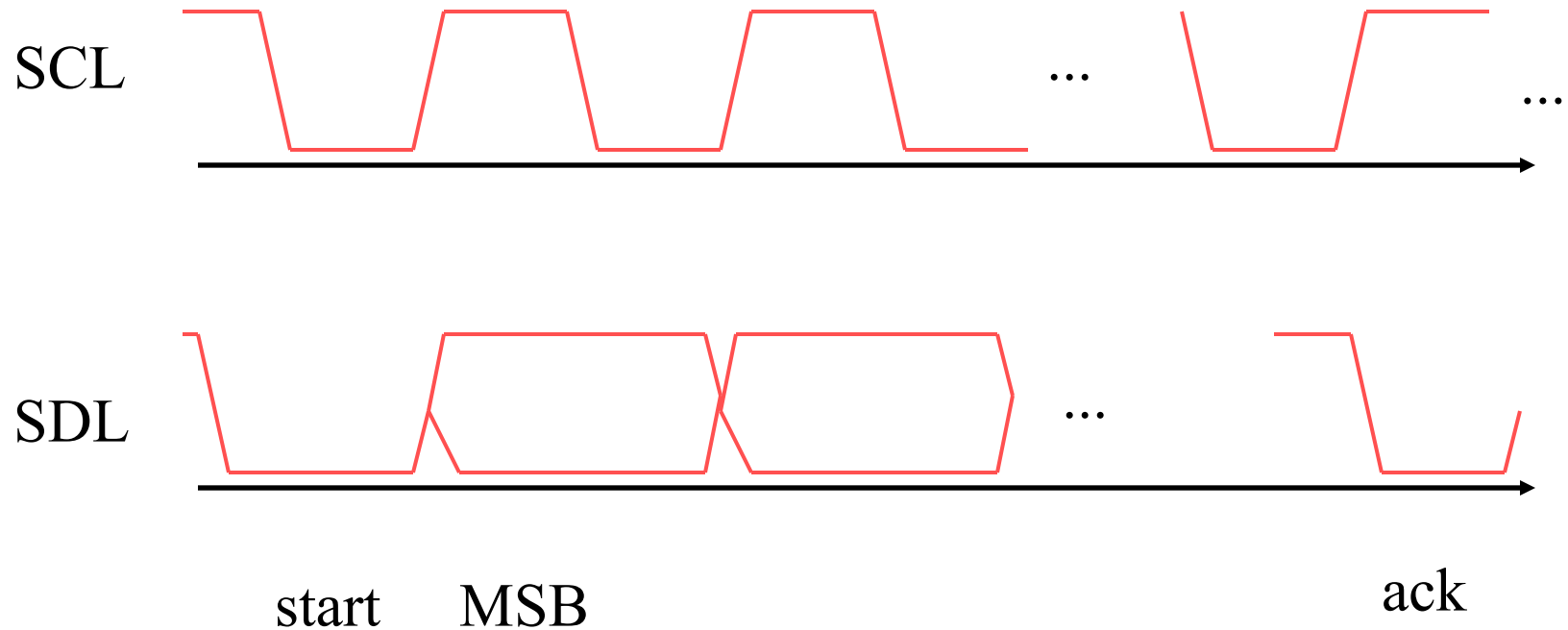
I²C bus

- Designed for low-cost, medium data rate applications.

- Characteristics:
  - serial;
  - multiple-master;
  - fixed-priority arbitration.

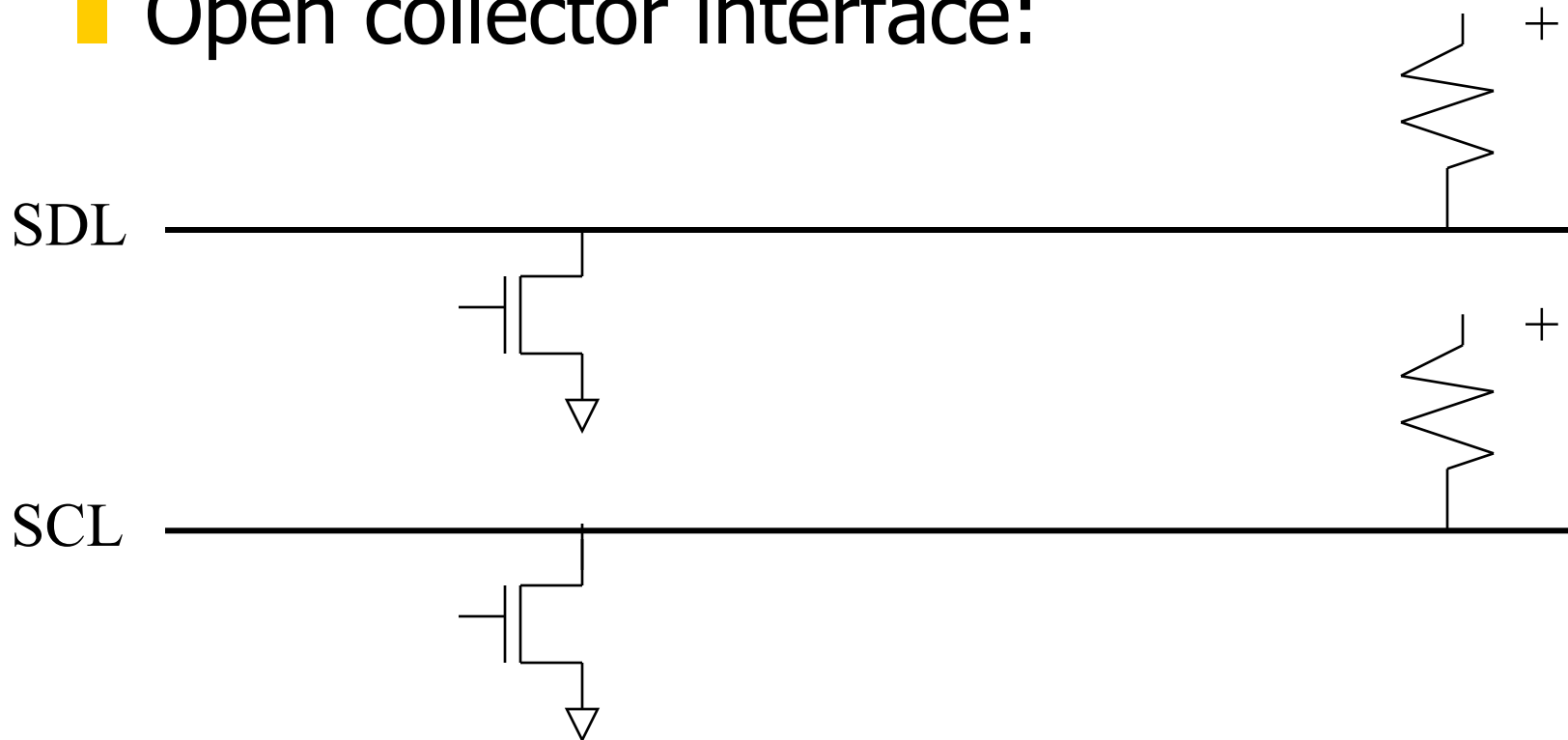- Several microcontrollers come with built-in I^{2}C controllers.

I²C physical layer

© 2008 Wayne Wolf

I²C data format

SCL

SDL

start   MSB   ...   ack

Overheads for *Computers as Components 2nd ed.*

I²C electrical interface

■ Open collector interface:

SDL

SCL

Overheads for *Computers as Components 2nd ed.*

I²C signaling

- Sender pulls down bus for 0.

- Sender listens to bus---if it tried to send a 1 and heard a 0, someone else is simultaneously transmitting.

- Transmissions occur in 8-bit bytes.

I²C data link layer

- Every device has an address (7 bits in standard, 10 bits in extension).
  - Bit 8 of address signals read or write.
- General call address allows broadcast.
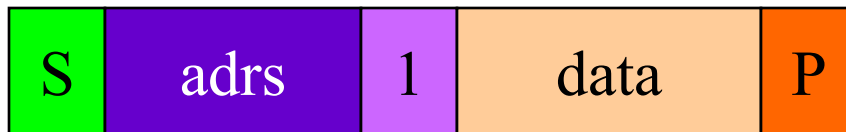
I²C bus arbitration

- Sender listens while sending address.

- When sender hears a conflict, if its address is higher, it stops signaling.

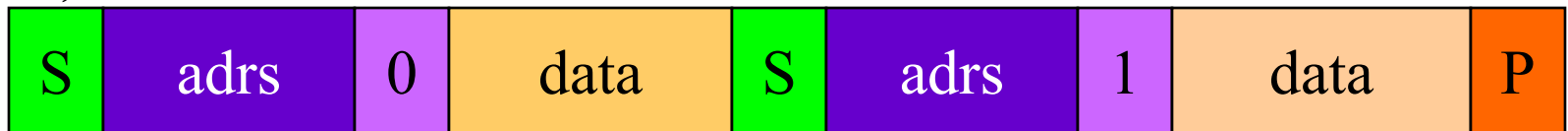- Low-priority senders relinquish control early enough in clock cycle to allow bit to be transmitted reliably.

Overheads for *Computers as Components 2nd ed.*

I²C transmissions

multi-byte write

| S | adrs | 0 | data | data | P |

read from slave

| S | adrs | 1 | data | P |

write, then read

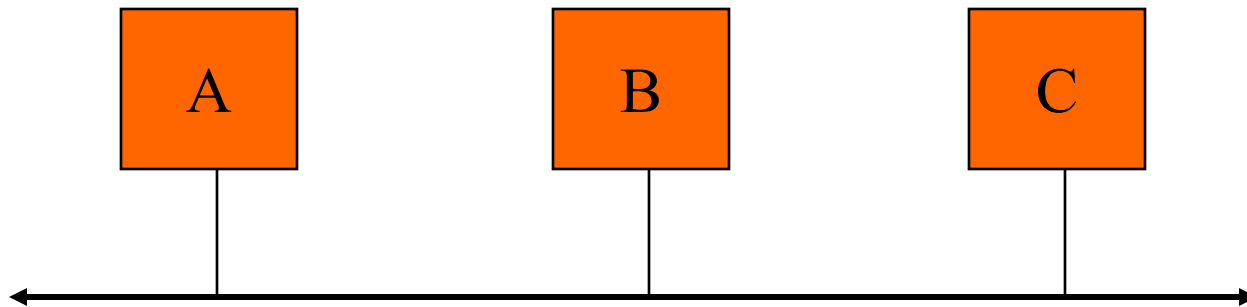| S | adrs | 0 | data | S | adrs | 1 | data | P |

Ethernet

- Dominant non-telephone LAN.

- Versions: 10 Mb/s, 100 Mb/s, 1 Gb/s

- Goal: reliable communication over an unreliable medium.

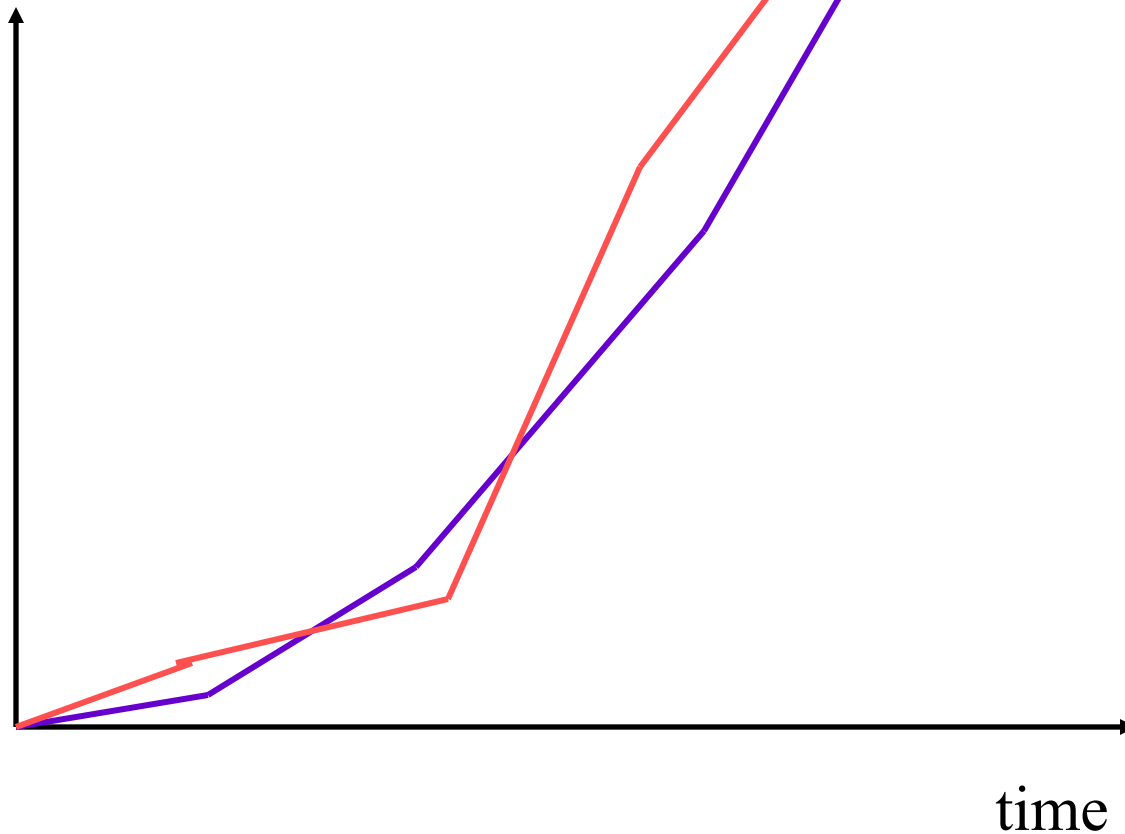# Ethernet topology

- Bus-based system, several possible physical layers:

CSMA/CD

Carrier sense multiple access with collision detection:

- sense collisions;

- exponentially back off in time;

- retransmit.

Overheads for *Computers as Components* 2nd ed.

Exponential back-off times

time

Overheads for *Computers as Components* 2nd *ed.*

Ethernet packet format

preamble	start frame	source adrs	dest adrs	length	data payload	padding	CRC

Overheads for *Computers as Components 2nd ed.*

Ethernet performance

- Quality-of-service tends to non-linearly decrease at high load levels.

- Can't guarantee real-time deadlines. However, may provide very good service at proper load levels.

Fieldbus

- Used for industrial control and instrumentation---factories, etc.

- H1 standard based on 31.25 MB/s twisted pair medium.

- High Speed Ethernet (HSE) standard based on 100 Mb/s Ethernet.