I/O External Serial Interfaces, and Timers

December 01

CSC201 Section 002

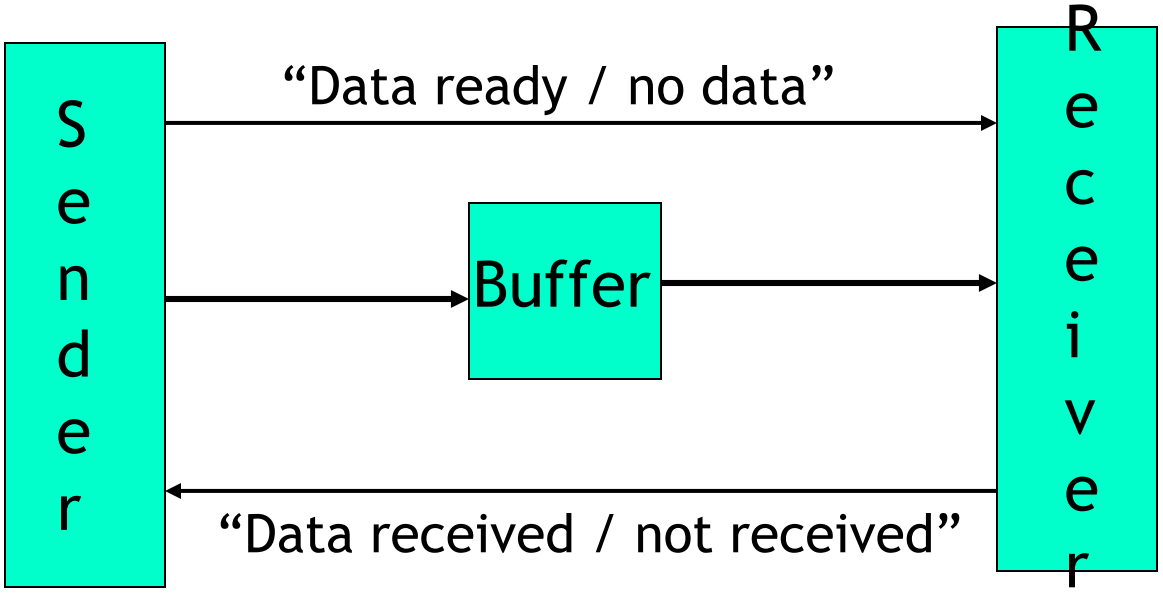
Fall, 2000

Codes

- ASCII -- only uses 128 out of 256 (2^8) different values
  - extensions to ASCII (specifying the other 128 characters)
- Unicode
  - 16 bit characters; no multibyte sequences or escape sequences needed
  - Used to encode symbols from many different languages
  - supported in Java, Windows NT, html browsers, ...
  - > half the codes (> 30000) are assigned already

Handshaking Between Controller and Device

- Needed when there is no synchronization between controller and device
- Signals
  - Sender specifies "data sent" or "no data"
  - Receiver specifies "ready to receive" or "received"



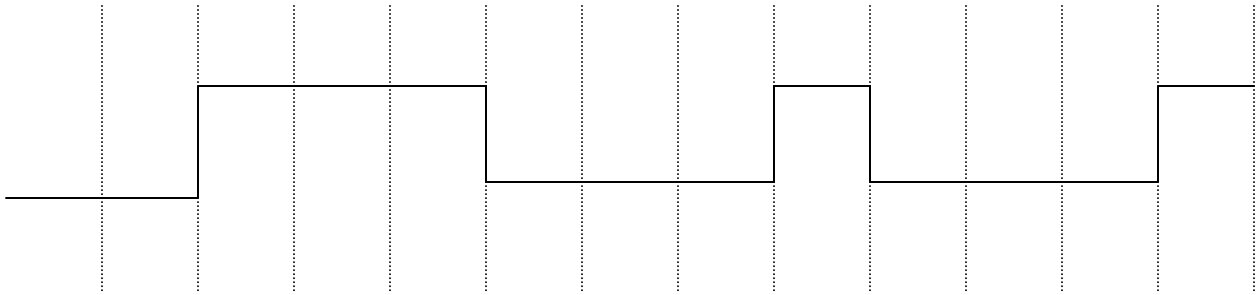
- Sender-Initiated Sequence
  1. --no data to send, receiver is ready to receive--
  2. Sender: put data in buffer
  3. Sender: "data sent"
  4. Receiver: receive data from buffer
  5. Receiver: "received"
  6. Sender: "no data"
  7. Receiver: "ready to receive"

Handshaking (Continued)

- Receiver-initiated
  1. --no data, previous data received--
  2. Receiver: "ready to receive"
  3. Sender: put data in buffer
  4. Sender: "data sent"
  5. Receiver: receive data from buffer
  6. Receiver: "received"
  7. Sender: "no data"

Serial Interface

- Sending data over a single wire
- How recognize individual bits on the wire?
  - Synchronize sender and receiver, send 1 bit per synchronized time interval
- How recognize when there is a character being sent, vs. nothing being sent?
  - "start bit" precedes the character
  - "stop bit" ends the character
  - with 1 start bit + 8-bit character + 1 stop bit, 20% of the bits are "overhead"

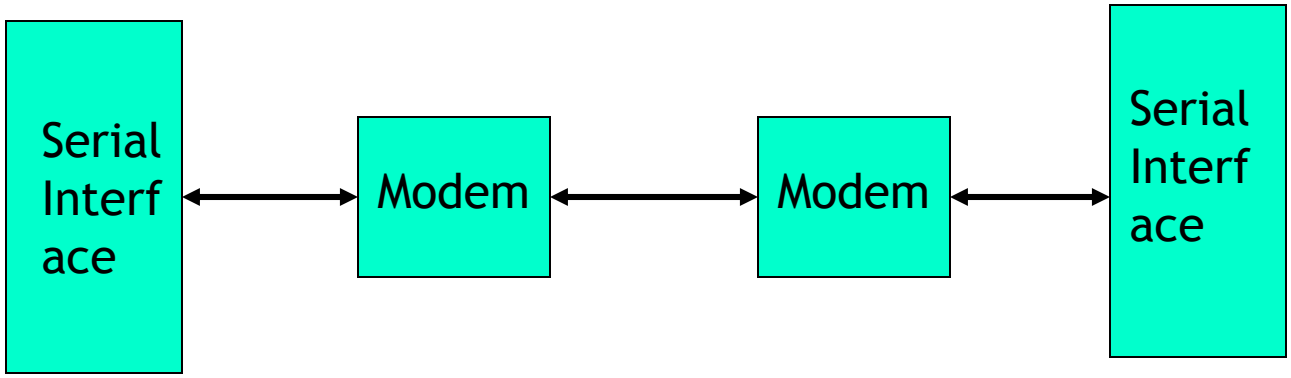


Serial Interface

- A UART
  - converts bytes or words into a (serial) bit-stream
  - converts a serial bit-stream back into bytes and words
- RS232 standard: serial interface with rate up to 115,200 bits/s

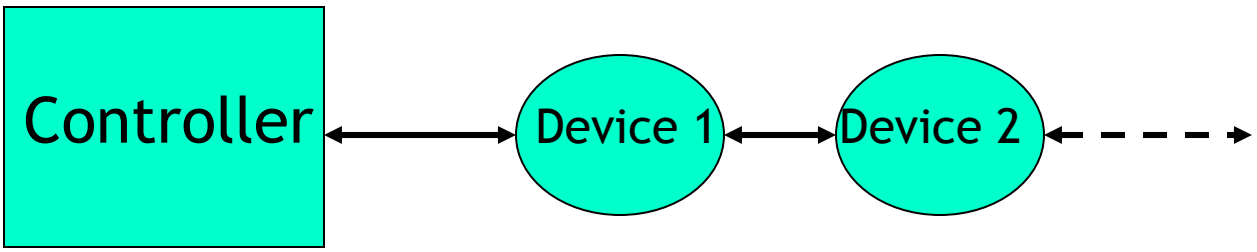
Modems

- Convert bits (digital signals) to/from analog signals (such as phone signals)
- "Half duplex": transmit in only one direction at a time over a wire
  - "full duplex": transmit in both directions at the same time
- Current rates up to 57,600 bits/s



Universal Synchronous Bus (USB)

- Relatively new serial interface standard
- Up to 1.5 Mb/s
- Up to 127 devices can be connected
  - Connected in a "daisy-chain", or through a hub



Daisy-chain

- Hot-pluggable, no reboot needed
  - Autoidentify and autoconfigure each device

Firewire Serial Bus

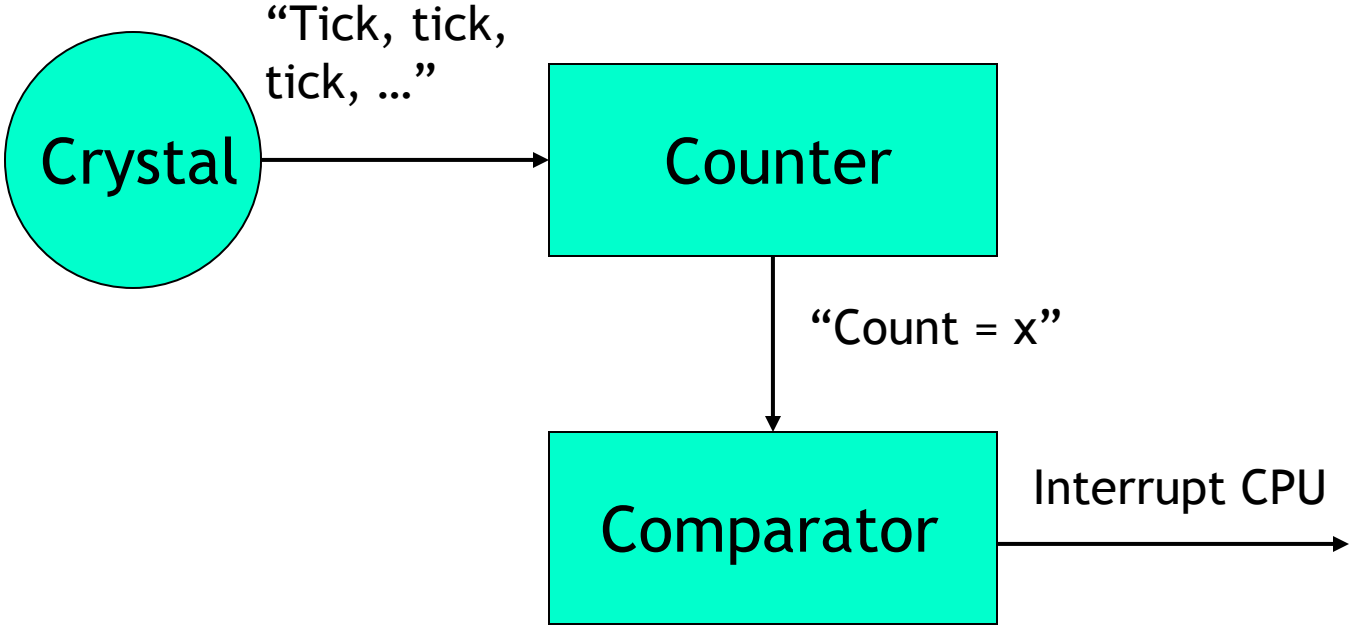
- IEEE standard 1394
  - Used for computers and consumer electronics
- Data rates from 25Mb/s to 400 Mb/s
- Multiple devices can be connected (daisy-chain or hub)
- Hot-pluggable, autoidentify and autoconfigure devices

System Clocks and Timers

- "Sort of" a device
- Functions
  - Maintaining the system date and time of day
  - Controlling multitasking
  - Alarms, alerts, reminders -- for operating system, and user applications
- Clock time source (source of "ticks")
  - Input voltage source (AC)
  - or, a built-in crystal (much higher resolution)

System Clocks and Timers

- A counter keeps track of how many ticks have occurred
- A comparator tells you when the count has reached a desired value
  - Generate an interrupt when count reaches desired value, and reset count to zero
  - The "desired value" is programmable



On Each Timer Interrupt…

1. Update the time of day
2. Check whether current task time slice has expired
3. Check whether any requested event time (alarm or alert) has elapsed
   - "spin down disk after 2 minutes of inactivity"
   - "prompt user for input after 30 seconds"
   - "switch out the currently-executing task after 100 ms"
4. Update elapsed time per event type