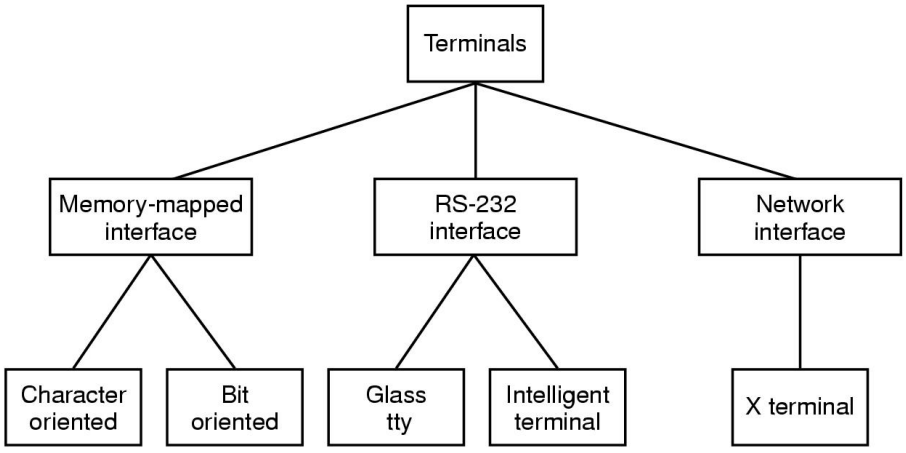
CSCE 351
Operating System Kernels

Terminals

Steve Goddard
goddard@cse.unl.edu

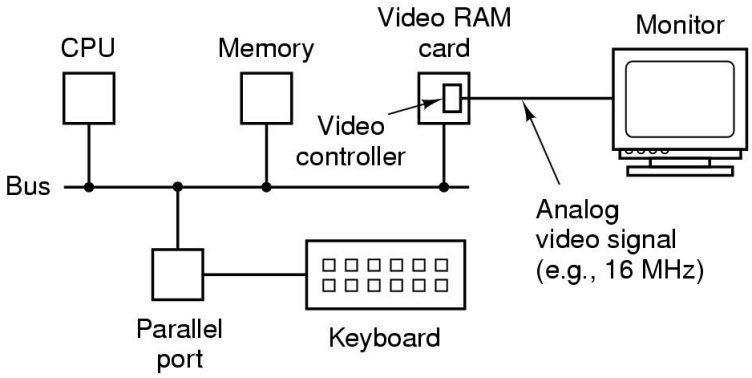
http://www.cse.unl.edu/~goddard/Courses/CSCE351

Terminal Hardware



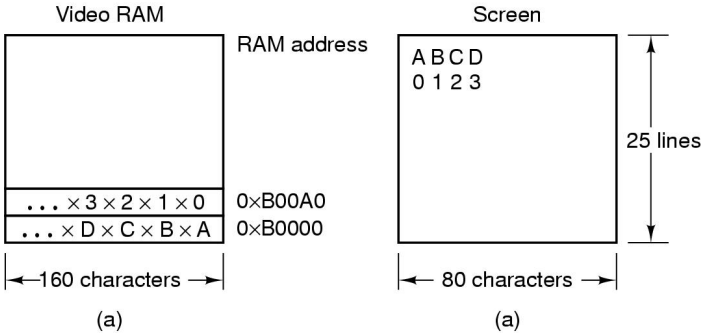
Memory Mapped Terminals

- CPU writes data to special memory called Video RAM
- The video controller (often a graphics controller now) reads the characters (or commands) from the video RAM and displays (executes) them.



IBM PC Monitor Interface

- Simplest mode is the character-mapped console mode: 25 lines of 80 characters each.
- Each character takes two bytes of storage
  - » 1 byte for the ASCII code
  - » 1 byte for the attribute: color, reverse video, blinking, etc.



IBM PC Monitor Interface

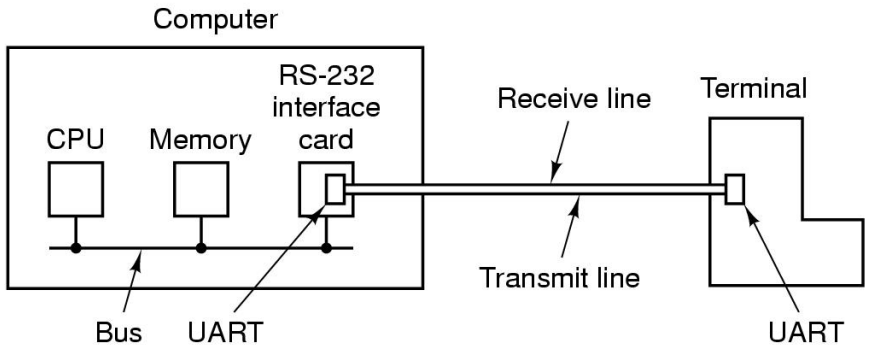
- Other modes are bit-oriented or graphic modes of various resolution: e.g., 480x640 or 768x1024
- Each pixel is individually controlled.
  - » A number associated with the pixel represents its display attributes
  - » 1-32 bits per pixel are used
    - 1 bit/pixel is for monochrome displays
    - 24 bits/pixel is common with a byte for each color attribute
    - 32 bits/pixel is for “true color” displays
  - » 768x1024 resolution with 24 bits/pixel requires 2MB of RAM just to hold the current image.
  - » Controllers often support double buffering to prevent flickering when screen changes occur

IBM PC Keyboard Interface

- Serial interface to a controller chip on main board
- Keyboard contains an embedded microprocessor
- Interrupt is generated for each key “action”
- Keyboard sends the key code, not the ASCII code in the I/O register
- Flexible interface
  - » Allows sending raw data to applications
  - » Allows sending processed (line-oriented) data to applications

RS-232 Terminals

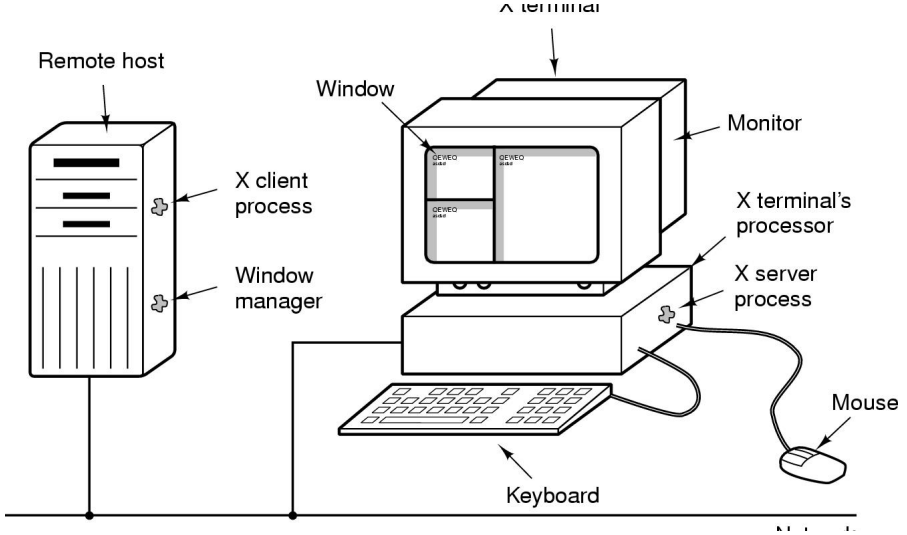
- Keyboard and display are a single device connected to the computer via a serial RS-232 interface line.
- Universal Asynchronous Receive Transmitters (UARTs) to do character to serial and serial to character conversions



RS-232 Terminals

- Glass ttys (or dumb terminals)
  - » Functionally the same as hardcopy ttys
  - » Glass ttys and hardcopy ttys are both obsolete
- Intelligent Terminals
  - » Have CPU, memory, and SW in ROM
  - » Supports some escape sequences

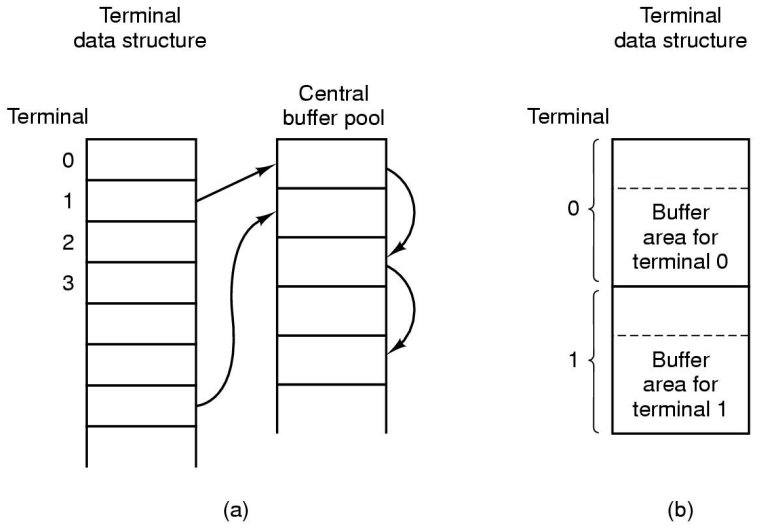
X Terminals



Terminal Software

- Keyboard driver and display driver are almost independent.
- Many keyboard drivers use a loadable key map that maps the key codes to characters.
- Keyboard driver has two modes:
  - » Character-oriented
    - ❖ Called raw mode in UNIX
    - ❖ non-canonical mode in POSIX
  - » Line-oriented
    - ❖ Called cooked mode in UNIX
    - ❖ Called canonical mode in POSIX

Buffering for Cooked mode



Echoing in Cooked mode

- Echoing displays the characters typed
  - » Editing and other “special” characters are not displayed
  - » A special character, such as <cntrl>v, is used to display normally non-visible characters
  - » Complicated by trying to display more characters on the line than the physically possible
    - ❖ Line wrapping
    - ❖ Discarding “extra” characters
  - » Carriage return and linefeed are assumed when one types the “enter” key
    - ❖ Driver must make sure that the cursor is correctly posistioned on the display, which may mean echoing both carriage return and a line feed.

The termios structure

```
struct termios {
  tcflag_t c_iflag; /* input modes        */
  tcflag_t c_oflag; /* output modes       */
  tcflag_t c_cflag; /* control modes      */
  tcflag_t c_lflag; /* local modes        */
  speed_t c_ispeed; /* input speed        */
  speed_t c_ospeed; /* output speed       */
  cc_t c_cc[NCCS];  /* control characters */
};
```


Output Terminal Drivers

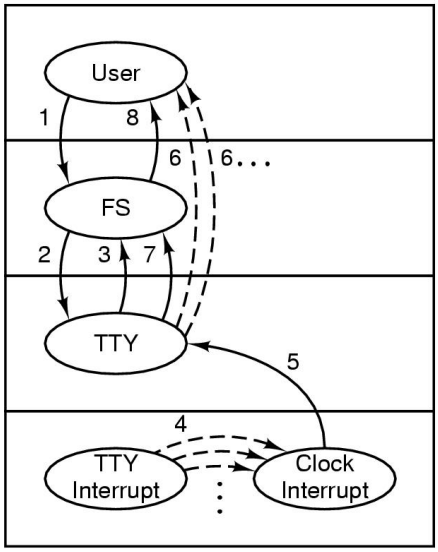
- RS-232 terminal drivers are very different from memory-mapped terminals
- RS-232 terminals
  - » Output buffers are associated with each terminal
  - » Characters written or echoed are copied to a buffer and transmitted one at a time, synchronously
- Memory-mapped terminals
  - » Characters are written to the video RAM
  - » Driver keeps track of
    - ❖ Cursor position
    - ❖ Backspace, CR, NL processing
    - ❖ Scrolling
- Most drivers support ANSI defined Escape sequences for display, see Figure 3-36

Overview of MINIX Terminal Driver

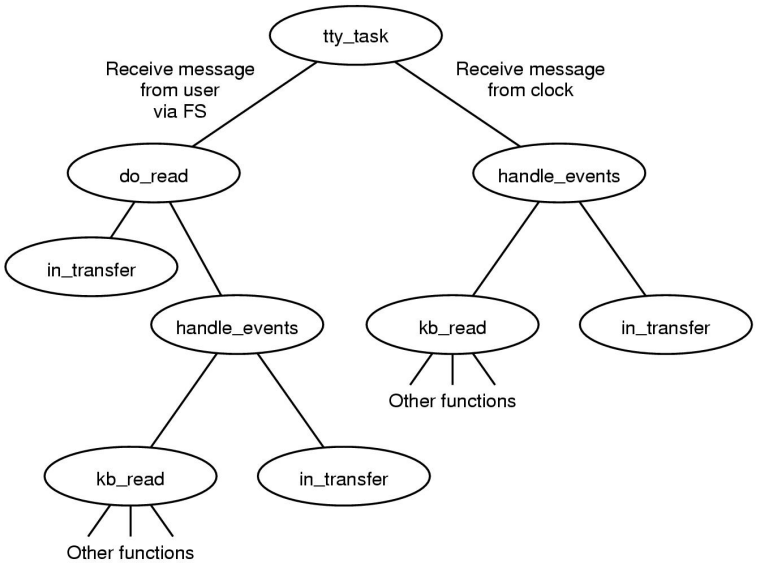
- Terminal driver is the largest and most complex driver in MINIX (and most operating systems)
  - » Handles keyboard, display, and RS-232
- Contained in 6 files and accepts 7 messages:
  - » Read from the terminal (from FS for user)
  - » Write to the terminal (from FS for user)
  - » Set terminal parameters for IOCTL (from FS for user)
  - » I/O occurred during last clock tick (from clock interrupt)
  - » Cancel previous request (from FS when a signal occurs)
  - » Open a device
  - » Close a device

Terminal Read

with no pending characters



Terminal Read
input call tree



Terminal (console) Output

- Console output is simpler than terminal input
- Console is memory mapped, so no interrupts
- User output example:
  - » User calls **`printf`**
  - » **`printf`** calls **`WRITE`** to send a message to the FS
  - » The FS sends a message to the terminal driver, **`tty_task`**
  - » The **`tty_task`** copies the characters from user space to video RAM

Overview of Driver Calls for Terminal Output