Serial Devices
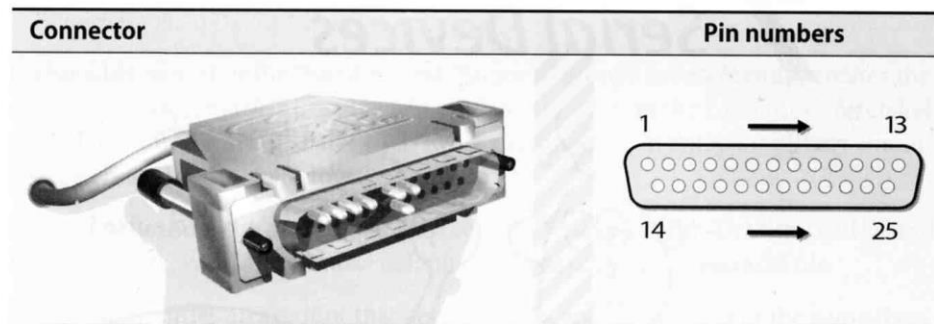
Serial devices

❑ Terminal

❑ Modem

❑ Mice

❑ ...

Serial standard (1)

❑ RS-232 standard on DB25 connector

- Electrical characteristics
- Meaning of each signal wire
- Ping assignment
- DB25P   (male)
- DB25S   (female)
- DTE (Data Terminal Equipment)
- DCE (Data Circuit-terminating Equipment)
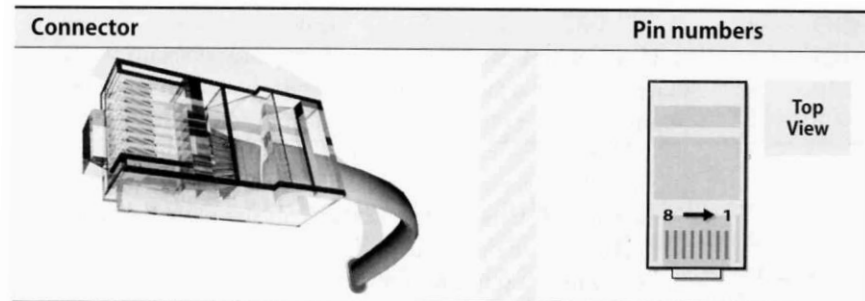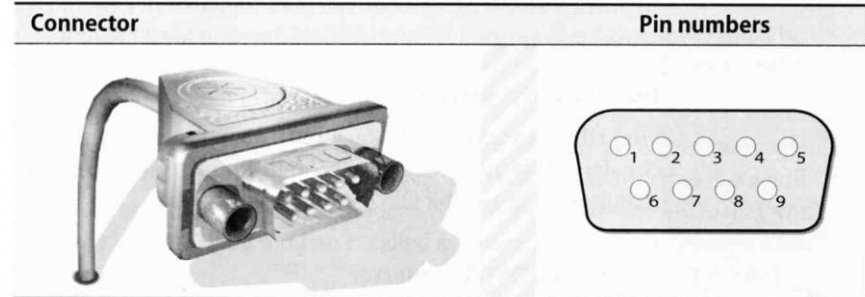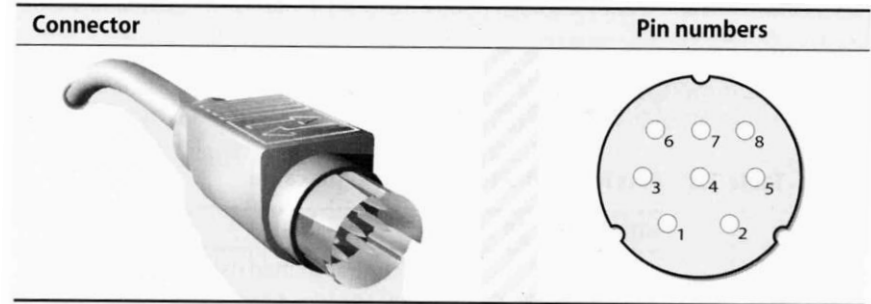
Serial standard (2)

❑ RS-232 signals and ping assignment

Pin	Name	Function	Pin	Name	Function
1	FG	Frame ground	14	STD	Secondary TD
2	TD	Transmitted data	15	TC	Transmit clock
3	RD	Received data	16	SRD	Secondary RD
4	RTS	Request to send	17	RC	Receive clock
5	CTS	Clear to send	18	–	Not assigned
6	DSR	Data set ready	19	SRTS	Secondary RTS
7	SG	Signal ground	20	DTR	Data terminal ready
8	DCD	Data carrier detect	21	SQ	Signal quality detector
9	–	Positive voltage	22	RI	Ring indicator
10	–	Negative voltage	23	DRS	Data rate selector
11	–	Not assigned	24	SCTE	Clock transmit external
12	SDCD	Secondary DCD	25	BUSY	Busy
13	SCTS	Secondary CTS			

Serial standard (3)

❑ Alternative connectors

- Since RS-232 is overkill for all real-world situation
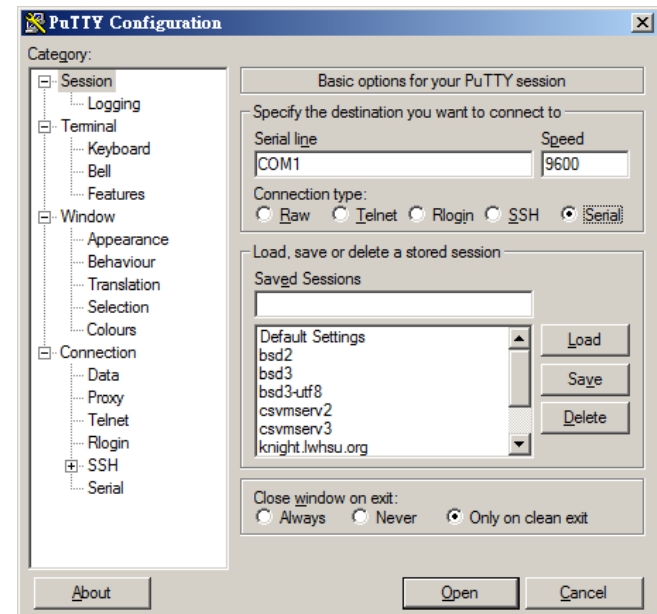 - ➢ Mini DIN-8
  - ➢ DE-9
  - ➢ RJ-45

Serial standard (4)

❑ Cable Length

- RS-232 specifies a maximum length of 75 feet at 9600 bps
  - ➢ 75 * 30.5 ≒ 22 m
- In reality, they hit the limit between 800 ~ 1000 feet

Serial Console

❑ /boot/loader.conf

- console="vidconsole,comconsole"

❑ Connect

- PuTTY
- tip(1)
- comms/minicom

❑ http://www.freebsd.org/doc/en/books/handbook/serialconsole-setup.html

Serial Device File

❑Serial ports are represented by device files under /dev

❑The name of the device file is no big deal

- behavior is determined by the major and minor device number

System	Device files for the first two serial ports
FreeBSD	/dev/ttyu[0,1] (com 1, com 2)
Red Hat	/dev/ttyS[0,1]
Solaris	/dev/term[a,b]
SunOS	/dev/tty[a,b]

```
liuyh@NASA ~ $ ls -l /dev/ttyu0*
crw------- 1 root  wheel   0,  39 Sep 25 10:57 /dev/ttyu0
crw------- 1 root  wheel   0,  40 Sep 25 10:57 /dev/ttyu0.init
crw------- 1 root  wheel   0,  41 Sep 25 10:57 /dev/ttyu0.lock
```

Kernel Configuration

❑ Kernel configuration file

- device uart

❑ dmesg

- % grep uart /var/run/dmesg.boot (8.x)

> **uart0: <16550 or compatible> port 0x3f8-0x3ff irq 4 flags 0x10 on acpi0**
> **uart0: [FILTER]**
> **uart1: <16550 or compatible> at port 0x2f8-0x2ff irq 3 on isa0**
> **uart1: [FILTER]**

❑ Kernel Module

- % kldload uart
- uart_load="YES" in /boot/loader.conf
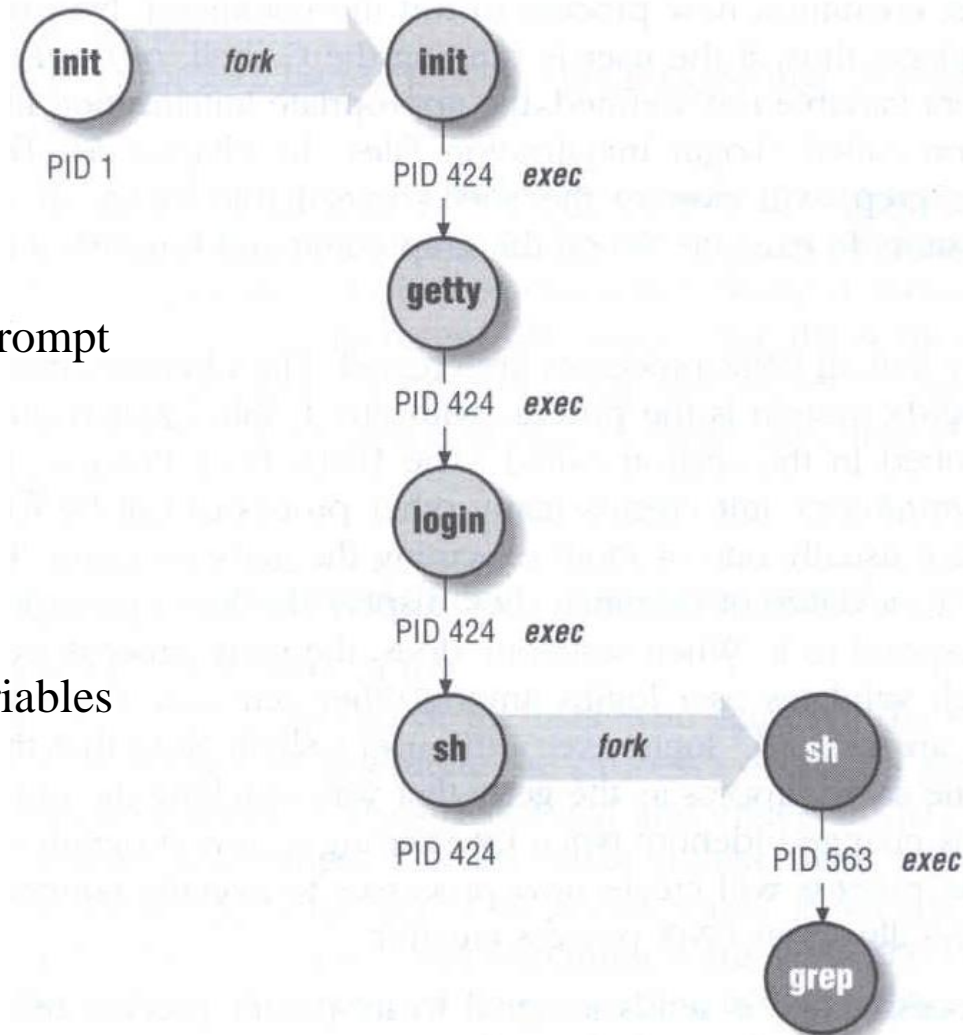
Configuration of Hardwired Terminals (1)

❑ Two main tasks

- Make sure each process is attached to a terminal to accept logins
- Make sure that information about the terminal is available once a user login

Configuration of
Hardwired Terminals (2)

❑ The login process

- init spawn getty according to /etc/ttys

- getty sets the port's initial characteristics and print the prompt

- User enter login name

- getty executes login program

- login request password

- login prints /etc/motd

- login sets up environment variables

- login runs a shell for user

- login(1), getty(8)

Configuration of Hardwired Terminals (3)

❑ Terminal Configuration Files

- On/Off
  - ➢ whether the terminal should be run a getty
- Term type
  - ➢ virtual console, network, dial-in
- Parameter
  - ➢ Terminal parameters, such as speed

System	On/Off	Term Type	Parameters	Monitor
FreeBSD	/etc/ttys	/etc/ttys	/etc/gettytab	getty
Red Hat	/etc/inittab	/etc/ttytype	/etc/gettydefs	getty
SunOS	/etc/ttytab	/etc/ttytab	/etc/gettytab	getty
Solaris	_sactab	_sactab	zsmon/_pmtab	ttymon

Configuration of Hardwired Terminals (4)

❑ FreeBSD: /etc/ttys

- Format

  device program termtype {on|off} [secure]

- Restart init process

  ➢ kill -1 1

  ➢ kill -HUP 1

- ttys(5)

```
#name    getty                          type     status  comments
ttyv1    "/usr/libexec/getty Pc"        cons25   on      secure
ttyv2    "/usr/libexec/getty Pc"        cons25   on      secure
ttyd0    "/usr/libexec/getty std.9600"  dialup   off     secure
ttyd1    "/usr/libexec/getty std.9600"  dialup   off     secure
ttyp0    none                           network
ttyp1    none                           network
```

Configuration of Hardwired Terminals (5)

❑ FreeBSD: /etc/gettytab

- Associate symbolic names with port configuration information, such as speed, parity, prompt

- man gettytab

```
default:\
        :cb:ce:ck:lc:fd#1000:im=\r\n%s/%m (%h) (%t)\r\n\r\n:sp#1200:\
        :if=/etc/issue:
2|std.9600|9600-baud:\
        :np:sp#9600:
P|Pc|Pc console:\
        :ht:np:sp#115200:
```

Special Characters and The terminal driver

❑ The terminal driver supports several special function when typing special keys

Name	Default	Function
Erase	^H	Erases one character of input
WErase	^W	Erases one word of input
Kill	^U	Erases the entire line of input
EOF	^D	Sends an "end of file" indication
INTR	^C	Interrupts the currently running process
Quit	^\	Kills the current process with a core dump
Stop	^S	Stops output to the screen
Start	^Q	Restarts output to the screen
Discard	^O	Throws away pending output
Suspend	^Z	Suspends the current process
LNext	^V	Interprets the next character literally

stty –
Set Terminal Options

❑ Change and query various settings of the terminal drivers

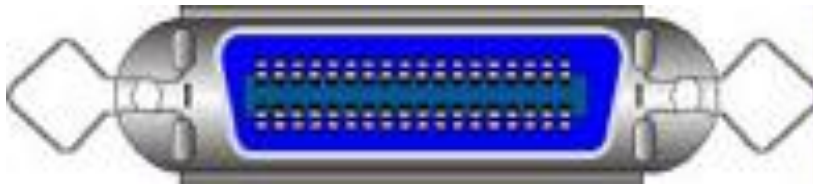- There are about a zillion options
- tty(4), stty(1)

❑ Example

- stty intr "^C" kill "^U" erase "^H"
- stty -a
- reset tty
  - ➢ reset
  - ➢ stty sane

```
speed 38400 baud; 24 rows; 80 columns;
lflags: icanon isig iexten echo echoe -echok echoke -echonl echoctl
    -echoprt -altwerase -noflsh -tostop -flusho pendin -nokerninfo
    -extproc
iflags: -istrip icrnl -inlcr -igncr ixon -ixoff ixany imaxbel -ignbrk
    brkint -inpck -ignpar -parmrk
oflags: opost onlcr -ocrnl -oxtabs -onocr -onlret
cflags: cread cs8 -parenb -parodd hupcl -clocal -cstopb -crtscts -dsrflow
    -dtrflow -mdmbuf
cchars: discard = ^O; dsusp = ^Y; eof = ^D; eol = <undef>;
    eol2 = <undef>; erase = ^?; erase2 = ^H; intr = ^C; kill = ^U;
    lnext = ^V; min = 1; quit = ^\; reprint = ^R; start = ^Q;
    status = ^T; stop = ^S; susp = ^Z; time = 0; werase = ^W;
```

Other Common I/O ports (1)

❑ Parallel ports

- Similar to serial ports in concept, but parallel ports transfer 8 bits of data at once

- IEEE-1284 standard

- Male DB25 ←→ male Centronics connector

Female Centronics connector

Male Centronics connector

Other Common I/O ports (2)

❑ USB – Universal Serial Bus

- Up to 127 devices can be connected
- Standardized connectors
- Devices can be connected and disconnected without powering down
- Up to 12Mb/s

❑ USB 2.0

- Up to 480Mb/s

❑ USB 3.0 (USB 3.1 Gen1)

- Up to 5Gbps

❑ USB 3.1 (USB 3.1 Gen2)

- Up to 10Gbps

❑ USB 3.2