I/O süsteemid

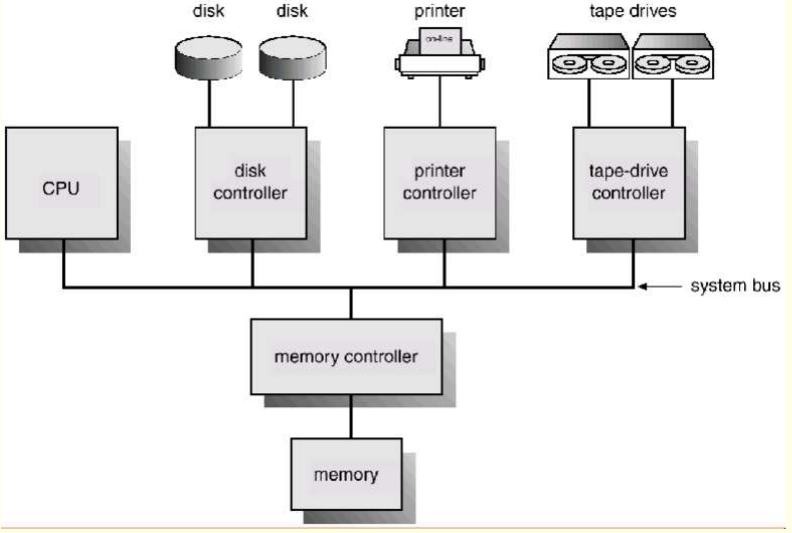
Ülevaade

- Riistavara (pordid, siinid, kontrollerid, . . .)
- Rakenduste liides I/O kasutamiseks
- Tuuma I/O alamsüsteem
- Stream'id
- Jõudlus

IO riistvara

- IO seadmetega seotud riistvara koosneb neljast osast:
  - Siin (*Bus*) — protsessori ja seadmete vaheline ühendusliin; tihti jagatud erinevate seadmete vahel
  - Port — seadme ühenduspunkt süsteemiga; koosneb juhtimis-, staatus- ja andmeregistritest
  - Kontroller — seadme juhtkomponent; edastab siinist tulnud käsud füüsilisele seadmele, vahendab andmeid siini ja seadme vahel
  - Füüsiline seade ise
- Seadmete juhtimiseks IO-käsud
  - Sarnane protsessori käsustikuga, kuid igal seadmel erinev
- Seadmete selekteerimine hosti poolt läbi aadressite

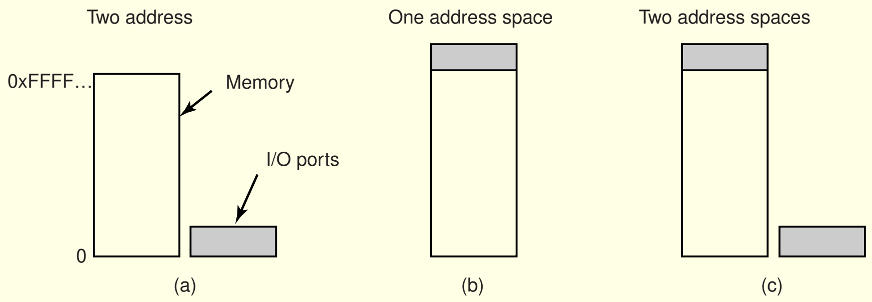
Arvutisüsteemi struktuur



Seadme registrid

- Host suhtleb kontrolleriga läbi registrite
- Kontrolleris on tüüpiliselt neli registrit:
  - *Status* — näitab, kas seade on hõivatud, andmed on valmis, tekkinud on viga, . . .
  - *Control* — seadme poolt järgmisena täidetav käsk
  - *Data-in* — seadme poolt saadetavad andmed
  - *Data-out* — seadme poolt vastu võetavad andmed
- Iga register on reeglina 1 – 4 baiti
- Sisuliselt on tegemist mäluga, millele protsessor pääseb ligi
  - Asub füüsiliselt mitte põhimälus, vaid kontrolleris
  - On eraldi aadressruumis (*Port I/O*) või kujutatud põhimäluga ühtsesse aadressruumi (*Memory Mapped I/O*)

Port IO vs. MMIO



- Port IO korral spetsiaalsed protsessorkäsud seadmeregistrite inspekteerimiseks/modifitseerimiseks
- MMIO korral on registritega manipuleerimine tavaline mälu poole pöödumine

Seadmete IO portide võimalikud aadressid PC-s

IO aadessvahemik	IO seade
000 – 00F	DMA kontroller
020 – 021	katkestusete kontroller
040 – 043	kell
2F8 – 2FF	sekundaarne järjestikport (COM2)
320 – 32F	HDD kontroller
378 – 37F	paralleelport (LPT)
3D0 – 3DF	graafika kontroller
3F0 – 3F7	FDD kontroller
3F8 – 3FF	primaarne järjestikport (COM1)

Seadmega suhtlemine

- Programmeeritud IO (PIO, *Programmed I/O*) — andmevahetus seadme ja põhimälu vahel toimub läbi protsessori
  - ”Küsitlus” — *Polling*
  - Katkestused — *Interrupts*
- Otsene mäluga opereerimine (DMA, *Direct Memory Access*) — andmevahetus seadme ja põhimälu vahel toimub otse ilma protsessori poole pöördumata
  - Võimaldab protsessoril andmevahetuse ajal tegeleda muu tööga
  - Vajalik DMA kontrolleri olemasolu

Seadmega suhtlemine — *Polling*

- Host ootab (hõivatult) staatusregistri taga, kuni hõivatusbit on püsti
- Host seab käsuregistris näit., kirjutusbiti ning kirjutab baidi väljundregistrisse
- Host seab käsuregistris valmisbiti
- Kontroller näeb, et kasuregistri valmisbit on püsti ning seab staatusregistris hõivatusbiti
- Kontroller näeb, et käsuregistris on kirjutamiskäsk ning teostab IO väljundregistrist füüsilisele seadmele
- Kontroller puhastab käsuregistri ning hõivatusbiti staatusregistrist

Seadmega suhtlemine — *Polling*

```
copy_from_user(buffer, p, count);                /* p is the kernel bufer */
for (i = 0; i < count; i++) {                    /* loop on every character */
    while (*printer_status_reg != READY) ;       /* loop until ready */
    *printer_data_register = p[i];               /* output one character */
}
return_to_user( );
```


Seadmega suhtlemine — katkestused

- Seadmedraiver algatab IO ning blokeerub
- Kontroller teostab IO, mille lõppedes genereerib katkestuse
- Protsessor, saades katkestuse kätte, annab juhtimise katkestusprotseduurile
- Katkestusprotseduur töötleb andmed; kui kogu IO tehtud, siis vabastab blokeerunud draiveri; vastasel korral algatab järgmise IO tsükli
- Protsessor jätkab katkestatud ülesande täitmist

Seadmega suhtlemine — katkestused

```
copy_from_user(buffer, p, count);
enable_interrupts( );
while (*printer_status_reg != READY) ;
*printer_data_register = p[0];
scheduler( );
```

(a)

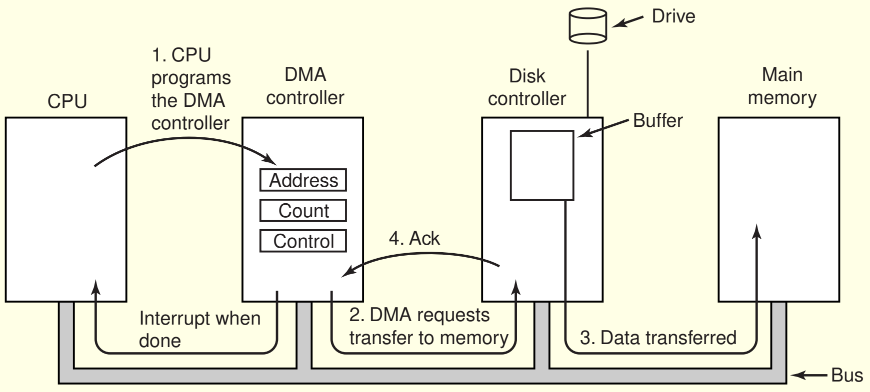
```
if (count == 0) {
    unblock_user( );
} else {
    *printer_data_register = p[i];
    count = count – 1;
    i = i + 1;
}
acknowledge_interrupt( );
return_from_interrupt( );
```

(b)

Katkestused

- Katkestuste haldamine toimub katkestuste kontrolleri poolt
- Lisaks seadmekontrolliritele põhjustavad katkestusi ka erindid (näit. nulliga jagamine) või püüniseed (so. tarkvaralised katkestused)
- Katkestusele vastava töötlemisprotseduuri määrab katkestusvektor
- Jagatavad katkestused vs. palju katkestusi
- Maskeeritavad ja mittemaskeeritavad katkestused (*NMI*)
- Katkestuste prioriteeditasemed

Seadmega suhtlemine — DMA



Seadmega suhtlemine — DMA

```
copy_from_user(buffer, p, count);
set_up_DMA_controller( );
scheduler( );
```

(a)

```
acknowledge_interrupt( );
unblock_user( );
return_from_interrupt( );
```

(b)

Rakenduste liides sisendile-väljundile

- Programmeerimisliides seadmeklasside kaupa
- Seadmedraiverite kiht peidab ära erinevate seadmete, siinide jms omapärad
- Seadmed erinevad mitmetes aspektides:
  - Plokkseadmed vs. märkseadmed
  - Järjestikpöördus vs. otsepöördus
  - Sünkroonne vs. asünkroone režiim
  - Jagatav või ühele kasutajale korraga
  - Kiirused erinevad paljude suurusjärkude poolest
  - Lugemiseks, kirjutamiseks, lugemiseks-kirjutamiseks

Plokkseadmed ja märkseadmed

- Plokkseadmed — andmete edastus toimub fikseeritud suurusega plokkide kaupa
  - Näit. kettaseadmed, lint
  - Käsud: `read`, `write`, `seek`
  - Failisüsteem vs. otsejuurdepääs (*raw I/O*)
- Märkseadmed — andmete edastus toimub baithaaval
  - Näit. klaviatuur, hiir, modem
  - Käsud: `get`, `put`
- Mitte kõik seadmed pole klassifitseeritavad plokk-/märkseadmeteks

Võrguseadmed, kellad, etc.

- Võrguseadmed — piisavalt erinev, et omada oma liidest
  - Enamasti ehitatud peale protokollistik ja rakendused kasutavad seda protokollistikku, mitte otse seadet
  - Soklid (*sockets*) — üks programmeerimisliides võrgurakendustele
  - `select()` liides mitme sokliga korraga tegelemiseks
  - Lisaks soklitele muid liideseid (torud (*pipe*), FIFOd, streamid, järjekorrad, postkastid)
- Kellad
  - Reaalajakellad on pea igas arvutis
  - Annavad jooksva aja, möödunud aja, taimerid
- `ioctl()` süsteemifunktsioon — tagauks otse draiverisse rakenduste tasemelt. Igal draiveril oma ioctl'id

Blokeeruv ja mitteblokeeruv I/O

- Blokeeruv I/O — protsess/lõim blokeeritakse kuni operatsiooni lõppemiseni või vea saamiseni (sünkroonne liides)
  - Blokeeritud protsessi äratab katkestus
  - Lihtne kasutada ja aru saada
  - Teatud juhtudel pole piisav
- Mitteblokeeruv I/O — protsessile tagastatakse nii palju kui on olemas
  - Väljastusväärtuseks loetud/kirjutatud baitide arv
  - Näit. kasutajaliides, andmete kopeerimine (puhverdatud I/O)
  - Reeglina realiseeritud lõimede abil kasutades ”küsitlust”
- Asünkroonne I/O — protsess algatab I/O ning jätkab täitmist
  - I/O alamsüsteem tevitab protsessi I/O valmis saamisest
  - Keeruline kasutada

Tuuma I/O alamsüsteem

- I/O planeerimine
  - I/O nõuete järjestamine kasutades seadmejärjekordi
  - Efektiivsus, õiglus
- Puhverdamine — andmete salvestamine mällu seadmete vaheliseks andmevahetuseks
  - Seadmete erinevad kiirused
  - Erinevad plokisuurused
  - Kopeerimissemantika
- Vahemälu (*cache*) — kiire mälu, mis hoiab koopiat andmetest
  - Alati ainult koopia
  - Hädavajalik efektiivseks andmevahetuseks

Tuuma I/O alamsüsteem

- Spuulimine (*spooling*)
  - Seadmetes, mis saavad serveerida korraga ainult üht nõuet
  - Näit. printer
- Seadmete reserveerimine
  - Süsteemifunktsioonid seadme hõivamiseks ja vabastamiseks
  - Võivad tekkida tupikud
- Veatöötlus (errno vs. keerulisemad asjad)
- Seadmete numbrid (seadme klass ja seadme alamnumber) — vanasti tabelitena, uuemal ajal dünaamilisem

STREAMS

- UNIX System V liides draiverikoodi dünaamiliselt kombineerimiseks
- *Stream head* — liidestub kasutajaprotsessiga
- *Driver end* — liidestub seadmega
- Vahele saab dünaamiliselt `push`'ida draiverimooduleid — rea redigeerimist, võrguprotokolle, ...
- Vookontroll streami sees
- getmsg()/putmsg()

Jõudlus

- Kontekstivahetuste arvu minimeerimine
- Andmete kopeerimiste arvu vähendamine
- Katkestuste sageduse vähendamine suuremate plokkide kaudu transportimise abil
- Paralleelsuse suurendamine DMA kasutamise abil
- Tõsta rohkem tööd riistvarasse
- Balansseerida protsessori, mälu alamsüsteemi, siinide ja I/O seadmete jõudlus, et osa süsteemist tühikäigul ei oleks