

FKEF.02.143

Arvuti arhitektuur

Computer Architecture and Organization



dotsent
Toomas Plank

©Toomas Plank, 2008

FKEF.02.143

Arvuti mälu



10. loeng,
2. mai 2008

©Toomas Plank, 2008

Jutujuht

Tänases loengus räägime



- staatilisest suvapöördusmälust
- dünaamilisest suvapöördusmälust
- püsimälust

Järgmises loengus räägime

- Vahemälust

Ülejärgmises loengus räägime

- Virtuaalmälust



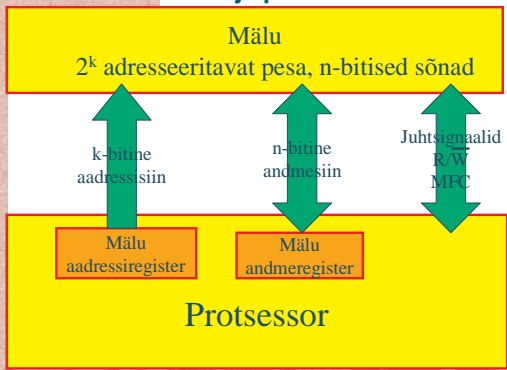
Arvuti arhitektuur FKEF.02.143

Mälust üldiselt (1)

- Ideaalne mälu on
 - kiire,
 - suur
 ja
 - maksab vähe
- Need soovid on kahjuks vastandlikud...
- Mälu maksimaalse suuruse määrab ära adresseeritav ala
 - 16-bitised aadressid võimaldavad kasutada $2^{16} = 65\,536$ mälupeasa
 - 32-bitised aadressid võimaldavad kasutada $2^{32} = 4\,294\,967\,296$ mälupeasa jne
- Infot salvestatakse ja loetakse enamasti sõnade kaupa
- Bittide arv, mida saab ühe käsuga mälust lugeda või mälu kirjutada on kõige tüüpilisemaks sõna pikkuse definitsiooniks

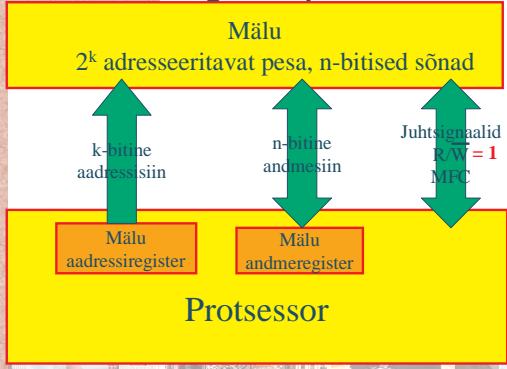
Arvuti arhitektuur FKFE02.143

Mälu ja protsessori ühendus

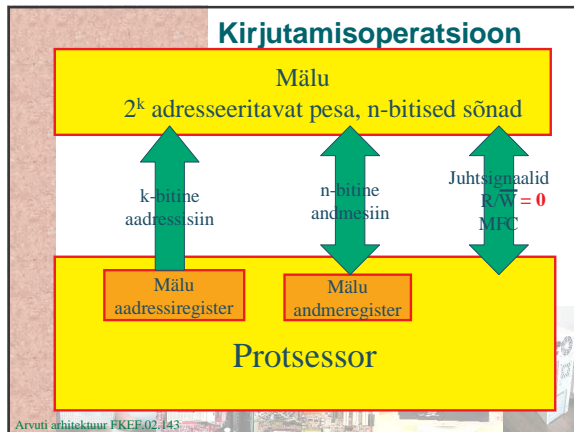


Arvuti arhitektuur FKFE02.143

Lugemisoperatsioon



Arvuti arhitektuur FKFE02.143



Mälust üldiselt (2)

- Mälu aadress võidakse ette anda
 - iga sõna jaoks
 - või ainult esimese sõna jaoks ploki
- Mäluga suhtlust võivad juhtida
 - taktsignaaliid
 - või spetsiaalsed siini juhtsignaalid

Arvuti arhitektuur FKFE02.143

Mälust üldiselt (3)

- Mälu pöördumisaeg (*memory access time*)
 - Ajavahemik *Read/Write* ja *Memory Function Completed* signaalide vahel
- Mälutsükli kestus (*memory cycle time*)
 - Ajavahemik kahe järjestikuse *Read/Write* signaali vahel
- Suvapöördusega mälu (RAM – *random access memory*)
 - kõigis mälupesades sisalduv info saadakse kätte ühesuguse kiirusega
- Jadapöördusseadmete puhul on andmete juurdepääsu aeg sõltuvuses konkreetse andmete asukohast/aadressist

Arvuti arhitektuur FKFE02.143

Vahemälu ja virtuaalmälu

- Protsessor suudab reeglina infot kiiremini töödelda kui seda võimaldab tavaline mäluotsükli kestus
- Üheks võimaluseks mäluotsükli kestust lühendada on vahemälu (*cache memory*) kasutamine
 - väike aga kiire mälu, mis on paigutatud protsessori ja põhimälu vahele
 - selles hoitakse programmi jaoks aktiivset infot (nii käsked kui ka andmeid)
- Mälu näiva mahu suurendamist võimaldab virtuaalmälu tehnoloogia
 - põhimälu hoitakse ainult aktiivset osa virtuaalsest aadressiruumist
 - Ülejäänud paikneb aeglasemal andmekandjal, tavaliselt kõvakettal
 - Programmi töö käigus aktiivse piirkonna asukoht muutub ja vastavalt uuakse ka kõvakettalt infot põhimälu (ja visatakse sealt midagi heikel vähem olulist minema)

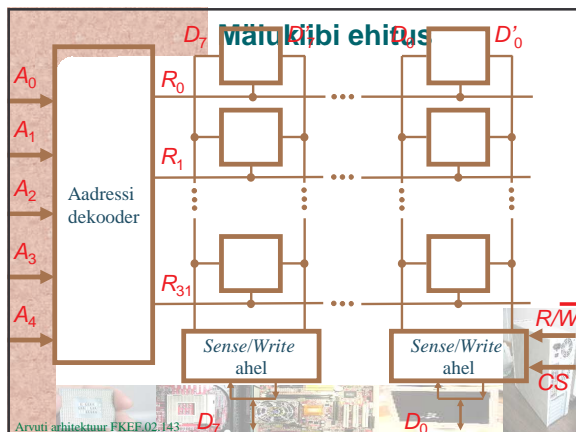
Arvuti arhitektuur FKEE02.143

Pooljuhtmaterjalist suvapöördusmälu (RAM)

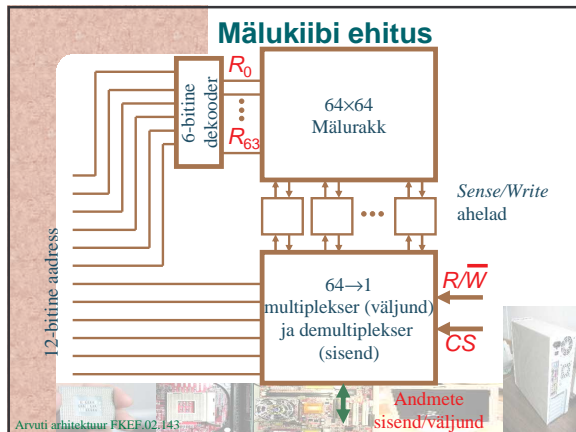
- RAM (*Random Access Memory*)
 - võeti kasutusele eelmise sajandi kuuekümnendate aastate lõpus
- Mälukiibi ehitus
 - mälurakud paigutatakse enamasti massiividena, kus iga element suudab salvestada ühe biti infot

Arvuti arhitektuur FKEE02.143

Mälukiibi ehitus



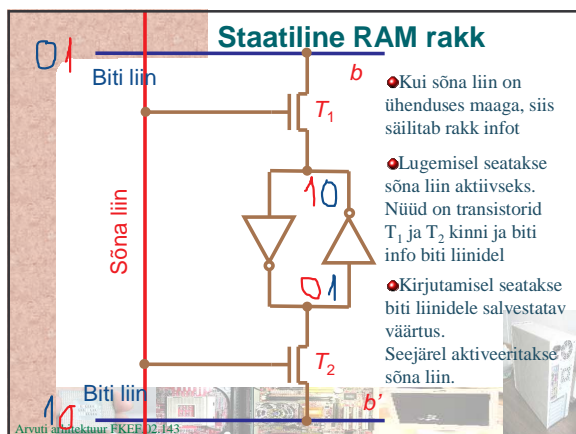
Arvuti arhitektuur FKEE02.143

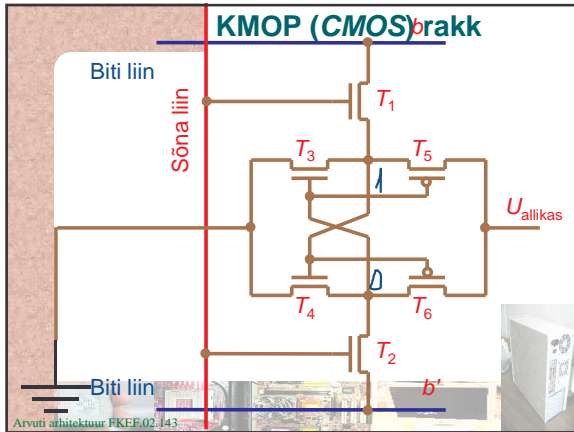


SRAM

- SRAM (*Static Random Access Memory*)
- Need mäلود suudavad säilitada infot seni kuni nad saavad voolu
- Voolu väljalülitamisel info kaob

Arvuti arhitektuur FKEE.02.143

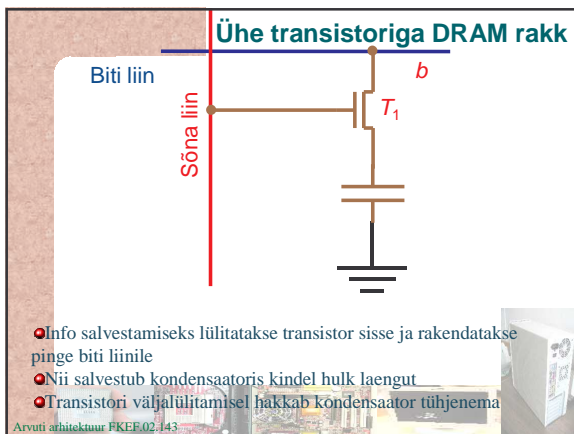




Asünkroonne DRAM

- Staatilised mä-lud on kiired, aga kallid
 - igas rakus palju transistore
- DRAM (*Dynamic Random Access Memory*)
- Info salvestatakse kondensaatori laenguna
- DRAM'id on
 - oluliselt odavamad
 - info nendes kustub teatava aja möödudes
 - vajavad värskendamist
- Laeng säilib kondensaatoril mõned kümned millisekundid

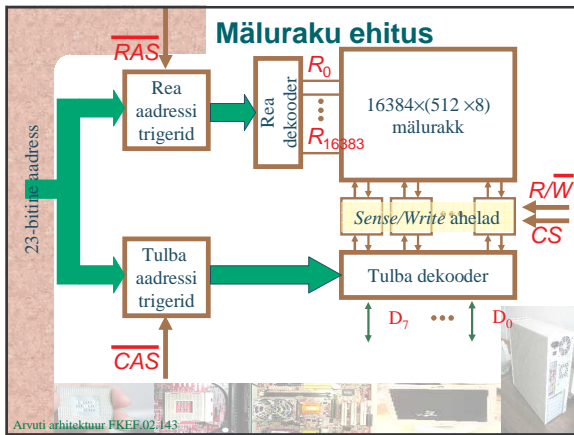
Arvuti arhitektuur FKKEE.02.143



Lugemine

- Info lugemisel lülitatakse transistor sisse
- Biti liiniga ühendatud võimendi saab aru, kas kondensaator on laetud üle läveväärtuse või mitte
 - kui on, siis ühendatakse biti liin pingevalikuga ja laetakse kondensaator uuesti täis
 - mälurakus on salvestatud **loogiiline 1**
 - kui ei ole, siis ühendatakse biti liin maaga ja laetakse kondensaator täiesti tühjaks
 - mälurakus on salvestatud **loogiiline 0**
- Nagu näeme, värskendab mälu lugemine automaatselt mälus sisalduva info

Arvuti arhitektuur FKEE.02.143



Asünkroonne DRAM

- Värskendusahelad
 - sageli juba DRAM kiibi sees
 - sel juhul kasutaja ei tajugi mäluraku dünaamilist ehitust
- Seni vaadeldud DRAM rakkude juhtsignaalid olid asünkroonsed
 - spetsiaalne mälu juhtloogika ahel genereeris vajalikud juhtsignaalid
 - *asynchronous DRAM*

Arvuti arhitektuur FKEE.02.143

Fast page mode

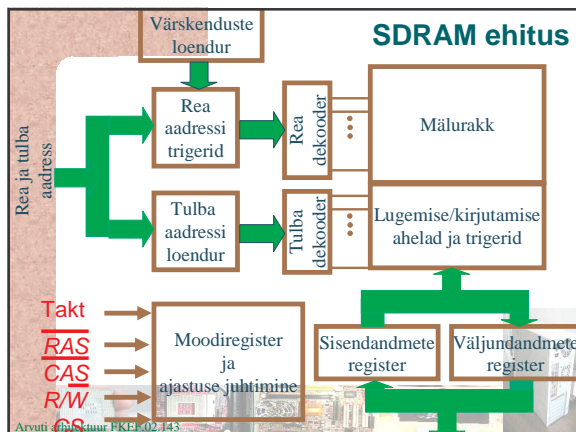
- Eelmisel slaidil kujutatud DRAM mooduli puhul aktiveeriti terve rida, aga reaalselt loeti ainult 8 bitti andmeid
- Lisades tulpade liinide otsa trigerid, saame salvestada kogu antud rea sisu ja pärast, ilma rida uuesti aktiveerimata, selle info edastada
- Üheks rakenduseks oleks plokkide kaupa (järjestikuste) andmete edastamine
 - järjestikused tulpade aktiveerimised järjestikuste CAS signaalide juhtimisel
 - *fast page mode*
 - see on oluliselt kiirem kui juhusliku mälupea lugemine
- Kus kasutatakse?
 - regulaarse mustriga mälu-lugemised (näiteks graafika terminalides)
 - andmete lugemisel vahemällu

Arvuti arhitektuur FKKE1.02.143

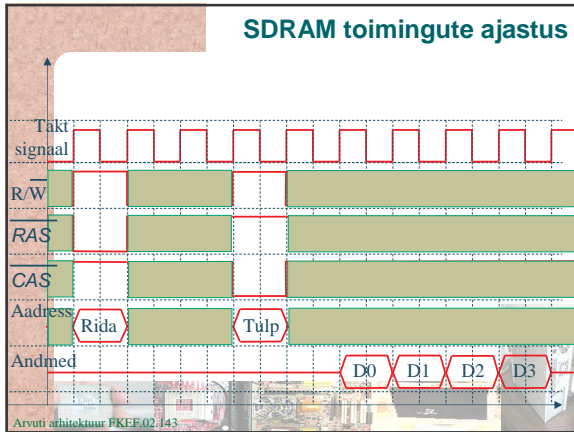
Sünkroonne DRAM

- *Synchronous Dynamic Random Access Memory* (SDRAM)
- Selle mälu suhtlus toimub sünkroonselt taktsignaaliga
- Ehitus väga sarnane asünkroonse DRAM'i ehitusega

Arvuti arhitektuur FKKE1.02.143



Arvuti arhitektuur FKKE1.02.143



Latentsusaeg ja läbilaskevõime

Latentsusaeg

- *Latency*
- Aeg, mis kulub ühe sõna ülekandmiseks mälust või mällu
- Andmete plokkide kaupa ülekandmisel on see aeg, mis kulub esimese sõna kättesaamiseks

Läbilaskevõime

- *Bandwidth*
- Bittide või baitide arv, mida on võimalik ajaühikus üle kanda (plokkide kaupa andmeid liigutades)
- seda numbrit mõjutavad nii mälu kui ka siini kiirus

Arvuti arhitektuur FKEE.02.143

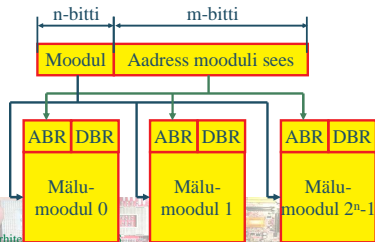
DDR SDRAM

- *Double Data Rate Synchronous Dynamic Random Access Memory (DDR SDRAM)*
- Tavalise SDRAM'i toimingud tehakse üks kord takti jooksul (näiteks taktsignaali tõusval frondil)
- DDR toiminguid tehakse kaks korda takti jooksul, taktsignaali mõlemal frondil
- **Latentsusaeg sellest ei lühene**
- **Küll aga kasvab läbilaskevõime**
- DDR, DDR2, DDR3

Arvuti arhitektuur FKEE.02.143

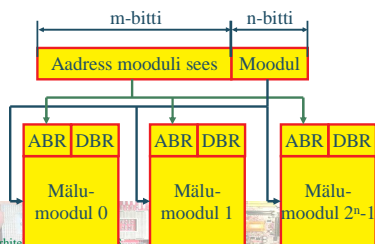
Vaheldamine (Interleaving)

- Kui meil on rohkem kui üks mälu moodul, siis võib olla võimalik suhelda nende moodulitega samaaegselt
- Oluline on seejuures aadresside jaotumine mälu moodulite vahel



Vaheldamine (Interleaving)

- Siin paiknevad järjestikused mälu aadressid eri moodulites
- Seega saab neid lugeda sisse samaaegselt



Näide: üks mälu moodul

- Olgu meil tarvis liigutada 8-sõnaline plokk põhimälust vahemälusse
- Lepime kokku ülesande lähteandmetes:
 - Aadressiinfo saatmine mälule võtab aega 1 takti
 - Esimese sõna saame mälust kätte 10 taktiga
 - Järjestikused sõnad saame mälust kätte 5 taktiga
 - Üks takt kulub veel sõna saatmiseks vahemälusse
- Ühe mooduli korral on info vahemälusse jõudmise aeg:

$$1 + 10 + (7 \times 5) + 1 = 47 \text{ takti}$$

Näide: neli mälmoodulit

- Olgu meil tarvis liigutada 8-sõnaline plokk põhimälust vahemälusse
- Lepime kokku ülesande lähteandmetes:
 - Aadressiinfo saatmine mälule võtab aega 1 takti
 - Esimesed sõnad saame mäludest kätte 10 taktiga
 - Üks takt kulub iga sõna saatmiseks vahemälusse
 - Järjestikused sõnad saame mäludest kätte 5 taktiga
 - Samal ajal saame lugeda mälust järjestikuse sõna andmeid ja edastada eelmise lugemise tulemuse vahemälule
- Nelja üheaegselt töötava mooduli korral on info vahemälusse jõudmise aeg:

$$1 + 10 + 4 + 5 + 4 = 20 \text{ takti}$$

Arvuti arhitektuur FKEE02.143



Värskendamisele kuluv aeg

- Kõigi dünaamiliste mälude sisu on vaja aeg-ajalt värskendada
- Kui palju selleks aega kulub?
- Näide
 - 16384 rida
 - 4 tsüklit rea andmete lugemiseks
 - taktsagedus 166 MHz
 - $16384 \times 4 / (166 \times 10^6) = 395 \times 10^{-6} = 0,395 \text{ ms}$
 - Kui värskendame iga 64 ms järel, siis täiendav ajakulu:
 $0,395 / 64 = 0,6\%$ mäluga suhtlemise ajast

Arvuti arhitektuur FKEE02.143

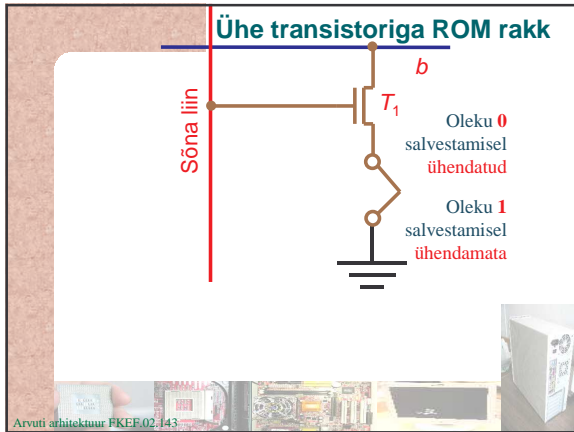


Püsimälu

- Vajadus sellise mälu järgi, mis säilitaks infot ka voolu väljalülitamisel
- ROM (*read only memory*)

Arvuti arhitektuur FKEE02.143





Püsimälu

- Vajadus sellise mälu järgi, mis säilitaks infot ka voolu väljalülitamisel
- ROM (*read only memory*)
- PROM (*programmable read only memory*)
- EPROM (*erasable programmable read only memory*)
- EEPROM (*electrically erasable programmable read only memory*)
- välmälu (*flash memory*)

Arvuti arhitektuur FKEE.02.143

Kasutatud kirjandus

- Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer organization 5th edition (2002) 805 p.
- Miles Murdocca, Vincent Heuring, Computer Architecture and Organization: An Integrated Approach (2007) 544 p.

Arvuti arhitektuur FKEE.02.143
