

Principper for Samtidighed og Styresystemer

Hardwareunderstøttelse

René Rydhof Hansen

April 2008

Husk!

Tilmelding til “ekstra-forelæsning” om memory management *før* 01 MAJ 2008.

Opgaver

- Opgave 1: Banker's Algorithm
- Opgave 2: Logiske adresser
- Opgave 3: Adresseoversættelse (direkte sidetabel)

- At kunne redegøre for “instruction fetch”-løkken
- At kunne redegøre for (logisk) placering og brug af MMU
- At kunne redegøre for brugen af diverse caching-mekanismer

- Styresystemet har behov for hjælp til
 - Adresseoversættelse
 - Adskillelse af adresserum
 - Håndtering af CPU exceptions
 - Interrupthåndtering
 - I/O
 - Timer
 - ...
- Hardwaren har behov for hjælp til
 - Opsætning af interrupttabeller
 - Sidetabeller
 - Demand paging
 - Processkifte
 - Scheduling (CPU og I/O)
 - ...

- Von Neuman-arkitektur
 - Processor og primær lager er adskilt
 - Programmet er i primær lager sammen med andre data
- Processoren
 - Læser instruktioner fra primærlager (fetch)
 - Udfører instruktionen (execute)
 - “Instruction fetch-løkken”
- Processorregistre
 - Instruktionsregister
 - Den instruktion processoren er ved at udføre
 - Programtæller
 - Adresse for næste instruktion
 - Statusregister
 - Stackpointer
 - Adresse for øverste element af stakken

“Instruction fetch”-løkken

- 1 **Start:** hent instruktion fra adresse givet ved programtælleren og gem instruktionen i instruktionsregisteret
- 2 Inkrementer programtælleren
- 3 Afkod instruktionen
- 4 Hent operander
- 5 Udfør instruktionen
- 6 Gem resultatet og opdater statusregisteret
- 7 Gå tilbage til start

Registre på Pentium

- EAX, EBX, ECX, EDX, EBP, ESI, EDI
 - Registre til generel brug for programmøren
 - Floating-point instruktioner bruger særlige FP registre
- EIP og ESP
 - Programtæller og stack pointer
- EFLAGS
 - Statusregister
- CD, DS, ES, FS, GS, SS
 - Segmentregistre (CS = code segment, SS = stack segment)
- Kontrolregistre
 - Interrupttabellen, page directory, segmenttabel, etc.

“Instruction fetch”

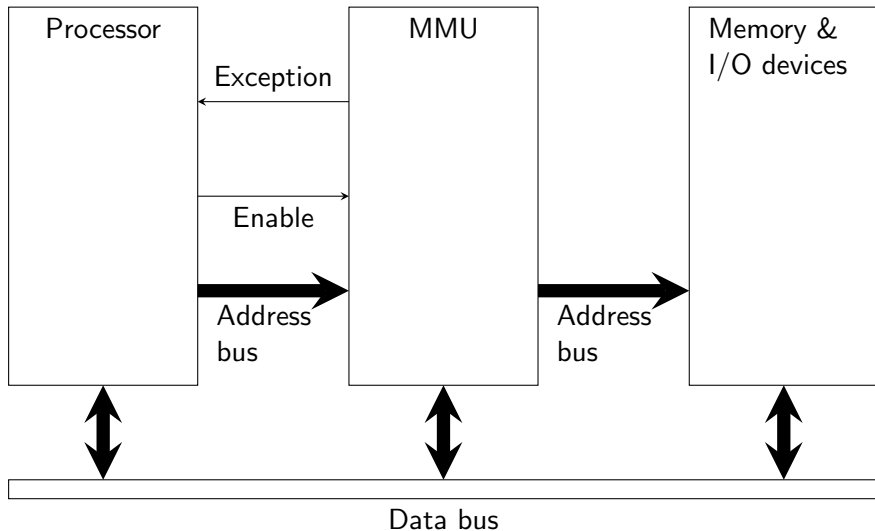
- Instruktioner udføres sekventielt
 - Programtælleren inkrementeres efter fetch
- Jump-instruktioner modificerer programtælleren
 - Svarer til goto i højniveau programmeringsprog
 - Kan være betingede af værdien af statusregisteret
- Call-instruktioner modificerer programtælleren
 - Gemmer tidligere værdi for programtælleren
 - `return`-instruktionen gendanner seneste gemte værdi
 - Bruges til at implementere procedurekald

- Lagerområde til
 - Returadresser
 - Lokale adresser
 - Parametre og returværdier
- Toppen af stakken udpeges af stack-pointeren
- På Pentium vokser stakken nedad
 - `push` dekrementerer stack-pointeren og gemmer værdi i ny adresse
 - `pop` henter værdi og inkrementerer stack-pointeren

Memory Management Unit (MMU)

- Virtuel hukommelse kræver hardware til adresseoversættelse
- Logisk placeret mellem processor og hukommelse
 - Oftest integreret på samme chip for moderne processorer
- MMU bruger side- og segment-tabeller til adresseoversættelsen
 - Kan tilgå nødvendige processorregistre til at finde placeringen af disse tabeller
 - Tabellerne vedligeholdes af styresystemet

Memory Management Unit



- Ved paging med 2-niveau sidetabel skal MMU:
 - ① Lave et opslag i sidekataloget (første niveau)
 - ② Lave et opslag i sidetabellen (andet niveau)
 - ③ Hente data fra eller skrive data til primærlageret

Da sidekataloget og primærtabellen ligger i primærlager bliver adgangstiden til primærlager **tre gange** længere end adgangstiden uden virtuel hukommelse... I bedste fald (uden sidefejl)!

Translation Lookaside Buffer

- Translation Lookaside Buffer (TLB)
 - Cache som MMU'en bruger til adresseoversættelse
 - Gemmer seneste sidetabelsopslag
 - Virker på grund af **lokalitetsprincippet**
- Opslag i TLB og i sidetabellen kan parallelliseres
- TLB vedligeholdes typisk af MMU'en
 - Softwarestyret TLB kan forekomme
 - Giver styresystemet mulighed for at vælge sidetabelsorganisering
 - MMU'en må tømmes helt eller delvist når sidetabellen skifter eller ændres
 - Fx ved processkifte eller swapping
 - Gøres af styresystemet

Translation Lookaside Buffer

- Nogle få hundrede indgange placeret i MMU
- Associativ hukommelse
- Meget hurtig, men dyr

- Processorer har ofte cache til indholdet af primærlager
- Ikke det samme som TLB!
- Ofte opdelt i 1. og 2. niveau (level) cache
 - Evt. 3. niveau kan forekomme
 - Niveau 1 kan være opdelt i instruktionscache og datacache
 - Hent fra 1. niveau hvis muligt, ellers prøv 2. niveau, ellers primærlager
- Skrivning
 - **Write-through**: Skriv direkte til primærlager
 - **Write-back**: Skriv til cache og opdater primærlager senere

Pop-Up Test #1

- Prøven er **anonym**
- Multiple-choice
- Tid: 30 minutter

Opsummering og næste gang

- Moderne arkitektur
- “Instruction fetch”-løkken
- MMU'en
- Næste gang: Interrupts og kernen