
Operativsystem

Historia
(1.2 i boken)

1940-talet

- *Inga operativsystem*
- *Programmering via mekaniska brytare, bit för bit, i maskinkod*
- *Senare kom hålkort in i bilden, programmen matades via hålkort*
- *Senare utvecklades assembler, med engelsk-liknande förkortningar*
- *2:a världskriget hade stor inverkan på utveckling av datamaskiner*
- *transistorn uppfanns 1947!*
- *Howard Aiken, John von Neuman, Konrad Zuse...*

1950-talet



- *GM Research Laboratories utvecklade tidigt på 50-tal ett operativsystem för IBM 701, första??*
- *Tidiga system körde ett "jobb" åt gången, med operatör inblandad*
 - *Behov av "batch"-system*
- *Batch-system*
 - *Jobben lästes in på t.ex. magnetisk media, varefter jobben kördes i en serie (batch)*
- *Programmerare var ofta fortfarande tvungna att själva kontrollera systemets resurser*

1960-talet

- *Batch-system, men man använder systemet mer effektivt genom körning av flere job "samtidigt"*
 - *I/O-intensiva jobb*
 - *CPU-intensiva jobb*
- *Kallas även multiprogrammerbara system*
 - *"Degree of multiprogramming" – hur många job som samtidigt kan hanteras*
- *IBM S/360 - 1964*
 - *Hårdvarukompatibel*
 - *OS/360 – operativsystem för kompatibilitet*



1960-talet

- *Flera samtidiga användare*
 - *CTSS (Compatible Time-Sharing System) - MIT*
 - *Project MULTICS - MIT, GE, Bell Labs*
 - *Misslyckades, mycket p.g.a. dålig marknadsföring*
 - *Dock stort inflytande på dagens operativsystem*
 - *Bl.a. UNIX bottnade i MULTICS*
 - *MINIX (Tanenbaum)*
 - » *Linux*
- *Kostnader*
 - *CPU-timme på mainframe: \$500*
 - *Timlön för programmerare: \$4*

1970-talet

- *Tidsdelande system (timesharing) börjar bli standard*
- *tidiga persondatorer*
- *TCP/IP-standarden börjar bli allmänt använd*
- *LAN kommer för nätverk*
- *Enkryptering börjar bli relevant*

1980-talet

- *Persondatorer (PC)*
 - *IBM 1981*
 - *Macintosh 1984*
- *GUI i operativsystemet*
 - *X-windows (MIT)*
 - *Macintosh*
- *Distribuerade system – klienter och servrar*

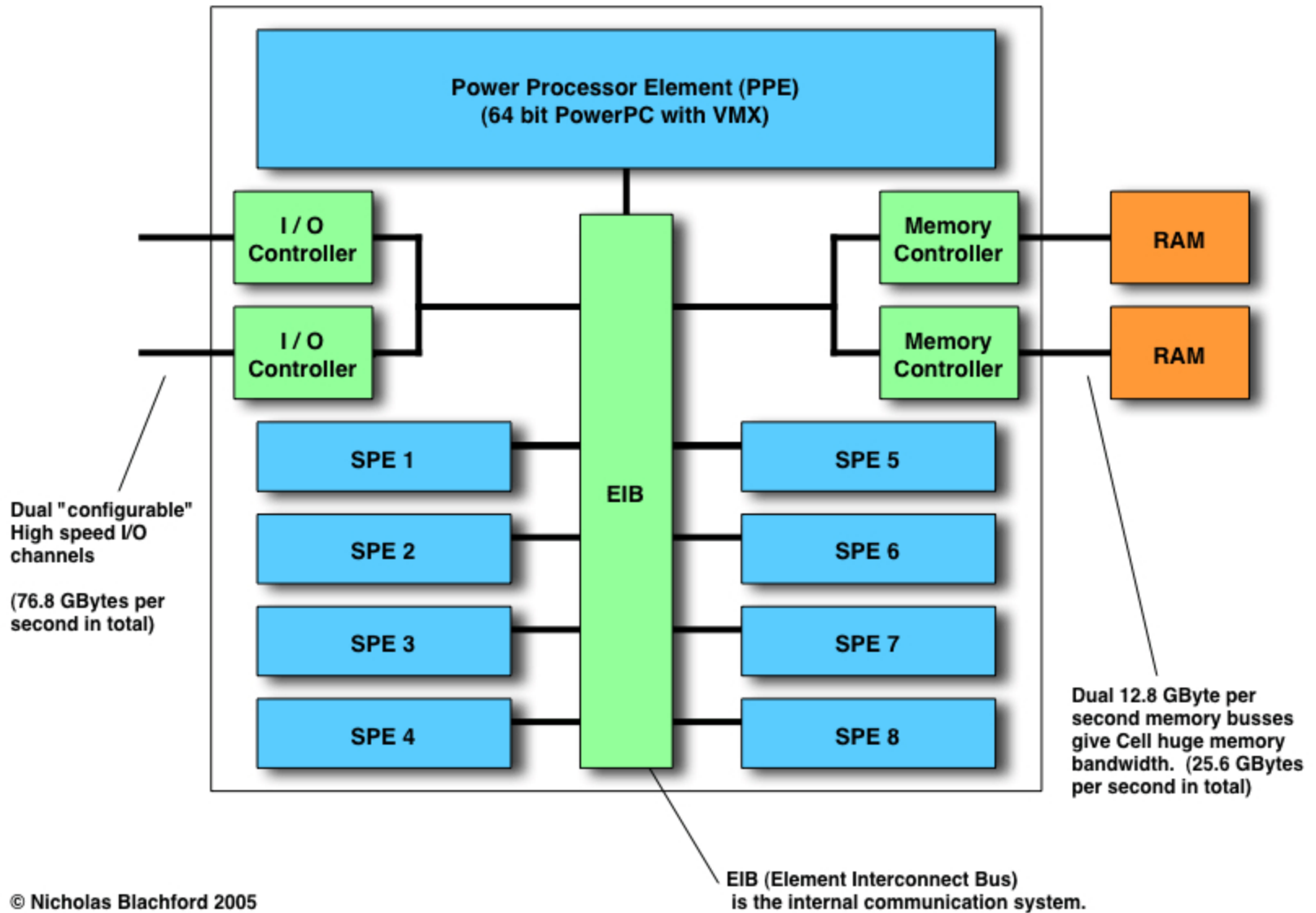
1990-talet

- *WWW*
 - *”Distribuerade system”*
- *Nätverksstöd blev standard i operativsystem*
- *Användarvänligheten ökar*
- *Objekt-orienterade operativsystem*
- *Open-source-rörelsen*

2000-talet

- *Middleware: länkar två applikationer*
 - *T.ex. mellan web-server och databassystemet*
- *Distribuerade system*
 - *T.ex. virtuella butiker på nätet*
 - *Förmedlar varor från något annat system*
- *Standardiserade användargränssnitt i operativsystem*
- *Operativsystem kliver in i många nya apparater, såsom PDA, mobiltelefoner*
- *Multi-core processorer, många processorer i en apparat, cell-processorer (playstation 3: 2 teraflops)*

Cell Processor Architecture

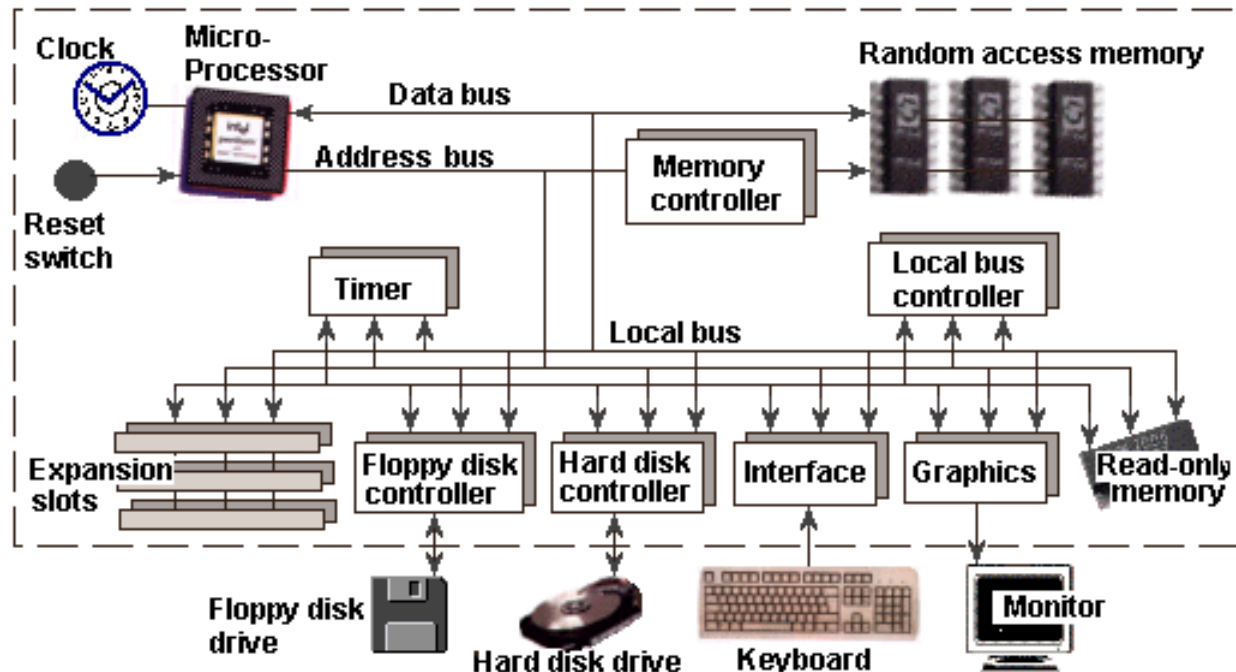


Operativsystem

Hårdvara
(1.4 i boken)

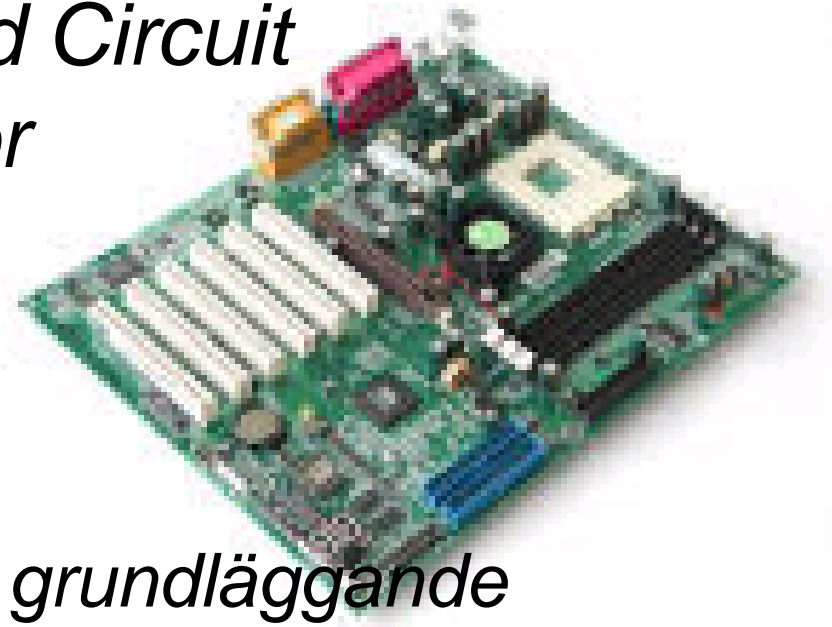
Hårdvaruöversikt

- *Generellt: CPU:n, minnet och I/O-enheter är uppkopplade till varandra via en buss (t.ex. PCI)*



Moderkort

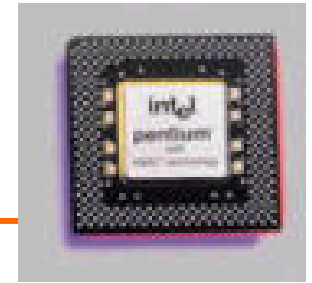
- *Centrala PCB (Printed Circuit Board) i en bordsdator*
- *Flere chips, bl.a. för BIOS (Basic Input Output System)*
 - *Chipset (innehållande grundläggande arkitekturkomponenter)*
 - *Front Side Bus, avbrottshantering, minnesstyrning, nätverksadapter, hårddiskadapter, ljud, grafik*



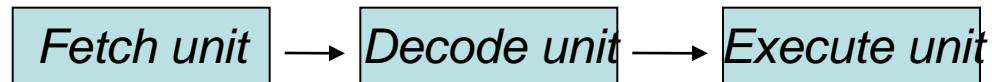
Busstyper

- *En buss är en samling elektriska kopplingar*
 - *databuss*
 - *adressbuss*
- *Front Side Bus (FSB)*
 - *Kopplar bl.a. processorn till minnet*
- *Peripheral Component Interconnect (PCI) buss*
 - *Hårdvaruenheter, såsom t.ex. ljudkort*
- *Accelerated Graphics Port (AGP)*
 - *För att koppla grafikkort till systemet*

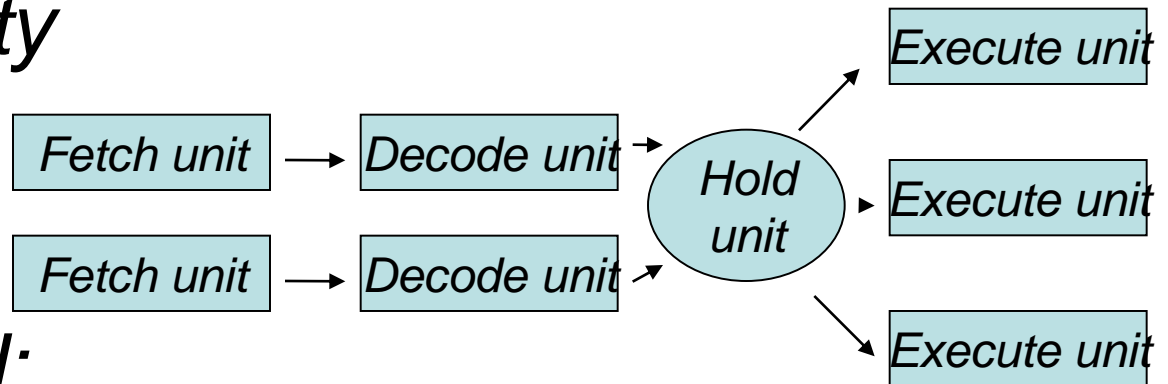
CPU



- *Register: PC, SP, generella register, PSW*
- *Pipeline*

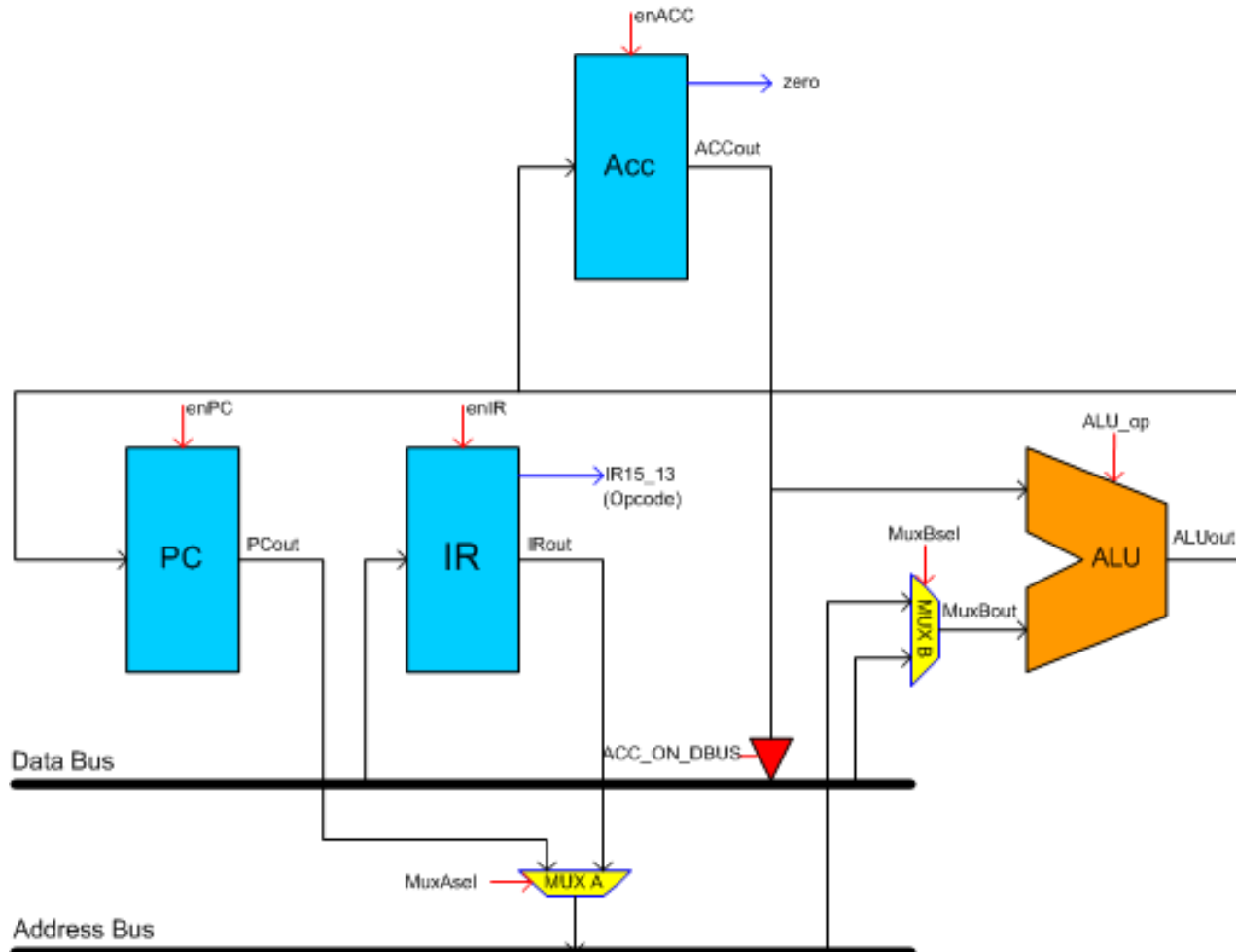


- *Superscalarity*



- *CPU tillstånd:*
 - *Minst 2: Kernel mode, user mode*

CPU - simplifierat



Minnestyper

	Söktid	Storlek	Pris (€/GB)
<i>Register</i>	<i>1 ns</i>	<i>1 KB</i>	<i>???</i>
<i>Cache</i>	<i>2 ns</i>	<i>1 MB</i>	<i>50000</i>
<i>Primärminne</i>	<i>10 ns</i>	<i>512MB-6GB</i>	<i>20</i>
<i>Magnetisk skiva</i>	<i>10 ms</i>	<i>100-1000 GB</i>	<i>0,10</i>
<i>Magnetiskt band</i>	<i>100 s</i>	<i>20-100 GB</i>	<i>0,10</i>
<i>Optisk skiva</i>	<i>500 ms</i>	<i>5 GB</i>	<i>0,20</i>

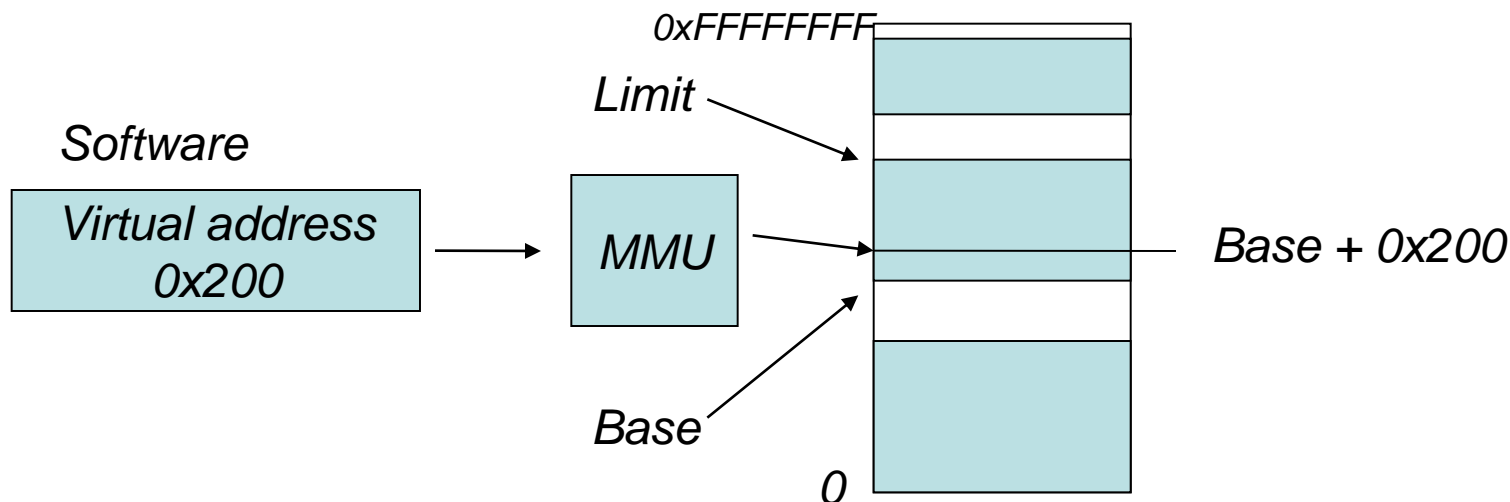
Types of main memory:

RAM: SIMM, DIMM, CMOS

ROM: EPROM, EEPROM

Primärminne

- *Måste hantera multiplexing i rymden:*
 - 1. *Hur skydda program från varandra*
 - 2. *Hur se till att program kan flyttas i minnet*



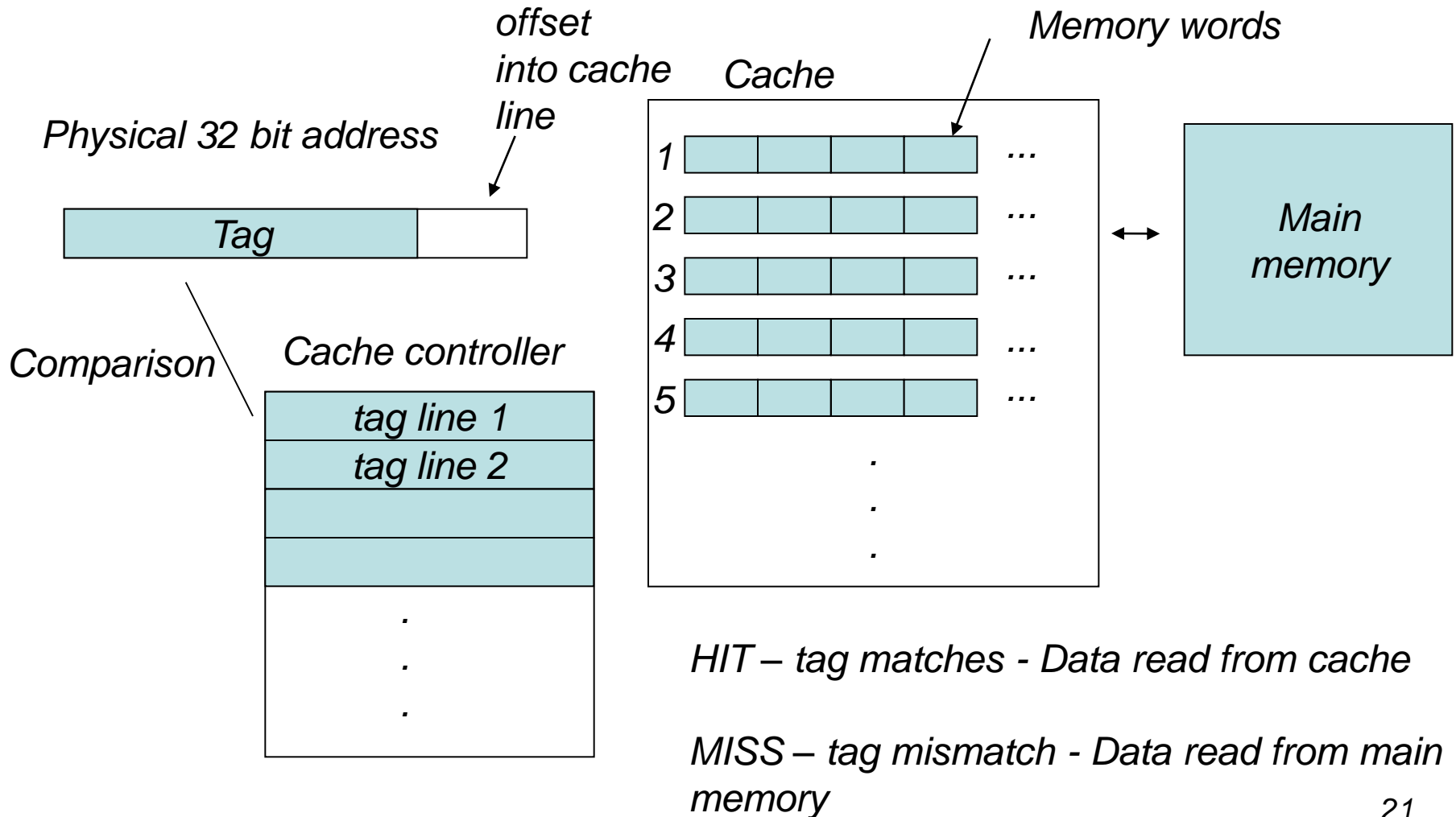
Hårdvaru-cache

- *CPU är mycket snabbare än primärminne*
 - *CPU väntar många klockcykler på data*
- *Lokala egenskaper vid minnesaccess:*
 - *Framtida minnesreferenser ofta i närheten av de nuvarande (t.ex. kod eller räckor)*
- *Mindre och snabbare minnesenheter innehållande senast använda kod/data*
- *Cache-linjer innehållande senast använda bytes*

Hårdvaru-cache

- *Fully mapped*
 - *A line in main memory is always stored in the exact same location in cache*
- *Fully associative*
 - *A line in memory can be stored in any location in the cache*
- *N-way associative (the normal way)*
 - *A line in memory can be stored in any of N lines of the cache*

Hårdvaru-cache



DMA

- *Direct Memory Access*
- *De flesta I/O-operationer transporterar data mellan centralminnet och I/O-enheter*
 - *Detta sköttes länge genom det som kallas Programmet I/O (PIO)*
 - *Processorn måste se till att varje byte förflyttas från minne till I/O-enhet*
- *DMA tillåter direkt kopiering av data från I/O-enhet till minnet, utan intervention från processorn*

I / O-enheter

- *Disc (IDE, EIDE, SCSI (Ultra320: 320 Mb/s), SATA I: 1,5 Gb/s, SATA II 3 Gb/s)*
- *Floppy*
- *Screen*
- *Keyboard*
- *Serial (115 kb/s), Parallell*
- *USB (1.1: <12 Mb/s, 2.0: 480 Mb/s)*
- *Firewire (800 Mb/s)*
- *WUSB, UWB, Bluetooth, Wibree, ZigBee*

Hårdvarusupport för OS

- *Processor:*
 - *User mode, Kernel mode, privileged instructions*
- *Skyddning samt hantering av minnet*
 - *Gränser för minnesområdet, virtuellt minne*
- *Avbrott och exeptioner*
 - *I/O-enheter kan "meddela" systemet att något har hänt*
- *Timers och klockor*
 - *För att t.ex. med jämna mellanrum skapa avbrott*
- *Bootstrapping*
 - *System i BIOS för att överhuvudtaget få något program att snurra i en dator*

Maskinspråk, assembler

- *En dator (läs CPU) förstår enbart maskinspråk*
 - *Processorspecifikt*
 - *I grunden 1:or och 0:or:*
 - 1300042774
1400593419
1200274027
- *Assembler (assembly language)*
 - *Engelsk-liknande förkortningar motsvarande maskinspråket*
 - LOAD BASEPAY
ADD OVERPAY
STORE GROSSPAY

Tolkar och kompilatorer

- *Maskinspråk / assembler tar lång tid att skriva*
 - *Högnivå-språk utvecklas (t.ex. C/Java/Visual Basic, C++...)*
- *Typer av högnivåspråk*
 - *Tolkade språk (Visual Basic, Python, PHP)*
 - *Tolkas i samband med exekveringen till mer primitiva byggstenar*
 - *Kompilerade språk (C/C++/Fortran)*
 - *Översätts på förhand med hjälp av kompilator till maskinspråk*
 - *Byte-kod (Java)*
 - *Kompilerat språk, där "maskinspråket" är Javas virtuella maskin (JVM)*
 - *Varje byte-kod-instruktion måste dock mappas till den aktuella hårdvaran*

Kompilering & Länkning

- *Kompilering*
 1. *Analys av källkod -> tokens*
 2. *Parser -> abstrakt träd av konstruktioner*
 3. *Generering av mellankod -> assembler*
 4. *Kodoptimering -> assembler*
 5. *Kodgenererare -> maskinkod*
- *Länkning*
 - *Integrering av maskinkod från olika bibliotek till en exekverbar helhet*

Firmware

- *Ofta skrivet som mikroprogrammering, dvs ett lager under det egentliga maskinspråket*
- *Enkla, fundamentala instruktioner för att implementera själva maskinspråket*
- *Även den mjukvara som är resident i ett system (t.ex. digitalkamera / printer / nätverksenhet)*

Operativsystem

Modeller
(1.7 i boken)

Monolitisk arkitektur

- *Varje del är innesluten i kerneln och kan direkt kommunicera med andra delar*
 - *T.ex: filsystemet kan direkt kalla på minnesallokatorn*
- *Kerneln körs i obegränsat (CPU) tillstånd:*
 - *Alla operationer är tillåtna (och felaktiga är ödesdigra för systemet...)*
- *OS/360, VMS, Linux kan kategoriseras som monolitiska operativsystem*

Arkitektur i lager

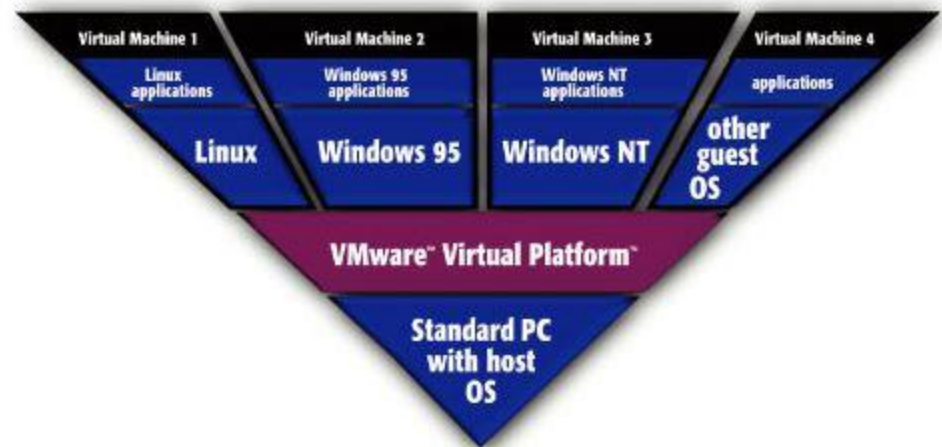
- *Komponenter som utför liknande funktioner läggs i lager*
- *Lager på lägre nivå erbjuder service för lager på högre nivå*
- *Gör systemet mer modulärt – klarare design*
- *Sämre prestanda jämfört med ett monolitiskt system*

Mikrokernelarkitektur

- *Själva kärnan erbjuder mycket begränsad mängd service*
 - *minneshantering*
 - *IPC*
 - *processsynkronisering*
- *Annan service är utanför kärnan*
 - *processhantering*
 - *nätverk*
 - *filsystem*
- *Mycket modulära*
- *Intermodulär kommunikation – minskad prestanda i systemet*

Virtuella maskiner

- *Ett tidsdelande (timesharing system) kan erbjuda en exakt kopia av den existerande hårdvaran*
 - *De olika virtuella maskinerna kan köra olika operativsystem*
 - *Systemanrop fångas först av det egna operativsystemet, som sedan när det i sin tur utför ett kommando, fångas av den virtuella maskinen*
 - *Exempel: VMWare (www.wmware.com),*



Nätbaserade distribuerade system

- *Processer kan accessera resurser som finns på andra oberoende datorer i ett nätverk*
- *Ofta klient/server-modellen*
- *Exempel: Sun NFS, DCOM, ActiveX, CORBA, Web 2.0*

Monolitisk vs. microkernel

- *Monolitisk*

<i>Fördelar</i>	<i>Nackdelar</i>
<ul style="list-style-type: none">• <i>Prestanda</i>	<ul style="list-style-type: none">• <i>Mera utsatt (dvs.</i>• <i>Statisk</i>

- *Microkernel*

<i>Fördelar</i>	<i>Nackdelar</i>
<ul style="list-style-type: none">• <i>Modularitet</i>• <i>Klar design</i>	<ul style="list-style-type: none">• <i>Prestanda</i>

Men...

- *WinNT (ursprungligen microkernel)*
 - *Har modifierats mot monolitisk konstruktion (bl.a. grafikprestanda har krävt detta)*
- *Linux (monolitisk konstruktion)*
 - *Rör sig mot microkernel*
 - *Kerneltrådar (servrar)*
 - *Moduler (modulär kernel, fast detta inte direkt motsvarar microkernelkonstruktion)*