

Számítógép architektúrák

- Számítógépek felépítése
- Digitális adatábrázolás
- Digitális logikai szint
- Mikroarchitektúra szint
- Gépi utasítás szint
- Operációs rendszer szint
- **Assembly nyelvi szint**
- Probléma orientált (magas szintű) nyelvi szint
- Perifériák

Ajánlott irodalom

• <http://it.inf.unideb.hu/~halasz>

• **S. Tanenbaum: *Structured computer organization* (Prentice Hall, 2006) (T). Magyarul: Számítógép-architektúrák 2. átdolgozott, bővített kiadás (Panem 2006).**

• Patterson D.A., Henessy J.L.: *Computer organization & Design*, Morgan Kaufmann Publ. (2 ed.) 1998.

• Rob Williams: *Computer System Architecture (A Networking Approach)*, Addison Wesley, 2001.

• Sima D., Fountain T., Kacsuk, P.: *Korszerű számítógép architektúrák tervezési tér megközelítésben*, Szak Kiadó, 1998.

• **Randall Hyde: *The Art of Assembler Language*, Randall Hyde, 2003.**

• Osborne: *80386/80286 Assembly Language Programming*, Mc Graw-Hill, 1986.

A **digitális számítógép** olyan gép, amely a neki szóló **utasítások** alapján az emberek számára problémákat old meg.

Azt az utasítássorozatot, amely leírja, hogyan oldjunk meg egy feladatot, **programnak** nevezzük.

A legtöbb gépi utasítás ritkán bonyolultabb mint:

- Adj össze két számot!
- Ellenőrizz egy számot, vajon nulla-e!
- Egy adatot másolj a számítógép memóriájában egyik helyről a másikra!

Egy számítógép utasításainak együttese egy olyan nyelvet alkot, amelyen az ember a számítógéppel képes kommunikálni. Az ilyen nyelvet **gépi nyelvnek** nevezzük.

Egyszerűbb gépi nyelv ==>

egyszerűbb elektronika ==>

olcsóbb gép ==>

az ember számára nehézkes

Legyen L_0 a gépi nyelv, és L_1 egy az ember számára kényelmesebb nyelv.

Hogy hajtható végre az L_1 nyelven írt program?

Kellene olyan gép, amelynek gépi nyelve az L_1 nyelv.

Fordítás és értelmezés.

Fordítás: Először az L_1 nyelvű program minden utasítását helyettesítjük az L_0 nyelv utasításainak egy vele ekvivalens sorozatával. Az így nyert program teljes egészében az L_0 utasításaiból áll. Ezután az eredeti L_1 nyelvű program helyett a számítógép ezt az L_0 nyelvű programot hajtja végre.

Értelmezés: Az L_1 nyelvű program következő utasítását elemezzük, és a vele ekvivalens L_0 nyelvű utasítássorozatot azonnal végrehajtatjuk a számítógéppel.

A fordítás és az értelmezés is elvégezhető az L_0 nyelvű számítógéppel.

Olyan, mintha lenne olyan gépünk, amely végre tudja hajtani az L_1 nyelven írt programot: **virtuális gép.**

A gépi és az ember számára kényelmes nyelv között oly nagy az eltérés, hogy annak áthidalásához nyelvek és virtuális számítógépek hierarchiája alakult ki.

Strukturált számítógép-felépítés

n. szint **L_n nyelv,** **M_n virtuális gép**

Az L_n nyelvű programokat vagy az alsóbb szinten futó értelmező hajtja végre, vagy az alsóbb szinten futó fordítóprogram fordítja alsóbb szintre

n-1. szint **L_{n-1} nyelv,** **M_{n-1} virtuális gép**

...

1. szint **L_1 nyelv,** **M_1 virtuális gép**

0. szint **L_0 nyelv,** **M_0 valódi gép**

Végrehajtja az L_0 nyelvű programot.

1.1. ábra

(A.S. Tanenbaum: Számítógép-architektúrák)

Számítógép architektúra (architecture)

A felhasználónak látszó gép: adattípusok, utasítások, szolgáltatások összessége.

Számítógép-felépítés (organization): a közvetlenül nem szükséges részekkel is foglalkozik.

Strukturált felépítés

Számítógép – utasítás rendszer – programozás: gépi nyelv (kód).

Bonyolultabb nyelvek: fordítás vagy értelmezés.

Gépi, nyelvi szintek (1.2. ábra)

5. Probléma orientált nyelvi szint
fordítás (fordító program)
4. Assembly nyelvi szint
fordítás (assembler)
3. Operációs rendszer szint
részben értelmezés (operációs rendszer)
2. Gépi utasítás szint
ha van mikroprogram, akkor értelmezés
1. Mikroarchitektúra szint
hardver
0. Digitális logika szint

Gépi, nyelvi szintek (1.2. ábra)

0: **digitális logika** szintje: kapu (gate), *AND*, *OR*, ...
→ 1 bites, → több bites memória, regiszter

1: **mikroarchitektúra** szintje: mikroutasítások, mikroprogram
- nem minden gépen létezik, de a gépi utasítások végrehajtását gyakran mikroprogram végzi, ekkor ez a szint **értelmezi** a 2. szintet.

- Regiszterek, aritmetikai-logikai egység - **ALU**
- Adatfolyam - **adatút**

2: **gépi utasítás** szintje (tényleges gépi utasítások):
itt dől el a kompatibilitás kérdése.

3: **operációs rendszer** szintje: speciális kiegészítők (memóriakezelés, párhuzamos futtatás, ...).

Általában **értelmezés**. A szint utasításait

– az operációs rendszer

– vagy közvetlenül a 2. szint hajtja végre

Az eddigi szintek programjai hosszú számsorozatok (természetesen ma már szimbolikusan készülnek)

----- Eddig: rendszerprogramozók területe -----

4: **assembly nyelv** szintje, szimbolikus leírás

5: **probléma orientált nyelv** szintje: pascal, C, C++, ... , adatbázis kezelők, ...

Ezek tényleges nyelvek, fordítás

Gépi utasítás szintje

Az utasítások a memóriában vannak tárolva.

„4 címes” utasítás:

cím) **command dest, source1, source2, next**

cím: az utasítást tartalmazó memóriarekesz címe

command (=utasítás): az utasítás kódja

dest (destination=cél): itt képződik az eredmény

source1 (=forrás1): a művelet 1. operandusa

source2 (=forrás2): a művelet 2. operandusa

next(=következő): a következő végrehajtandó utasítás címe.

Ez legtöbbször az utasítás utáni első rekesz címe, ezért általában nem kell megadni, csak akkor, ha más utasítással folytatódik a program (ugró utasítás).

A next implicit operandus.

„3 címes” utasítás:

cím) **add** **dest, source1, source2**

hatására **dest** fölveszi a **source1 + source2** értéket. Ilyenkor természetesen elveszik **dest** régi értéke.

További implicit operandusok:

„2 címes” utasítás: (ez a tipikus)

cím) **add** **op1, op2**

hatására **op1** fölveszi az **op1 + op2** értéket.

„1 címes” utasítás:

cím) **add** **op**

hatására **A** fölveszi az **A + op** értéket, ahol **A** egy kitüntetett regiszter (**accumulator**).

Hardver, szoftver fejlődése

Többszintű gépek kialakulása.

- Kezdetben két szint:
 - digitális logika,
 - utasítások.
- Mikroprogram (hardver bővítése programozással): Wilkes, 1951. Gyorsan elterjedt.
Csúcs: hatvanas, hetvenes évek;
nagyon sok új utasítás (*, / , ..., ciklusszervezés, megszakítások) - később ezek az utasítások hardverrel is megvalósíthatókká váltak, és úgy gyorsabbak lettek.

Folyamatosan változó határok.

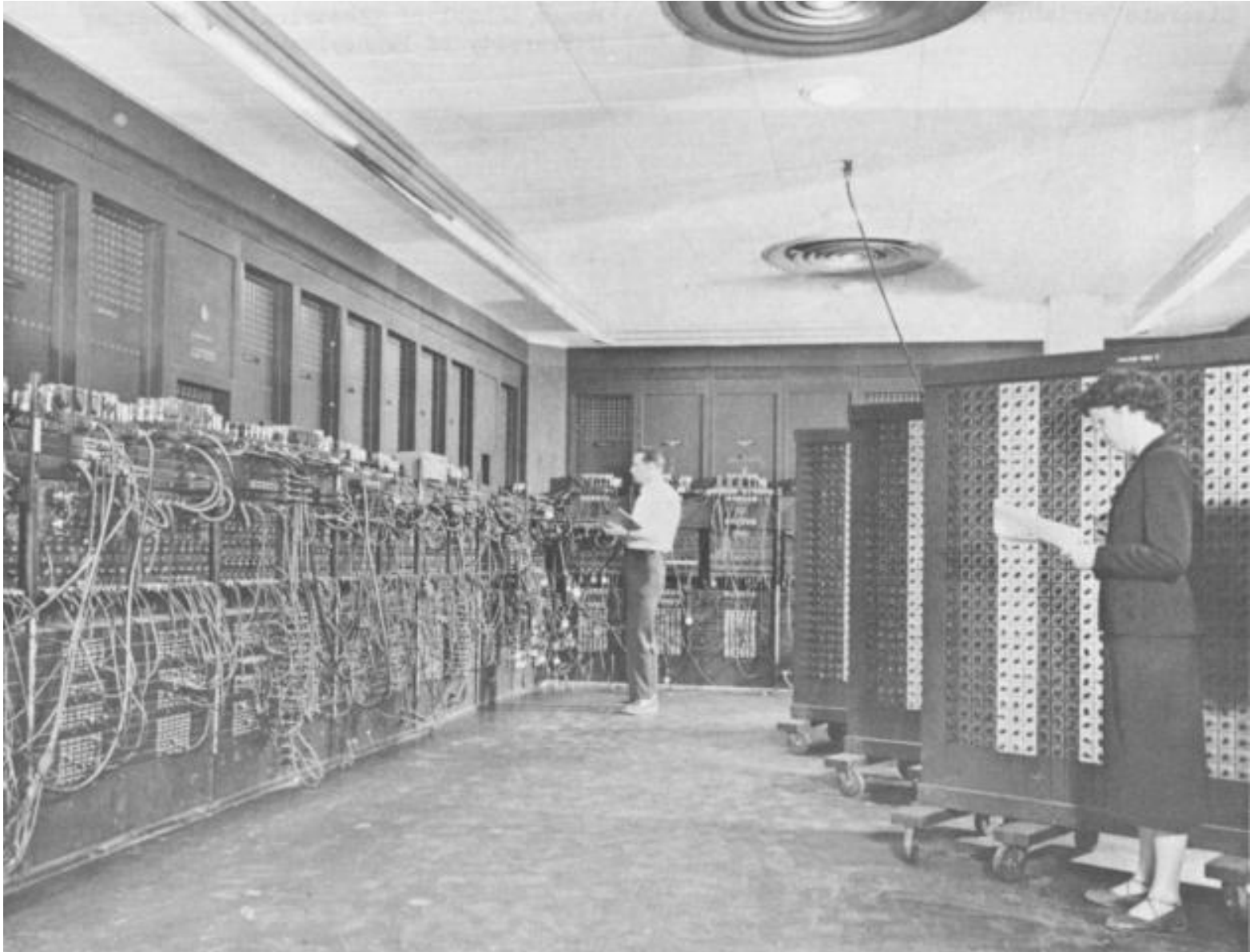
Operációs rendszerek

A hatvanas években készültek először:

- supervisor, rendszerhívások,
- kötegelt (batch) feldolgozás,
- közvetlen telefonos összeköttetés
remote terminálok, időosztás (timesharing).

- 1. generáció: elektroncső (1945-1955).**
- **COLOSSUS (Turing, 1943):** titkosírások megfejtése - 30 évre titkosítva.
 - **ENIAC (Electronic Numerical Integrator and Computer - Mauchley, Eckert, 1943):** 18000 cső, 140 KW, 30 tonna, 20 darab 10 decimális jegyes regiszter. 10 cső egy decimális számjegyhez!
Dugaszolással programozható.
1946-ig nem sikerült befejezni.

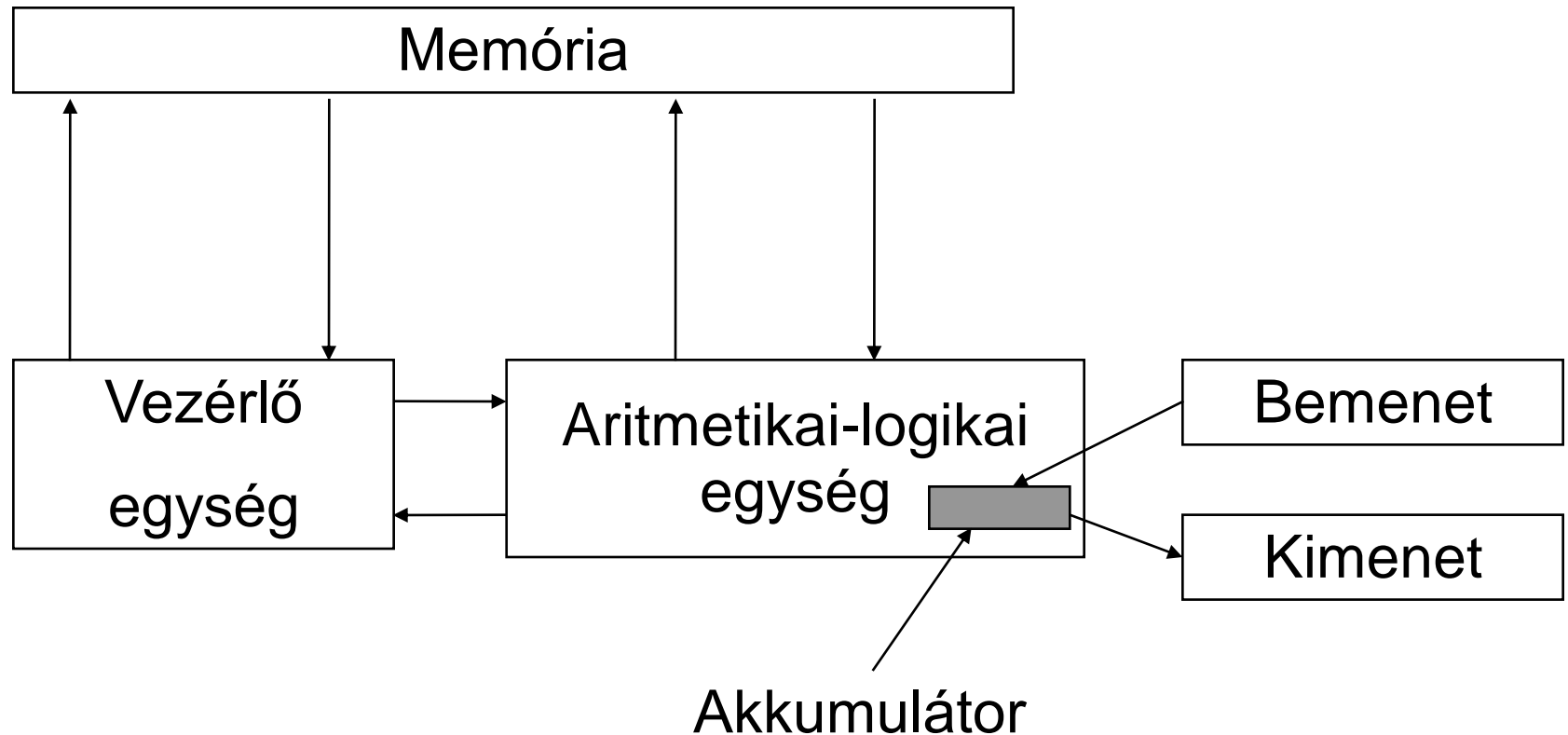
ENIAC



Neumann János



Neumann János megismerte az **ENIAC**-ot, és új gépet tervezett (**IAS**): bináris aritmetika, tárolt program.



1.5. ábra. Az eredeti Neumann-gép

A Neumann elv

A modern számítógépekkel szembeni követelmények:

- Fő funkcionális részek:
 - Vezérlő egység (control unit)
 - Arithmetikai és logikai egység (ALU)
 - Tár (memory), ami címezhető és újraírható
 - Ezek elektronikusak és bináris számrendszert használnak
- Tárolt program elvén működik
- A vezérlő egység határozza meg a működést a tárból kiolvasott adatok alapján

- **EDSAC**: 2-es számrendszer, 4096 szavas memória.
40 bites szavak, egy szóban:
 - előjeles egész, vagy
 - két utasítás.
- **Tárolt program**: 8 bites utasításkód, 12 bites cím.
Akkumulátor.
- Nem volt lebegőpontos aritmetika!
- **IBM 701** (1953) 2K 36 bites memória.
- Legnagyobb 1. generációs: **IBM 709** (1958).

2. generáció: tranzisztor (1955-1965).

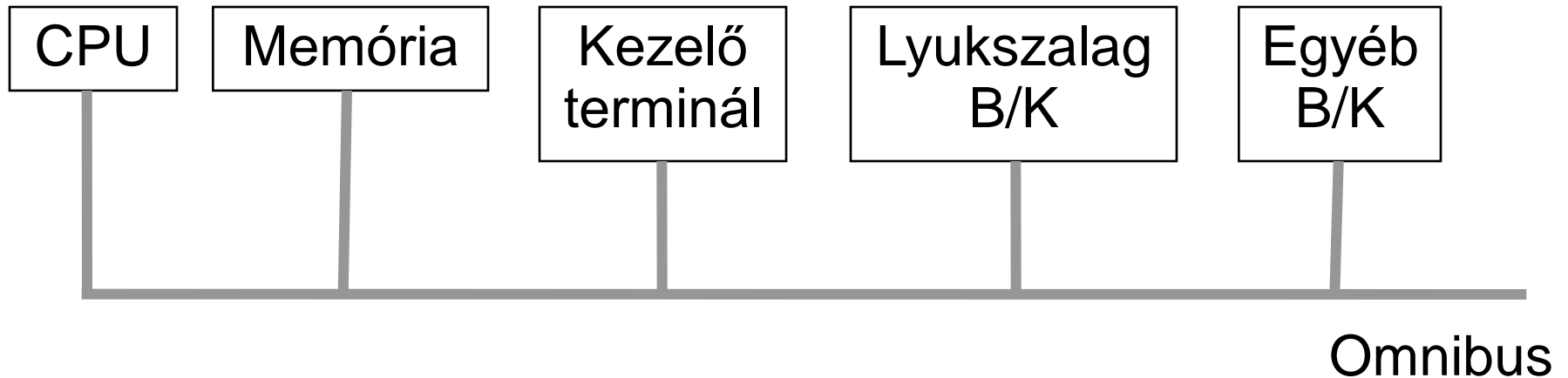
Tranzisztor feltalálása: 1948, Nobel díj: 1956.

- Első jelentős tranzisztoros gép: **DEC PDP-1** (1961).
Ciklusideje fele az **IBM 7090**-nek (az **IBM 709** tranzisztoros változata):
4K 18 bites szó,
5 μ s ciklusidő,
512x512 display!

Ára csak töredéke:

120 000 US\$. Több tucat eladott gép.

- Néhány évvel később: **PDP-8**: omnibusz (általános sín) – 15 000 US\$, 50 000 eladott gép.



1.6. ábra. A PDP-8 „omnibus”

- Magyarországon: KFKI, **TPAi**.
- **IBM 7090, 7094.**

- Első üzlet-orientált gép (1961): **IBM 1401**
byte-szervezésű memória.
- **Burroughs B5000** (1963),
Cél: hatékony ALGOL fordítót lehessen írni!
- Első szuper-gyors gép (1964): **CDC 6600**
Tervező: **Seymour Cray**. Párhuzamos működésre
képes egységek, külön egység az összeadásra,
szorzásra, osztásra.
Párhuzamos utasítás-végrehajtás.

3. generáció: integrált áramkör (1965-1980).

Robert Noyce (1958): szilícium alapú integrált áramkör
IBM problémája: 7094 és 1401 nem kompatibilis.

- (1964): **IBM 360**-as sorozat (**1.7 ábra**).
Később a szocialista országokban **R**-sorozat.

Tulajdonság	Model 30	Model 40	Model 50	Model 65
Relatív teljesítmény	1	3,5	10	21
Ciklus idő (ns)	1000	625	500	250
Maximális memória (KB)	64	256	256	512
Ciklusonként elérhető bájtt	1	2	4	16
Adatcsatornák max. száma	3	3	4	6

Emuláció: a sorozat gépein futtathatók az előző
7094 és **1401** típusok programjai is.
(mikroprogramozás).

Multiprogramozás (**DOS, POWER, OS**).

24 bites címtartomány (16 Mbyte). A nyolcvanas évek közepéig elég, akkor áttérés a 32 bites címre.

- **DEC: PDP-11** (1970): 16 bites

Magyarországon: **TPA70**. Nem kompatibilis a
PDP-11 -gyel.

----- Eddig csak számításigényes problémák -----

4. generáció: VLSI (Very Large Scale Integration) (1980-).
Néhány millió elem egy lapkán (chipen).

- Személyi számítógépek.
Kezdetben zacskóban: nyomtatott áramköri lap,
IC-k, köztük általában **INTEL 8080**,
kábelek, tápegység, hajlékony lemez
- **CP/M** operációsrendszer (Gary Kildall),
- **Commodore, Apple, Spectrum-80.**
- **IBM PC (I-8088 alapú)**
a terveket publikálták - klónok.
- Microsoft: **MS-DOS, OS/2, Windows.**
- ...

Technológiai fejlődés

- **Moore törvény (1965):**
Az egy lapkán elhelyezhető elemek száma másfél évenként duplázódik (**1.8. ábra**).
Azt várják, hogy 2020-ig teljesülni fog.
Minden más területen (lemezek, adatátvitel, ...) hasonló sebességű a fejlődés.

A szoftverek mérete, bonyolultsága is követi ezt:

- **Nathan első törvénye:**
A szoftver gáz: kitölti a rendelkezésre álló teret.

Technológiai fejlődés

A népszerűsítő irodalom kedvenc hasonlata szerint, ha az autóipar az utóbbi hetven évben úgy haladt volna, mint a számítástechnika, egy Rolls-Royce-t 20 \$-ért lehetne kapni, motorja gyufafej nagyságú lenne, sebessége 100 000 km/h és egymillió kilométeren 3 liter benzint fogyasztana

Vámos Tibor

1981

Típus	Ár (US \$)	Felhasználható például
Eldobható	0.5	Üdvözlőlapok, RFID (Radio Frequency IDentification)
Mikrovezérlő	5	Órák, autók, eszközök
Játék	50	Videojátékok
Személyi számítógép	500	Asztali/hordozható
Szerver	5 000	Hálózati szerver
Munkaállomás- gyűjtemény (COW)	50 000- 500 000	Tanszéki mini- szuperszámítógép
Nagyszámítógép	5 000 000	Időjárás előrejelzés...

1.9. ábra. A mai (2005) számítógép típusok választéka

Pentium 4. (1.11. ábra)

Lapka	Dátum	MHz	Tranz.	Mem.	Megjegyzés
I-4004	1971/4	0.108	2300	640	Első egylapkás mikroproc.
I-8008	1972/4	0.108	3500	16 KB	Első 8 bites mikroroc.
I-8080	1974/4	2	6000	64 KB	Első általános célú mikroproc.
I-8086	1978/6	5-10	29000	1 MB	Első 16 bites mikroroc.
I-8088	1979/6	5-8	29000	1 MB	Az IBM PC pocesszora
I-80286	1982/6	8-12	134000	16 MB	Memória védelem
I-80386	1985/10	16-33	275000	4 GB	Első 32 bites mikroproc.
I-80486	1989/4	25-100	1.2M	4 GB	8 KB beépített gyorsítótár
Pentium	1993/5	60-233	3.1M	4 GB	Két csővezeték, MMX
P. Pro	1995/3	150-200	5.5M	4 GB	Két szintű beépített gyorsítótár
P. II	1997/5	233-400	7.5M	4 GB	Pentium Pro + MMX
P. III	1999/2	650-1400	9.5M	4 GB	SSE utasítások 3D grafikához
P. 4	2000/11	1300-3800	42M	4 GB	Hyperthreading + több SSE

UltraSPARC III

Igény: UNIX-ot kisgépekre. Hálózati gépek: Ethernet.

- SUN (Stanford University Network - 1982). Motorola 68020 CPU alapú gépek.
1987-ben félmilliárd \$ a bevételük.
- SPARC (Scalable Processor ARChitecture - 1987). 32 bites, 36 MHz. Több cégnek átadták a gyártási jogot, verseny → gyors fejlődés!
- UltraSPARC I: 64 bites, multimédiás utasítások.
- UltraSPARC II, III: gyorsítás + kevés módosítás.
- UltraSPARC IV, kétprocesszoros UltraSPARC III.

8051

Lapka	Program- memória	Memória típus	RAM	Időzítők	Megszakít- tások
8031	0 KB		128	2	5
8051	4 KB	ROM	128	2	5
8751	8 KB	EPROM	128	2	5
8032	0 KB		256	3	6
8052	8 KB	ROM	256	3	6
8752	8 KB	EPROM	256	3	6

1.14. ábra. Az MCS-51 család tagjai

Beágyazott rendszerekben használatos. Évente 8 milliárd mikrovezérlőt adnak el! Ez a család a legnépszerűbb! Nagyon olcsó (10-15 cent).