

1. Bevezetés és alapfogalmak

1.1 Adat és információ

- **Adat:** egy jelenség rögzített leírása, például szám, szöveg, kép vagy mérési érték.
 - Példa: egy szenzor 22,5 °C hőmérsékletet mér.
- **Információ:** az adatok értelmezésével nyert, döntést segítő ismeret.
 - Példa: a 95 °C-os mérés alapján a gép túlmelegedett.
- **Bit:** két állapotot, általában 0-t és 1-et megkülönböztető információegység.

Claude Shannon 1948-ban megjelent információelmélete matematikai kapcsolatot teremtett az információ és a bizonytalanság között.

1.2 Analóg és digitális jelek

- **Analóg jel:** időben és értékben folytonosan változhat, például egy mikrofon feszültségjele.
- **Digitális jel:** meghatározott, diszkrét értékeket vesz fel. A számítógépek az adatokat binárisan ábrázolják.

A valós mérési jeleket az analóg-digitális átalakító (A/D konverter) teszi feldolgozhatóvá. A digitális ábrázolás előnye a megbízható tárolás, másolás és feldolgozás, de a mintavételezés és a kvantálás miatt közelítést tartalmaz.

Méréskor két jellemző különösen fontos:

- **mintavételi frekvencia:** másodpercenként hány minta készül. Ideális esetben legalább kétszer nagyobbak kell lennie a mérendő jel legnagyobb frekvenciájánál;
- **felbontás:** hány bit ír le egy mintát. Egy 12 bites átalakító $2^{12} = 4096$ különböző szintet tud megkülönböztetni.

Ipari példa: egy rezgésdiagnosztikai rendszerben a túl alacsony mintavételi frekvencia elfedheti a csapágyhibára utaló nagyfrekvenciás rezgéseket. A mintavétel előtt ezért analóg aluláteresztő, úgynevezett anti-aliasing szűrőt is alkalmaznak.

flowchart LR J[Analóg jel] --> AD[A/D átalakító] AD --> D[Digitális adat] D --> CPU[Feldolgozás]

1.3 A tárolt program elve

A Neumann-architektúra legfontosabb jellemzői:

- a programutasítások és az adatok a memóriában találhatók;
- a CPU beolvassa, értelmezi és végrehajtja az utasításokat;
- az adatokat és az utasításokat binárisan ábrázolja;
- a CPU, a memória és a be-/kiviteli egységek együtt alkotják a rendszert.

Az eredeti modell alapvetően egymás után hajtja végre az utasításokat. A modern processzorok ezt többek között futószalagos, többmagos és párhuzamos végrehajtással gyorsítják.

1.4 Turing-gép

A Turing-gép az algoritmus-végrehajtás elméleti modellje. Fő elemei a cellákra osztott szalag, az olvasó-író fej, az állapotok és az átmeneti szabályok. Nem egy konkrét számítógép felépítését írja le, hanem azt segít meghatározni, hogy egy feladat algoritmikusan kiszámítható-e.

Mérnöki példa: egy szenzor 0,01 mm eltérést mér. A vezérlő ezt összehasonlítja a 0,005 mm-es tűréssel, majd szükség esetén leállítja a gépet. A mérési érték adat; a határérték túllépésének felismerése már döntést támogató információ.

2. A számítógép fő hardverelemei

A hardver a számítógép fizikai részegységeinek összessége. Feladata az adatok bevétele, feldolgozása, tárolása és továbbítása.

2.1 Központi feldolgozó egység (CPU)

A CPU hajtja végre a program utasításait. Fontos részei:

- **vezérlőegység:** irányítja az utasítások végrehajtását;
- **aritmetikai-logikai egység (ALU):** számítási és logikai műveleteket végez;
- **regiszterek:** a végrehajtás során használt, nagyon gyors és kis tárolók;
- **gyorsítótárak (cache):** csökkentik a lassabb memória elérésének gyakoriságát.

A teljesítményt nem önmagában az órajel határozza meg. Számít a mikroarchitektúra, a magok száma, a gyorsítótár, a memória-sávszélesség és a futtatott program párhuzamosíthatósága is.

- Az **órajel** azt mutatja meg, hogy a processzor másodpercenként hány működési ciklust végez. Azonos órajelű processzorok teljesítménye eltérhet.
- A **magok száma** akkor jelent előnyt, ha a feladat több részre bontható.
- A **fogyasztási és hőmérsékleti korlátok** tartós terhelésnél csökkenthetik az elérhető órajelet.

Mérnöki választásnál nem mindig a legnagyobb számítási teljesítmény a döntő. Egy gépvezérlőnél a kiszámítható válaszidő, egy mobil mérőműszernél az energiafogyasztás, egy CAD-munkaállomásnál pedig az egy- és többmagos teljesítmény egyaránt fontos lehet.

2.2 Alaplap és összeköttetések

Az alaplap kapcsolja össze a CPU-t, a memóriát, a háttértárakat és a perifériákat. A kommunikációhoz különböző összeköttetések tartoznak:

- **adatvonalak:** adatokat továbbítanak;
- **címzési információk:** kijelölik az elérni kívánt memória- vagy eszközterületet;
- **vezérlőjelek:** például olvasási, írási és megszakítási műveleteket jeleznek.

Modern belső összeköttetés például a **PCI Express**. Az **NVMe** a PCIe-alapú SSD-khez használt kommunikációs protokoll.

Az összeköttetések két eltérő jellemzője a **sávszélesség** és a **késleltetés**. A sávszélesség az időegység alatt átvihető adatmennyiséget, a késleltetés egy művelet megkezdéséhez vagy válaszához szükséges időt jelenti. Nagy felbontású kameránál a sávszélesség, gyors szabályozási körnél gyakran a kis és kiszámítható késleltetés a fontosabb.

2.3 System-on-Chip (SoC)

Az SoC több funkciót integrál egyetlen lapkára, például CPU-magokat, GPU-t, memóriavezérlőt és kommunikációs vezérlőket. Mobiltelefonokban, beágyazott eszközökben és egykártyás számítógépekben gyakori.

A Raspberry Pi tipikus egykártyás számítógép. SoC-t, RAM-ot, USB-, HDMI- és hálózati csatlakozást, valamint szenzorok és aktuátorok illesztésére használható GPIO-t tartalmaz.

flowchart TD
CPU[CPU] <--> MEM[Memória]
CPU <--> IO[Be- és kiviteli egységek]
CPU <--> STORE[Háttértár]

3. Számítógépkategóriák és architektúrák

3.1 Számítógépkategóriák

- **Mikrokontroller:** vezérlési feladatra készült, kis fogyasztású eszköz, beépített memóriával és perifériákkal.
- **Személyi számítógép:** általános célú feladatokra és mérnöki szoftverek futtatására használható.
- **Munkaállomás:** nagy teljesítményű számítógép CAD-, képfeldolgozási vagy szimulációs feladatokra.
- **Szerver:** hálózati szolgáltatásokat és több felhasználó vagy alkalmazás adatfeldolgozását végzi.
- **Szuperszámítógép:** nagyméretű, erősen párhuzamos számításokat futtat, például időjárás-előrejelzést vagy mérnöki szimulációt.

Ezek nem egyszerűen teljesítmény szerinti fokozatok: eltérő célra tervezett rendszerek. Egy szerver például nem feltétlenül gyorsabb minden feladatban egy munkaállomásnál.

3.2 Utasításkészlet-architektúrák

Az utasításkészlet-architektúra (ISA) meghatározza, milyen gépi utasításokat ismer a processzor.

- **x86-64:** PC-kben, munkaállomásokban és szerverekben elterjedt.
- **ARM:** mobil, beágyazott és egyre több asztali, illetve szerverrendszerben használják; gyakori előnye a kedvező energiahatékonyság.
- **RISC-V:** nyílt szabványú ISA, amely oktatási, kutatási és ipari célokra egyaránt használható.

Az ISA nem azonos a konkrét processzor belső felépítésével, azaz a mikroarchitektúrával. A **SoC** pedig integrációs megoldás, nem az x86, az ARM vagy a RISC-V melletti negyedik ISA.

Ipari példa: egy tervezőirodai munkaállomás lehet x86-64 alapú, miközben egy robotba épített vezérlő ARM processzort, egy egyedi érzékelőegység pedig RISC-V mikrokontrollert használ. A választásnál a szükséges szoftverek, a fogyasztás, a valós idejű igények, az ár és a hosszú távú beszerezhetőség együtt számít.

4. Perifériák és interfészek

A perifériák biztosítják a kapcsolatot a számítógép és a környezete között. Az interfész az összekapcsolás fizikai és kommunikációs szabályait jelenti.

4.1 Perifériák feladat szerint

- **Bemeneti eszközök:** billentyűzet, egér, kamera, mikrofon, vonalkódolvasó és szenzor.
- **Kimeneti eszközök:** monitor, projektor, hangszóró, nyomtató, 3D nyomtató és aktuátor.
- **Kétirányú eszközök:** érintőképernyő, hálózati adapter és külső adattároló.

4.2 Interfészek felhasználási terület szerint

- **Általános perifériák:** USB, USB-C.
- **Kijelző és hang:** HDMI, DisplayPort.
- **Nagy sebességű külső kapcsolat:** USB4, Thunderbolt.
- **Hálózat:** Ethernet, Wi-Fi, Bluetooth.
- **Beágyazott és ipari kommunikáció:** GPIO, UART, I²C, SPI, CAN, RS-485 és ipari Ethernet.

Az USB-C a csatlakozó fizikai kialakítását jelöli; az átvitel sebességét és a támogatott funkciókat a használt USB-verzió és üzemmód határozza meg.

Az **interfész** és a **kommunikációs protokoll** nem ugyanaz. Az RS-485 például az elektromos jelátvitelt határozza meg, míg felette Modbus RTU protokoll is használható. Az Ethernet szintén több ipari protokoll, például PROFINET, EtherNet/IP vagy Modbus TCP alapja lehet.

Megoldás	Jellemző használat	Fontos tulajdonság
I ² C, SPI	Áramköri lapon belüli szenzorok	Rövid távolság, kis költség
CAN	Járművek, gépek, elosztott vezérlők	Zavartűrés, prioritásos üzenetek
RS-485	Ipari műszerek, több résztvevős hálózat	Hosszabb kábel, differenciális jelátvitel
Ipari Ethernet	PLC-k, hajtások, robotcellák	Nagyobb sebesség, hálózati integráció

flowchart LR S[Szenzorok és bemenetek] --> C[Számítógép vagy vezérlő] C --> A[Kijelzők és aktuátorok] C <--> N[Hálózat és adattárolók]

5. Memória és adattárolás

5.1 Memórahierarchia

A tárolók között sebesség-kapacitás-költség kompromisszum van. A CPU-hoz közelebbi szintek általában gyorsabbak és kisebbek.

- **Regiszterek:** a CPU közvetlen munkaterülete.
- **Cache (L1, L2, L3):** a gyakran használt adatok gyorsítótára.
- **RAM:** a futó programok és adataik átmeneti tárolója; kikapcsoláskor tartalma elvész.
- **SSD vagy HDD:** nagy kapacitású, tartós adattárolás.

flowchart LR R[Regiszter] --> C[Cache] --> RAM[RAM] --> SSD[SSD] --> HDD[HDD]

Balról jobbra általában nő a kapacitás és a hozzáférési idő, miközben csökken az egységnyi tárolás költsége. A következő értékek tájékoztató nagyságrendek; processzorgenerációtól és mérési módszertől jelentősen függhetnek.

Szint	Tipikus kapacitás	Közelítő hozzáférési idő	Szerep
Regiszter	néhány száz bájt/mag nagyságrend	közvetlenül az utasítás használja	az éppen feldolgozott értékek
L1 cache	32–128 KiB/mag	körülbelül 3–5 CPU-ciklus, gyakran 1–2 ns	a leggyakrabban használt utasítások és adatok
L2 cache	256 KiB–2 MiB/mag	körülbelül 10–20 ciklus, néhány ns	nagyobb, de lassabb magközeli gyorsítótár
L3 cache	néhány MiB–több tíz MiB, gyakran megosztott	körülbelül 30–80 ciklus, 10–30 ns	a magok által közösen használt adatok
RAM	néhány GiB–több száz GiB	körülbelül 50–100 ns	futó programok
NVMe SSD	több száz GiB–több TiB	jellemzően több tíz–száz µs	tartós, nagy sebességű tárolás
HDD	több TiB	jellemzően 5–15 ms	olcsó, nagy kapacitású tárolás

Egy 3 GHz-es processzor egy ciklusa körülbelül 0,33 ns. Így a RAM-ból érkező adatra várás alatt a CPU akár több száz ciklust is elveszíthet. A cache ezt a különbséget próbálja áthidalni.

- **Cache-találat (hit):** a keresett adat megtalálható a cache-ben.
- **Cache-hiba (miss):** az adatot egy lassabb szintről kell betölteni.
- **Időbeli lokalitás:** a nemrég használt adatra valószínűleg hamarosan ismét szükség lesz.
- **Térbeli lokalitás:** egy adat használata után gyakran a mellette lévő adatok következnek.

Ipari példa: képfeldolgozásnál egy kamera képének sorfolytonos feldolgozása jobban kihasználhatja a cache-t, mint a képpontok véletlenszerű elérése. Nagyméretű FEM-modellnél pedig előfordulhat, hogy nem a CPU számítási képessége, hanem a RAM sávszélessége korlátozza a sebességet.

5.2 RAM és háttértárak

- **DRAM:** a főmemória leggyakoribb technológiája; rendszeres frissítést igényel.
- **SRAM:** gyorsabb és drágább; elsősorban gyorsítótárakban használják.
- **DDR4, DDR5:** a DRAM szabványainak különböző generációi.

- **HDD:** mágneses, nagy kapacitású és olcsó, de mechanikus működése miatt lassabb.
- **SSD:** flashmemóriát használ, nincs mozgó alkatrésze.
- **NVMe SSD:** PCIe-kapcsolaton kommunikáló, kis késleltetésű SSD.
- **Pendrive és memóriakártya:** hordozható flashalapú tárolók.

5.3 Virtuális memória

Az operációs rendszer virtuális címtartományt biztosít a folyamatoknak, és szükség esetén a háttértárat is bevonhatja a memóriakezelésbe. Ez nem egyszerűen „RAM-bővítés”: többek között elkülöníti a folyamatokat és lehetővé teszi, hogy a programok a fizikai RAM-nál nagyobb címtartományt használjanak. A háttértárra történő lapozás azonban nagyságrendekkel lassabb a RAM elérésénél.

Mérnöki szimulációnál a RAM kapacitása korlátozhatja a modell méretét, míg az SSD sebessége a nagy adathalmazok beolvasását és mentését befolyásolja.

Ipari adatgyűjtő rendszernél nemcsak a tároló kapacitását kell méretezni, hanem a folyamatos írási sebességet, az adatmegőrzési időt, a meghibásodás hatását és a mentési eljárást is.

6. Párhuzamos feldolgozás és gyorsítók

A teljesítmény növelésének fontos eszköze, hogy több művelet részben vagy teljesen egy időben fusson.

6.1 Párhuzamosság a CPU-ban

- **Futószalag (pipeline):** az utasítás-végrehajtás lépései átfedésben működnek.
- **SIMD:** egy utasítás több adatelemen hajt végre azonos műveletet.
- **Többmagos feldolgozás:** több processzormag külön végrehajtási szálakon dolgozhat.
- **MIMD:** különböző utasításfolyamok különböző adatokon futnak; többmagos és többprocesszoros rendszerekre jellemző.

A gyorsulás mértékét korlátozza, hogy a feladat mekkora része párhuzamosítható, továbbá az adatmozgatás és a szálak összehangolásának költsége.

6.2 Speciális gyorsítók

- **GPU:** sok hasonló művelet párhuzamos végrehajtására alkalmas; grafika, gépi tanulás, képfeldolgozás, FEM és CFD gyorsítására használható.
- **FPGA:** újrakonfigurálható logikai áramkör; kis késleltetésű, testreszabott és valós idejű feldolgozásra alkalmas.
- **TPU és NPU:** neurális hálók műveleteire optimalizált gyorsítók.

flowchart TD
FEL[Számítási feladat] --> CPU[CPU: általános vezérlés]
CPU --> GPU[GPU: tömeges]

párhuzamosság] FEL --> FPGA[FPGA: testreszabott adatút] FEL --> NPU[NPU/TPU: neurális hálók]

Nem minden program gyorsul GPU-n vagy más gyorsítón: a megfelelő párhuzamosság mellett az adatátvitel költségét is figyelembe kell venni.

Mérnöki példák:

- **CPU:** vezérlési logika, operációs rendszer, változatos számítási lépések;
- **GPU:** képfeldolgozás, neurális háló, nagyszámú hasonló FEM- vagy CFD-művelet;
- **FPGA:** nagy sebességű kameraadatok előfeldolgozása, precíz időzítésű mérés és motorvezérlés;
- **NPU:** helyi hibafelismerés intelligens kamerában vagy edge eszközben.

Egy hibrid rendszerben ezek együtt is dolgozhatnak: a CPU szervezi a folyamatot, a GPU vagy az FPGA elvégzi a jól párhuzamosítható részt, a CPU pedig feldolgozza az eredményt.

7. Ipari és mérnöki számítógépes rendszerek

7.1 Beágyazott rendszerek

A beágyazott rendszer egy nagyobb berendezés meghatározott funkcióját végző számítógép. Gyakran korlátozott memóriával és fogyasztással, valamint valós idejű követelményekkel működik.

- **Mikrokontrollerek**, például az ESP32: szenzoradatok feldolgozása, kommunikáció és aktuátorvezérlés.
- **Egykártyás számítógépek**, például a Raspberry Pi: összetettebb operációs rendszert és alkalmazásokat is futtathatnak.

Az Arduino elsősorban fejlesztőplatform és ökoszisztéma; nem egyetlen mikrokontroller-típus.

7.2 PLC és ipari vezérlés

A **PLC** ipari gépek és folyamatok megbízható vezérlésére készült számítógép. Ipari I/O-modulokhoz kapcsolódik, ciklikusan hajtja végre a vezérlőprogramot, és folyamatos üzemre tervezték. Gyártósorok, robotcellák és szállítórendszerek vezérlésében gyakori.

A PLC tipikus működési ciklusa: bemenetek beolvasása → vezérlőprogram végrehajtása → kimenetek frissítése → diagnosztika és kommunikáció. A ciklusidőnek elég rövidnek és kiszámíthatónak kell lennie a vezérelt folyamathoz. Egy gyors hajtásszabályozás és egy tartály lassú szintszabályozása ezért eltérő hardvert és időzítést igényelhet.

flowchart LR
S[Szenzorok] --> PLC[PLC vagy beágyazott vezérlő]
PLC --> A[Aktuátorok]
PLC <--> HMI[HMI]
PLC <--> NET[Ipari hálózat]

7.3 IoT és edge computing

Az **IoT** hálózatba kapcsolt fizikai eszközökből áll, amelyek adatot gyűjtenek, továbbítanak vagy

beavatkozást végeznek. **Edge computing** esetén az adatok egy részét a szenzor vagy a gép közelében dolgozzák fel. Ez csökkentheti a késleltetést és a hálózati terhelést, valamint internetkapcsolat nélkül is lehetővé tehet bizonyos helyi funkciókat.

Ipari példák: állapotfelügyelet, prediktív karbantartás, energiafigyelés és minőség-ellenőrzés. Egy rezgésérzékelő edge eszköze például helyben kiszámíthatja a jellemző frekvenciákat, és a teljes nagyságsebességgű jelfolyam helyett csak a diagnosztikai eredményt küldi tovább.

7.4 Nagy teljesítményű számítástechnika (HPC)

A HPC sok processzort, gyorsítót és számítási csomópontot kapcsol össze nagy számítási igényű feladatokhoz. Tipikus alkalmazás a véges elemes analízis (FEM), az áramlási szimuláció (CFD), az időjárás-modellezés és a molekuláris dinamika.

HPC-rendszerénél a processzorok száma mellett a memória mérete és sávszélessége, a csomópontok közötti hálózat, a gyorsítók, valamint az adattárolás sebessége is meghatározó. Egy túl nagy FEM-modell például már nem fér el egy munkaállomás memóriájában, ezért több csomópont között kell felosztani.

A megfelelő rendszer kiválasztását mindig a feladat határozza meg: egyszerű vezérléshez mikrokontroller vagy PLC, helyi adatfeldolgozáshoz edge eszköz, CAD-hez munkaállomás, nagyméretű szimulációhoz pedig többmagos, GPU-s vagy HPC-rendszer lehet célszerű.

From:
<https://edu.iit.uni-miskolc.hu/> - Institute of Information Science - University of Miskolc

Permanent link:
https://edu.iit.uni-miskolc.hu/tanszek:oktatas:szamitastechnika:hardver_alapismeretek?rev=1788898286

Last update: 2026/09/08 20:11

