

Október 17, 18, 19, 20, 21-én teszt
az Irinyi 227-es teremben

a MOODLE vizsgáztató programmal
az október 17-e előtt
elhangzott előadások anyagából.

A vizsgáztató program az október 10-ével kezdődő
héten kipróbálható, gyakorolható lesz.

További információt a honlapomon, az előadás
mellékletek között fogok adni.

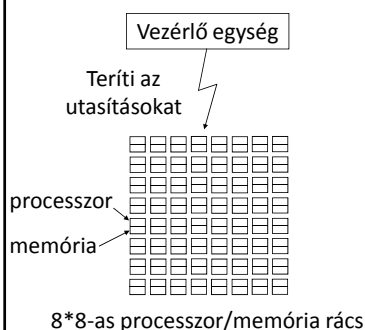
Máté: Architektúrák

4. előadás

1

Processzor szintű párhuzamosítás

• Tömb (array) processzor (2.7. ábra)



Sok azonos processzor
(ILLIAC IV: $(4 \times) 8 \times 8$,
1972.), mindnek saját
memóriája. Vezérlő
processzor adja ki a
feladatot.
Mindegyik processzor
ugyanazt csinálja, de a
saját adatain.

Már nem divatos (?)
(drága, nehéz
kihasználni).

Grafikus processzorok

Máté: Architektúrák

4. előadás

2

• Vektor processzorok

Vektor regisztereket használnak.

A vektor regiszter több hagyományos regiszterből
áll. Gyors szállítószalag gondoskodik a regiszterek
feltöltéséről, szintén gyors szállítószalag továbbítja a
regiszterek tartamát az aritmetikai egységbe, pl. a
vektor regiszterek összeadásához. Az eredmények
szintén vektor regiszterbe kerülnek.

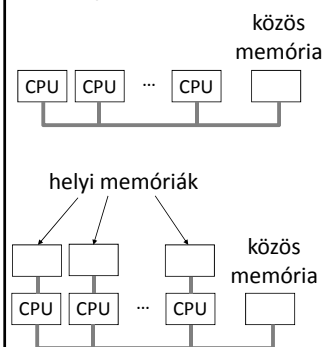
Jól kombinálhatók hagyományos processzorokkal.

Máté: Architektúrák

4. előadás

3

• Multiprocesszorok



A közös memória
megkönnyíti a feladat
megosztását.

- Csak közös memória.
Nagyon terheli a memória
sín.

- Lokális memória is van.

Sok (>64) processzoros
rendszer nehéz építeni a
közös memória miatt.

2.8. ábra

Máté: Architektúrák

4. előadás

4

• Multiszámítógépek: Nincs közös memória:

A CPU-k üzenetekkel tartják egymással a
kapcsolatot.

Néhány μ s üzenet idő.

2-3 dimenziós hálók, fák, gyűrűk.

Közel 10 000-es rendszer is van.

Máté: Architektúrák

4. előadás

5

Adattípusok

Alapkérdés: mit támogat a hardver (milyen utasítások
vannak)? Ami nincs (pl. dupla pontosságú egész
aritmetika), azt szoftveresen kell megcsinálni.

Numerikus típusok:

- előjel nélküli és előjeles egész számok
(8, 16, 32, 64 bites).
- lebegőpontos számok (32, 64, néha 128 bites),
- binárisan kódolt decimális számok: decimális
aritmetika (COBOL $\rightarrow$ Y2K = 2000. év probléma).

Máté: Architektúrák

4. előadás

6

Az egyes gépek által támogatott numerikus típusok

P: Pentium 4, U: UltraSPARC III, I: I-8051

típus	1 bit	8 bit	16 bit	32 bit	64 bit	128 bit
bit	I					
előjeles egész		P U I	P U	P U	U	
előjel nélküli egész		P U	P U	P U	U	
BCD		P				
lebegőpontos				P U	P U	U

5.7-9. ábra

Máté: Architektúrák

4. előadás

7

Karakterkódolás

ASCII (American Standard Code for Information Interchanges), 7 bites: vezérlőkérekek, az angol abc kis és nagy betűi, szimbólumok, **2.43. ábra**

Latin-1 kód: 8 bites.

IS 8859: kódlap, **IS 8859-2:** magyar betűk is.

UNICODE (IS 10646), 16 bites: kódpozíciók (code point). Általában egy nyelv jelei egymás után vannak – a rendezés könnyű.

- Kínai, japán, koreai: fonetikus szimbólumok, Han ideogramok (20992 jel, nincsenek szótár szerint rendezve). ... Japán íráshoz kevés (> 50000 kanji jel van).
- Új jelek? Braille nincs benne.

Máté: Architektúrák

4. előadás

8

További nem numerikus típusok

Logikai érték (boolean): igaz, hamis.

Leggyakrabban egy bájtban (szóban) ábrázolják.

Bit térkép.

Mutató (pointer): memória cím.

Bit: kapcsolók, lámpák beállítására, lekérdezésére beágyazott rendszerekben.

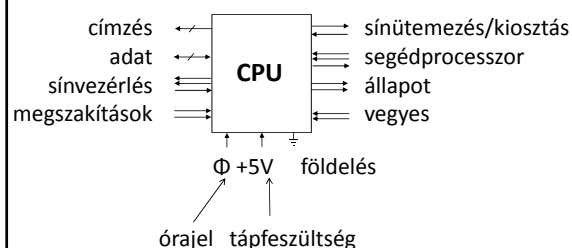
Máté: Architektúrák

4. előadás

9

CPU (Central Processing Unit)

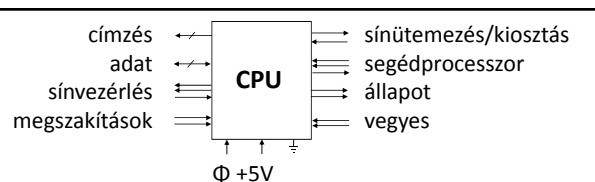
Általában egyetlen lapkán van. **Lábakon** keresztül tartja a kapcsolatot a többi egységgel (**3.34. ábra**).



Máté: Architektúrák

4. előadás

10

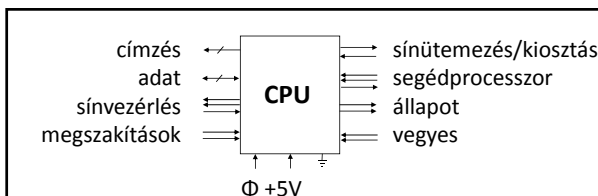


Lábak (**pins**) három típusa: **cím**, **adat**, **vezérlés**. Ezek párhuzamos vezetékeken, az un. **sínen** keresztül kapcsolódnak a **memória**, az **I/O** egységek hasonló lábaihoz.

Máté: Architektúrák

4. előadás

11



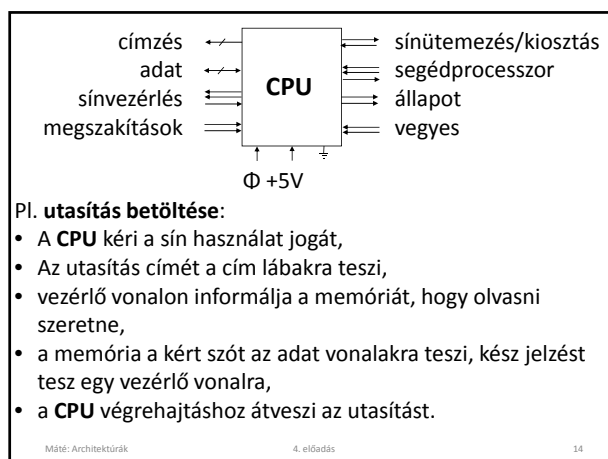
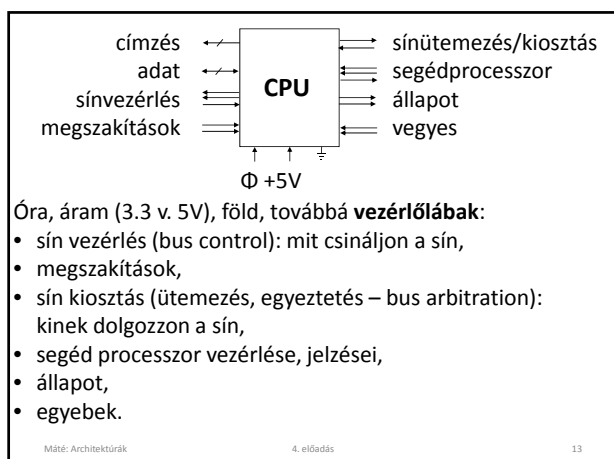
Lényeges a cím- és adatlábak száma (**3.34. ábra**):

- Ha m címláb van, akkor 2^m memóriarekesz érhető el (tipikus $m = 16, 20, 32, 64$).
- Ha n adatláb van, akkor egyszerre n bit olvasható illetve írható (tipikus $n = 8, 16, 32, 36, 64$).

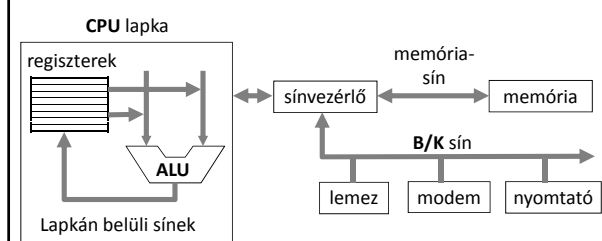
Máté: Architektúrák

4. előadás

12



Sín (bus): Korai személyi számítógépeknél egyetlen (külső) rendszersín, manapság legalább kettő van: egy belső és egy külső (I/O), **3.35. ábra**.



Sínprotokoll: a sín működésének + a csatlakozások mechanikai, elektronikus definíciója

Mesterek (masters): aktív (kezdeményező) berendezések (CPU, lemez vezérlő).

Szolgák (slaves): passzív (végrehajtó) berendezések (lemez vezérlő, CPU), **3.35. ábra**.

Ez a szereposztás tranzakciónként eltérő lehet.

Mester	Szolga	példa
CPU	Segéd proc.	CPU felkínálja az utasítást
Segéd proc.	CPU	Segéd proc. kéri az operandusokat

A memória sohasem lehet mester!

Máté: Architektúrák

4. előadás

16

A sínhez kapcsolódó lapkák lényegében erősítők.
 Mester – **sín vezérlő (bus driver)** – sín.
 Sín – **sín vevő (bus receiver)** – szolga.
 Mester–szolgánál: **sín adó-vevő (bus transceiver)**.

A csatlakozás gyakran **tri-state device** vagy **open collector** – **wired-OR** segítségével történik.

Sávszélesség: (továbbítható bitek száma) / sec.

Sávszélesség növelése:

Gyorsítás: probléma a sín aszimmetria (skew), kompatibilitás.

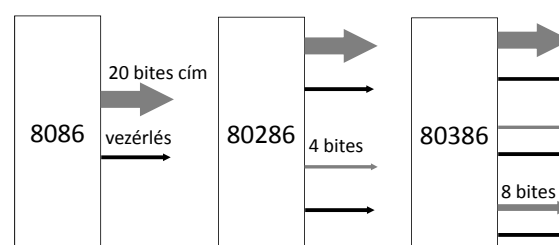
Sínszélesség: több vezeték → drágább, kompatibilitás.

Máté: Architektúrák

4. előadás

17

Sínszélesség (pl. IBM PC: 3.37., 3.51. ábra).



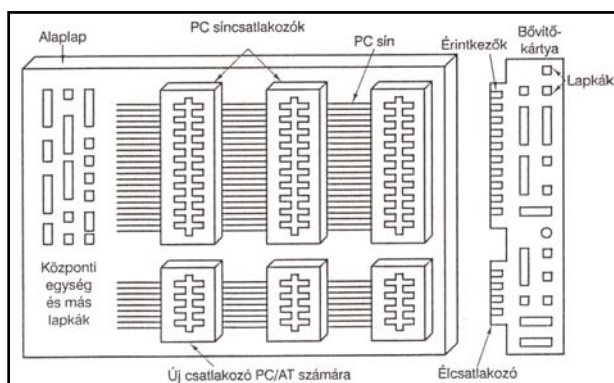
3.37. ábra. A cím szélességének növekedése az elmúlt időszakban

Máté: Architektúrák

Csak az elv kell

4. előadás

18



3.51. ábra. A PC/AT sín két komponense, az eredeti PC és az új rész

Máté: Architektúrák

Csak az elv kell

4. előadás

19

Alaplap (motherboard, parentboard, 3.51. ábra)

Rajta van a **CPU**, sín(ek), ezen illesztő helyek (slots) a memória és a **beviteli/kiviteli (Input/Output – I/O)** eszközök számára (3.51., 2.28. ábra).

I/O eszköz: maga az eszköz + vezérlő (controller) külön kártyán vagy az alaplapon (2.29. ábra).

Gyorsabb **CPU** gyorsabb sínt igényel!

Kívánság: a gép cseréjénél megmaradhasson a régi perifériák egy része: az új gépben is kell a régi sín!

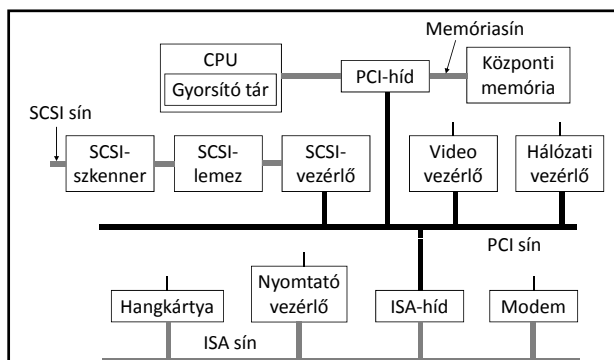
Sínek szabványosítása.

Egy gépen belül több sín is használható: 2.30. ábra.

Máté: Architektúrák

4. előadás

20



2.30. ábra. Egy tipikus modern PC PCI, SCSI és ISA sínnel

Máté: Architektúrák

Nem kell

4. előadás

21

Sokszorozott (multiplexed) **sín:** pl. először a cím van a sínen, majd az adat (ugyanazonokon a vezetékeken). Ilyenkor a sín szélessége lényegesen csökken (olcsóbb, kevesebb láb szükséges a sínhez való csatlakozáshoz), csökken a sáv szélesség is, de nem olyan mértékben.

Általában bonyolultabb a sín protokoll.

Máté: Architektúrák

4. előadás

22

Sínek időzítése

Szinkron sín: 5 – 100 MHz-es órajel van a sín egy vezetéken. Minden síntevékenység az órajelhez van igazítva.

Síntevékenységek: cím megadása, vezérlőjelek (**MREQ#**, **RD#**, **WAIT#**), adat megérkezése, ... (3.38. ábra)

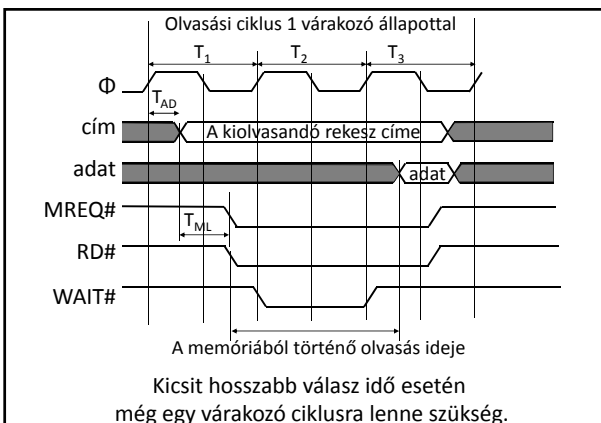
Jelölés	Tevékenység	min	max	idő
T_{AD}	Cím megérkezési ideje a sínre		11	ns
T_{ML}	Cím a sínen van MREQ# előtt	6		ns
...	...	...	...	...

Máté: Architektúrák

Csak az elv kell

4. előadás

23



Máté: Architektúrák

4. előadás

24

Minden sínművelet a ciklusidő (sín ciklus) egész számú többszörösége tart:
pl. 2.1 ciklusidő helyett 3 ciklusidő kell.

A leglassabb eszközhöz kell a sín sebességét igazítani, a gyors eszköz is lassan fog működni.

Máté: Architektúrák

4. előadás

25

Aszinkron sín:

Minden eseményt egy előző esemény okoz!

Nincs órajel, **WAIT**, van viszont:

MSYN# (kérés - Master SYNchronization),

SSYN# (kész - Slave SYNchronization).

Ugyanazon a sínen gyors és lassú mester - szolga pár is lehet.

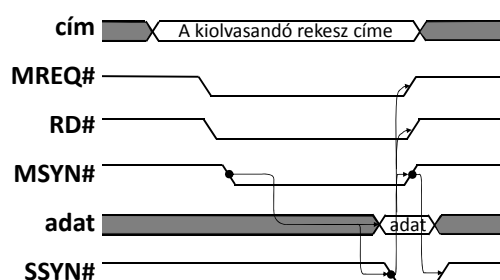
Máté: Architektúrák

4. előadás

26

Aszinkron sín működése (3.39. ábra)

Akkor indulhat újabb tranzakció, ha **SSYN#** negált.



Ugyanazon a sínen gyors és lassú mester - szolga pár is lehet.

Máté: Architektúrák

4. előadás

27

Teljes kézfogás (full handshake):

Akkor indulhat, ha **SSYN#** negált!

- Mester: kívánságok beállítása, majd **MSYN#**, vár,
- Szolga: látja **MSYN#**-t: dolgozik, majd **SSYN#**, vár,
- Mester: látja **SSYN#** -t (a szolga kész), dolgozik, ha kell, majd negálja **MSYN#** -t,
- Szolga: látja **MSYN#** negálását, negálja **SSYN#** -t.

Máté: Architektúrák

4. előadás

28

Sínütemezés (kiosztás)

Ha egyszerre többen is igénylik a sít (CPU, I/O vezérlő), akkor a **sínütemező** (bus arbiter) dönt.

Általában I/O elsőbbséget kap (cikluslopás).

Máté: Architektúrák

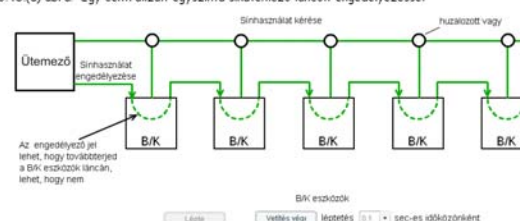
4. előadás

29

Sínütemezés (kiosztás – bus arbitration)

- **Centralizált (3.40. (a) ábra):** (margaréta) láncolás (daisy chaining), egy vagy többszintű lehet.

3.40.(a) ábra. Egy centralizált egyszintű sínütemező láncolt engedélyezéssel



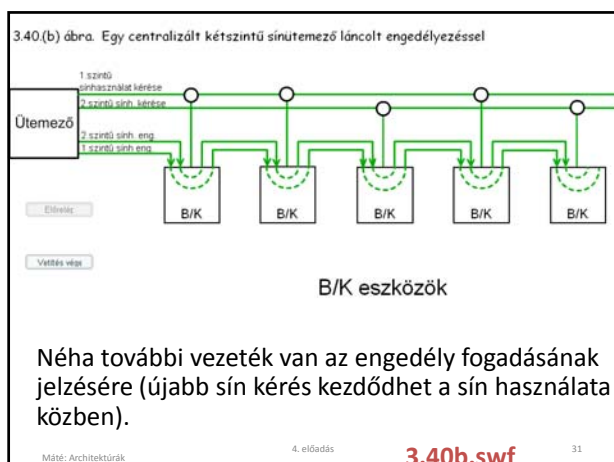
Ha van **kérés** és a **sín szabad: sín foglalási engedély**.

Máté: Architektúrák

4. előadás

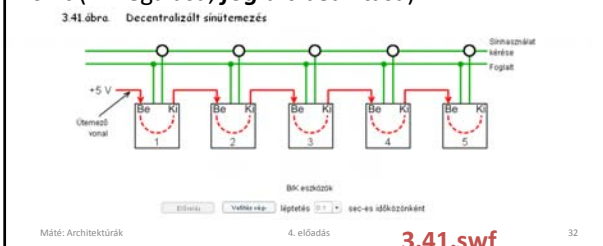
3.40a.swf

30

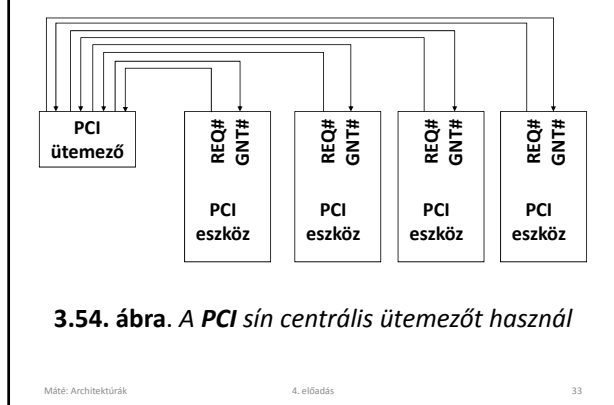


• decentralizált

- pl. 16 prioritású: 16 eszközhez 16 kérés vonal, minden eszköz minden kérés vonalat figyel, tudja, hogy a saját kérése volt-e a legmagasabb prioritású.
- **3.41. ábra:** ha nem **foglalt** és **be**, akkor kérheti a sít (**ki** negálása, **foglalt** beállítása).



PCI sín ütemezés (kiosztás): request, grant.



Sín műveletek

Az eddigiek **közönséges sín műveletek** voltak.

Blokkos átvitel (3.42. ábra): A kezdő memória címen kívül az adatsínre kell tenni a mozgatandó adatok számát. Esetleges várakozó ciklusok után ciklusonként egy adat mozgatása történik.

Megszakítás kezelés: később tárgyaljuk részletesen.

Több processzoros rendszerekben:

olvasás – módosítás – visszairás ciklus: szemafor.

Máté: Architektúrák 4. előadás 34

Példák sínekre

Az első **IBM PC (3.37. ábra)** 62 vonalas (vezeték, line), 20 címnek, 8 adatnak + **DMA**, megszakítás ...

PC/AT szinkron sín (3.51. ábra): további 36 vezeték (címnek összesen 24, adatnak 16, ...).

Microchannel (IBM OS/2 gépekhez), szabadalmak ISA (Industry Standard Architecture) lényegében 8.33 MHz-es PC/AT sín (sávszélesség: 16.7 MB/s).

EISA (Extended ISA) 32 bitesre bővített ISA (sávszélesség: 33.3 MB/s).

Színes TV-hez 135 MB/s sávszélesség kellene (1024*768 pixel, 3 bájtt*2, 30 kép/sec).
lemez → memória → képernyő

Máté: Architektúrák 4. előadás 35

PCI (Peripheral Component Interconnect): 32 bites adat átvitel (33,3 MHz, sávszélesség: 133 MB/s) szabadon felhasználható licenz.
Multiplexelt cím- és adatkivezetések.

Új változatai: 64 bites adat, 66 MHz, 528 MB/s.

Problémák:

- a memóriához lassú,
- nem kompatibilis az ISA bővítőkártyákkal.

Megoldás (3.52. vagy 2.30. ábra): több sín
Belső sín, PCI híd, PCI sín, ISA híd, ISA sín.

Máté: Architektúrák 4. előadás 36

Általános soros sín (USB)**Universal Serial Bus**

Igény: bármikor könnyen, azonos módon lehessen sokféle perifériát kapcsolni a géphez, akár a gép működése közben, hardver ismeretek nélkül:

- ne kelljen kikapcsolni a gépet,
- ne kelljen szétszedni a gépet,
- ne kelljen újra boot-olni,
- ne kelljen áramellátásról gondoskodni,
- ...

Plug 'n Play (csatlakoztasd és működik) perifériák.

Máté: Architektúrák

4. előadás

37

USB (Universal Serial Bus - általános soros sín):

Négy vezeték: adatok (2), tápfeszültség (1), föld (1).

USB 1.0 **1,5 Mbps** (billentyűzet, egér,...)

USB 1.1 **12 Mbps** (nyomtató, fényképezőgép,...)

USB 2.0 **480 Mbps** (DVD lejátszó,...)

A központi elosztó (**root hub**)

1 ms-onként üzenetekkel (**frame**, **3.59. ábra**) kommunikál az eszközökkel.

A frissen csatlakoztatott eszköz címe 0.

Ha a központi elosztó tudja fogadni az eszközt, akkor egyedi címet (1-127) ad neki (**konfigurálja**).

Máté: Architektúrák

4. előadás

38

Frame – keret

- **Control** – vezérlő:
 - Eszköz konfigurálás,
 - Parancs,
 - Állapot lekérdezés.
- **Isochronous** – izosinkron: valós idejű eszközök használják, pl. telefon. Hiba esetén nem kell ismételni az üzenetet.
- **Bulk** – csoportos: nagy tömegű adat átvitelére szolgál.
- **Interrupt** – megszakítás: Az **USB** nem támogatja a megszakítást, helyette pl. 50 ms-enként lekérdezhető az eszköz állapota.

Máté: Architektúrák

4. előadás

39

Frame – keret

Egy vagy több csomagból áll.

Az egyes csomagok haladhatnak a központból az eszközök felé vagy fordítva. A haladási irány egy kereten belül is változhat.

Az első csomag mindig **SOF**:

Start Of Frame – keret kezdet, szinkronizálja az eszközöket.

Máté: Architektúrák

4. előadás

40

A csomag lehet

- **Token** – parancs (központ küldi az eszköznek):
 - **SOF**.
 - **IN** – be: adatokat kér az eszköztől.
Az **IN** parancsban meg lehet adni, melyik eszköztől milyen adatokra van szükség.
 - **OUT** – ki: adatok fogadására kéri az eszközt.
 - **SETUP** – beállítás: konfigurálja az eszközt.

Máté: Architektúrák

4. előadás

41

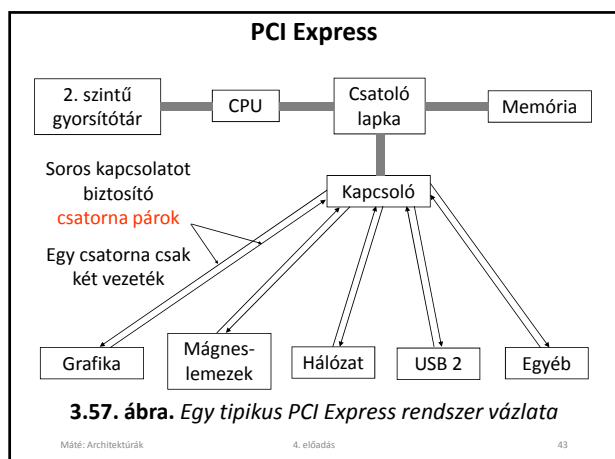
A csomag lehet (folytatás)

- **Data** – adat: 64 bájt információ mozgatása akármelyik irányban. A **Data** csomag részei:
 - **SYN**: 8 bit szinkronizáció,
 - **PID**: a csomag típusa (8 bites),
 - **PAYLOAD**: hasznos adat,
 - **CRC**: Cyclic Redundancy Code – ciklikus redundancia kód (16 bit az adatátvitel helyességének ellenőrzésére).
- **Handshake** – kézfogás:
 - **ACK**: az előző adatcsomagot hibátlanul vettem,
 - **NAK**: **CRC** hibát észleltem,
 - **STALL**: kérem, várjon, el vagyok foglalta.
- **Special** – speciális.

Máté: Architektúrák

4. előadás

42

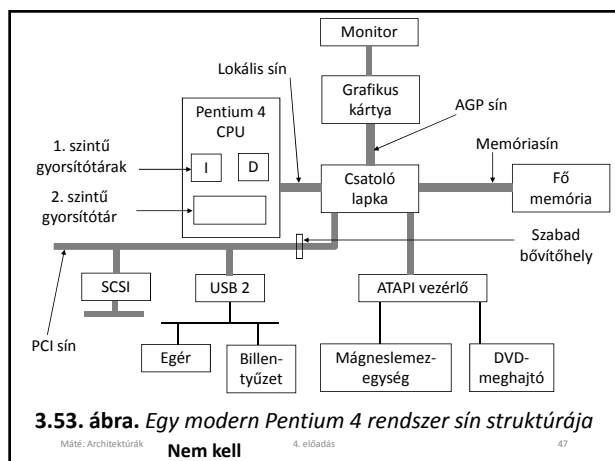
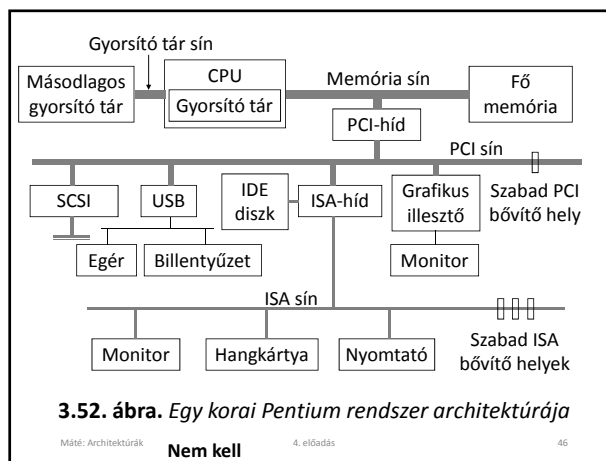
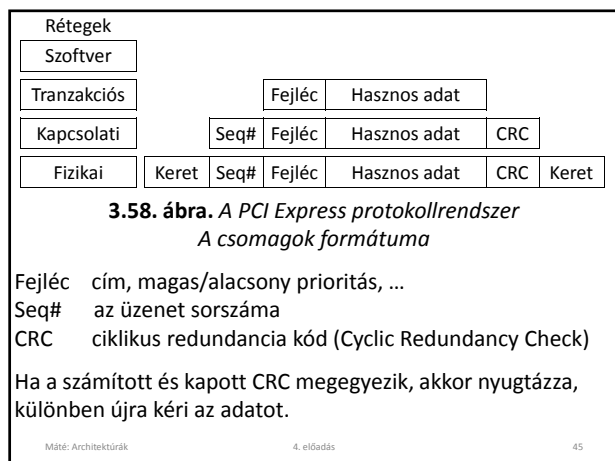


Hagyományos sín	PCI Express
Több leágazású sín	Központosított kapcsoló
Széles, párhuzamos sín	Keskeny, közvetlen soros kapcsolat
Bonyolult mester – szolga kapcsolat	Kicsi, csomagkapcsolt hálózat
	CRC kód: nagyobb megbízhatóság
	A csatlakozó kábel > 50 cm lehet
	Az eszköz kapcsoló is lehet
	Meleg csatlakoztatási lehetőség
	Kisebb csatlakozók: kisebb gép

- Nem kell nagy bővítőkártyával csatlakozni a sínhez
- A winchester a monitorba is kerülhet

Egy csatorna hasznos sávszélessége minimum 2 Gbps, de bíznak benne, hogy hamarosan 10 Gbps

Máté: Architektúrák 4. előadás 44



Input, output, interfész, I/O lapkák

UART (Universal Asynchronous Receiver and Transmitter) egy bájtot tud olvasni az adatsínról, és aztán sorosan továbbítja az eszközhöz (vagy fordítva), programmal konfigurálható (belső regiszterének beállításával):

- 5-8 bit szélesség,
- sebesség (50-19.200 bps),
- paritás ellenőrzés (páros, páratlan, nincs).

USART (Universal Synchronous Asynchronous Receiver and Transmitter): szinkron és aszinkron módon is tud működni.

Máté: Architektúrák 4. előadás 48

PIO (Parallel Input/Output, 3.60. ábra)

24 B/K vonal, amellyel TTL kompatibilis eszközökhöz (billentyűzet, kapcsolók, nyomtatók) tud kapcsolódni. Konfigurálható. Leggyakoribb az ábrán látható használat. Aszinkron eszközökhöz „kézfogas” logika van beépítve.



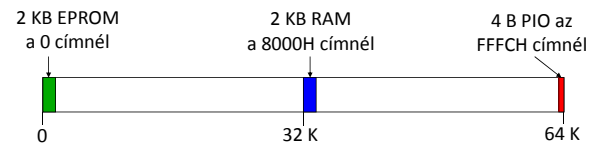
Máté: Architektúrák

4. előadás

49

Beviteli/Kiviteli (**B/K**, Input/Output, **I/O**) eszközök használata (chip selection):

- Valódi **B/K** eszköz.
- Memóriára leképezett **B/K** (memory-mapped **I/O**).
Pl.: **3.61. ábra**.



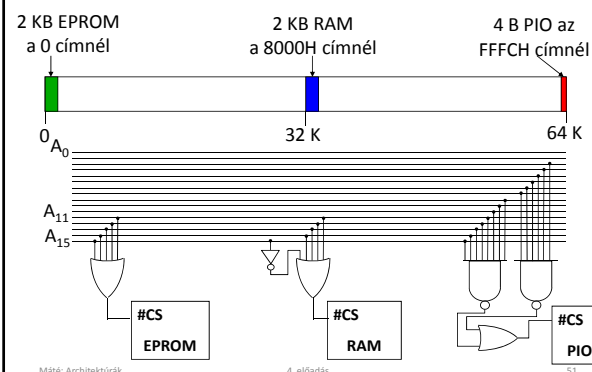
Máté: Architektúrák

4. előadás

50

3. 61-62. ábra. Teljes cím dekódolás.

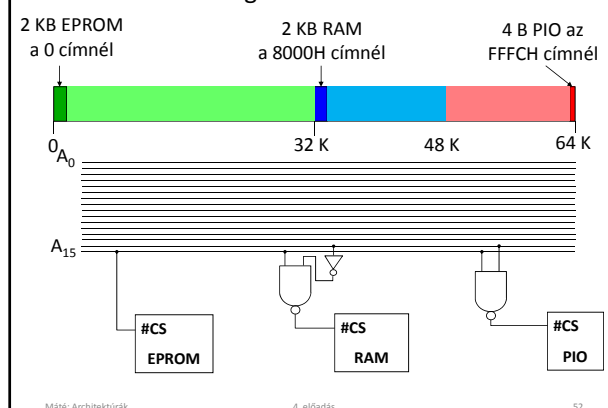
A_{15} a legmagasabb helyértékű bit.



Máté: Architektúrák

4. előadás

51

3.61-62. ábra. Részleges cím dekódolás.

Máté: Architektúrák

4. előadás

52

Feladatok

Hogy működik a tömb (array) processzor?
Mi a tömb (array) processzor előnye/hátránya?
Hogy működik a vektor processzor?
Mi a vektor processzor előnye/hátránya?
Mi a multiprocesszorok lényege?
Mi a közös/helyi memóriák szerepe a multiprocesszoros rendszerekben?
Miért nehéz sok processzoros rendszert építeni?
Mi a lényege multiszámítógépeknek?
Hogy tartják a kapcsolatot egymással a multiszámítógépek CPU-i?

Máté: Architektúrák

4. előadás

53

Feladatok

Hogy működik a tömb (array) processzor?
Mi a tömb (array) processzor előnye/hátránya?
Hogy működik a vektor processzor?
Mi a vektor processzor előnye/hátránya?
Mi a multiprocesszorok lényege?
Mi a közös/helyi memóriák szerepe a multiprocesszoros rendszerekben?
Miért nehéz sok processzoros rendszert építeni?
Mi a lényege multiszámítógépeknek?
Hogy tartják a kapcsolatot egymással a multiszámítógépek CPU-i?

Máté: Architektúrák

4. előadás

54

Feladatok

Milyen adat típusokat ismer?
 Milyen karakter kódolásokat ismer? Ismertesse ezek lényegét!
 Mit jelent a **CPU** rövidítés?
 Hogy tartja a kapcsolatot a **CPU** a környezetével?
 Milyen lábai vannak egy **CPU**-nak?
 Miért lényeges a cím és adat lábak száma?
 Hány cím adható meg **k** címvezetéken?
 Milyen lépésekből áll egy utasítás betöltése?
 Mi a központi memória feladata?

Máté: Architektúrák

4. előadás

55

Feladatok

Mit nevezünk sínnek?
 Mi a sínprotokoll?
 Mi a mester, és mi a szolga?
 Mit nevezünk sín vezérlőnek/vevőnek/adó-vevőnek?
 Mi a sávszélesség?
 Mit nevezünk sín vezérlésnek?
 Mit nevezünk sín ütemezésnek?
 Hogyan történik egy adat kiírása a memóriába?

Máté: Architektúrák

4. előadás

56

Feladatok

Mi a sínszélesség?
 Mi a sín aszimmetria?
 Hogy növelhető egy sín sávszélessége?
 Miért nem növelhető szabadon a sín szélessége?
 Miért nem növelhető szabadon a sín sávszélessége?
 Miért volt szükség a sínek szabványosítására?
 Mi az alaplap?
 Mit jelent a sokszorozott (multiplexed) sín?
 Milyen hatása van a sokszorozott sín használatának?

Máté: Architektúrák

4. előadás

57

Feladatok

Mi a sín időzítés, és miért fontos?
 Hogy működik a szinkron/aszinkron sín?
 Mire szolgál a mester/szolga szinkronizáció?
 Mi a teljes kézfogás?
 Milyen sín ütemezőket ismer?
 Hogy működik a centralizált sín ütemező?
 Hogy működik a decentralizált sín ütemező?
 Hogy működik a PCI sín ütemezője?
 Milyen sín műveleteket ismer?
 Miért előnyös a blokkos átvitel?

Máté: Architektúrák

4. előadás

58

Feladatok

Milyen sítet ismer?
 Ismertesse az **ISA** sítet!
 Ismertesse az **EISA** sítet!
 Ismertesse az **PCI** sítet!
 Hogy használható több sín egy gépen belül?
 Miért volt fontos az **ISA** sín megtartása fejlettebb sín alkalmazása esetén?
 Hogy történik a **PCI** sín ütemezése?
 Miért használnak külön sítet a memória eléréséhez?
 Hogy illeszkedik a **PCI** sín a memória sínhez?

Máté: Architektúrák

4. előadás

59

Feladatok

Milyen igények kielégítésére szolgál az általános soros sín (**USB**)?
 Milyen vezetékeket tartalmaz az **USB** sín, és miért?
 Miért kényelmes az **USB** sín használata?
 Mi a központi elosztó (root hub), és hogy működik?
 Mi történik eszköz **USB** porthoz csatlakoztatásakor?
 Mit jelent egy eszköz konfigurálása?
 Mi a keret (frame), és milyen kereteket ismer?
 Mi a csomag, és milyen csomagokat ismer?
 Milyen irányban haladhatnak a csomagok?

Máté: Architektúrák

4. előadás

60

Feladatok

Mire szolgál a **SOF** csomag?
 Mire szolgál az **IN/OUT** csomag?
 Mire szolgál az **ACK/NAK** csomag?
 Mi a **CRC**?
 Mire szolgál a **PCI Express**?
 Hogy kapcsolódik a **PCI Express** a **CPU**-hoz?
 Hogy kapcsolódik a **PCI Express** az eszközökhöz?
 Milyen előnyei vannak a **PCI Express**-nek a sínnel szemben?
 Milyen rétegei vannak a **PCI Express** protokollnak?

Máté: Architektúrák

4. előadás

61

Feladatok

Mi az **UART**?
 Mire szolgál az **UART**?
 Paritás ellenőrzés szempontjából hogy működhet az **UART**?
 Mi az **USART**?
 Milyen digitális áramkörök segítségével tudja megvalósítani a párhuzamos-soros átalakítást?
 Mit jelent a **CS#** jelölés egy chip lábán?
 Mit jelent a **PIO** rövidítés?
 Milyen lábai vannak **PIO** lapkának?
 Hány regisztere van **PIO** lapkának?

Máté: Architektúrák

4. előadás

62

Feladatok

Mi biztosítja, hogy kapcsolók állapotának lekérdezésére és lámpák működtetésére is használható a **PIO** lapka?
 Mit jelent a valódi/memóriára leképezett **I/O**?
 Melyiknek mi az előnye, hátránya?
 Mit nevezünk teljes/részleges cím dekódolásnak?
 Melyiknek mi az előnye, hátránya?
 Mit nevezünk port-nak?

Máté: Architektúrák

4. előadás

63

Az előadáshoz kapcsolódó**Fontosabb témák**

Processzor szintű párhuzamosítás
 Adat típusok, karakter kódolás
 CPU, Sínek. Sín protokoll. Mester – szolga. Sín vezérlő, vevő, adóvevő. Sáv szélesség, sín szélesség.
 Sokszorozott sín.
 Sín időzítés: szinkron, aszinkron sín, teljes kézfogás.
 Sín ütemezés.

Máté: Architektúrák

4. előadás

64

Az előadáshoz kapcsolódó**Fontosabb témák**

Sín műveletek.
 Általános soros sín.
 PCI express
 Parallel Input/Output (PIO), valódi és memóriára leképezett I/O eszköz. Teljes és részleges cím dekódolás

Máté: Architektúrák

4. előadás

65