

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

5. előadás

1

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 (címnak 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 báj*2, 30 kép/sec).
 lemez → memória → képernyő

Máté: Architektúrák

5. előadás

2

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:

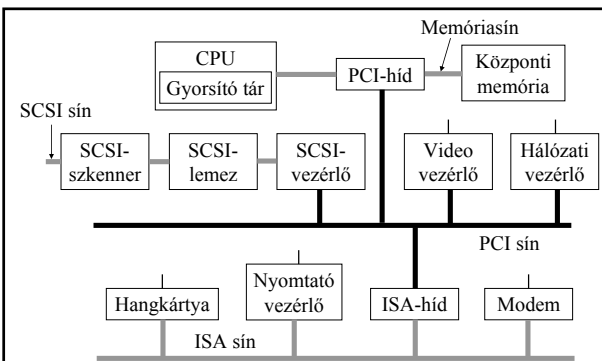
- a memóriához lassú,
- nem kompatibilis az ISA bővítőkártyákkal.

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

Máté: Architektúrák

5. előadás

3



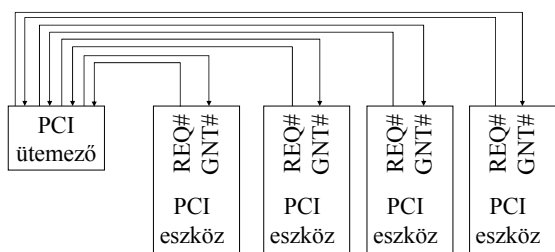
2.30. ábra. Egy tipikus PC PCI, SCSI és ISA sínnel 2000 körül

Máté: Architektúrák

5. előadás

4

PCI sín ütemezés (kiosztás): request, grant.



3.54. ábra. A PCI sín centrális ütemezőt használ.

Máté: Architektúrák

5. előadás

5

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

5. előadás

6

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, 3.54. 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 5. előadás 7

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 5. előadás 8

A keret lehet

- **Control** – vezérlő:
 - Eszköz konfigurálás,
 - Parancs,
 - Állapot lekérdezés.
- **Isochronous** – izoszinkron: 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 5. előadás 9

A csomag lehet

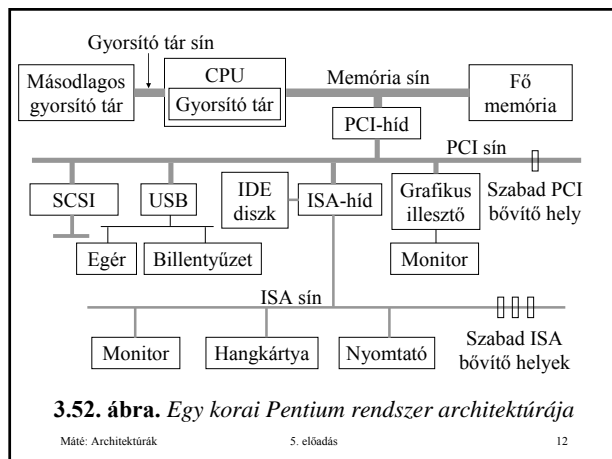
- **Token** – parancs (központ küldi az eszköznek):
 - **SOF**.
 - **IN** – be: adatokat kér az eszköztől.
 Az **IN** parancsban meg lehet adni, melyik eszköztől milyen adatokra van szükség.
 - **OUT** – ki: adatok fogadására kéri az eszközt.
 - **SETUP** – beállítás: konfigurálja az eszközt.

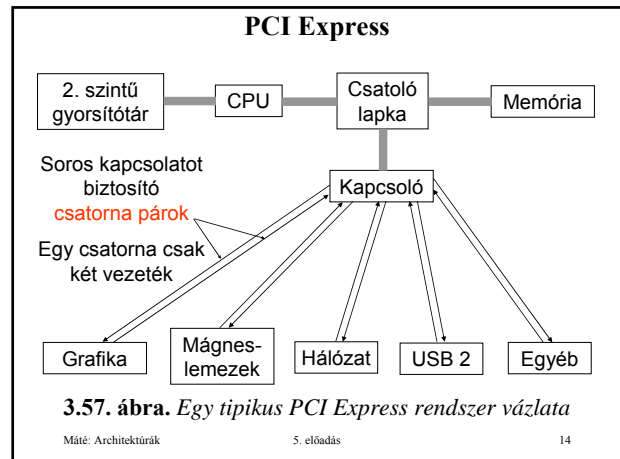
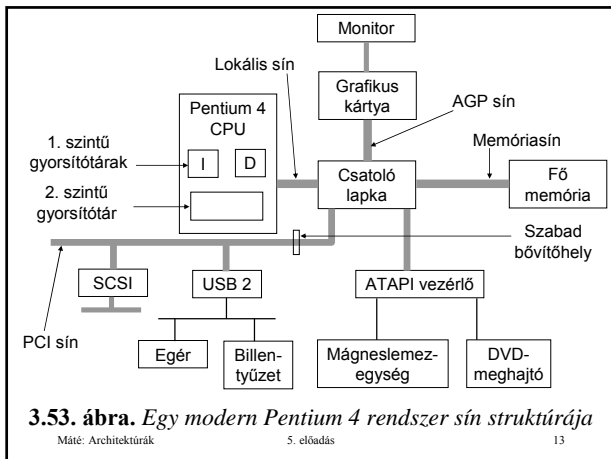
Máté: Architektúrák 5. előadás 10

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 adatsomagot hibátlanul vettem,
 - **NAK**: CRC hibát észleltem,
 - **STALL**: kérem, várjon, el vagyok foglalva.
- **Special** – speciális.

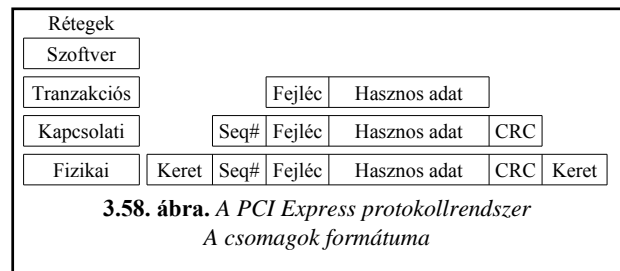
Máté: Architektúrák 5. előadás 11





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 bővítőkérdővel csatlakozni a sínhez
 - A winchester a monitorba is kerülhet
- Egy csatorna hasznos sáv szélessége minimum 2 Gbps, de bíznak benne, hogy hamarosan 10 Gbps
- Máté: Architektúrák 5. előadás 15



Fejléc cím, magas/alacsony prioritás, ...
Seq# az üzenet sorszáma
CRC ciklikus redundanciakód (Cyclic Redundancy Check)
Ha a számított és kapott CRC megegyezik, akkor nyugtázza, különben újra kéri az adatot.

Máté: Architektúrák 5. előadás 16

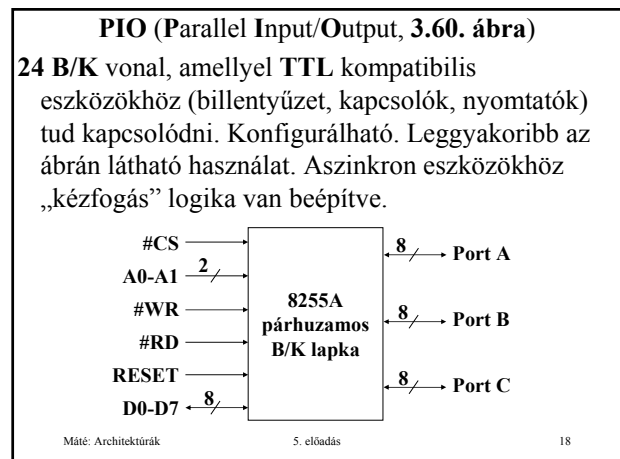
Input, output, interfész, I/O lapkák

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 5. előadás 17



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 5. előadás 19

3. 61-62. ábra. Teljes cím dekódolás.
 A_{15} a legmagasabb helyértékű bit.

Máté: Architektúrák 5. előadás 20

3.61-62. ábra. Részleges cím dekódolás.

Máté: Architektúrák 5. előadás 21

Mikroarchitektúra szint

Feladata az ISA (Instruction Set Architecture – gépi utasítás szint) megvalósítása.

Nincs rá általánosan elfogadott, egységes elv.

A ISA-szintű utasítások „függvények”, ezeket egy főprogram hívja meg végtelen ciklusban.

A függvények a mikroarchitektúra szintjén valósulnak meg (mikroprogram).

A mikroprogram változói (a regiszterek) definiálják a számítógép állapotát, pl.: PC (Program Counter, utasításszámláló).

Máté: Architektúrák 5. előadás 22

Mic-1 Adatút (Data Path, 4.1. ábra)

Memória vezérlő regiszterek

ALU, léptető

Vezérlő jelek

- B síne írja a regisztert
- C sít a regiszterbe írja
- memóriába
- memóriából

ALU bemenetei:

- H (Holding – tartó), B sín.

Egy cikluson belül lehetséges egy regiszterből olvasni és az eredményt akár ugyanoda visszaírni 4.3. ábra.

ALU vezérlés

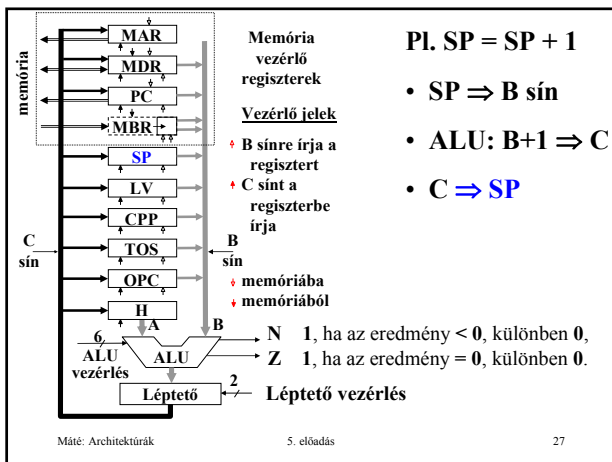
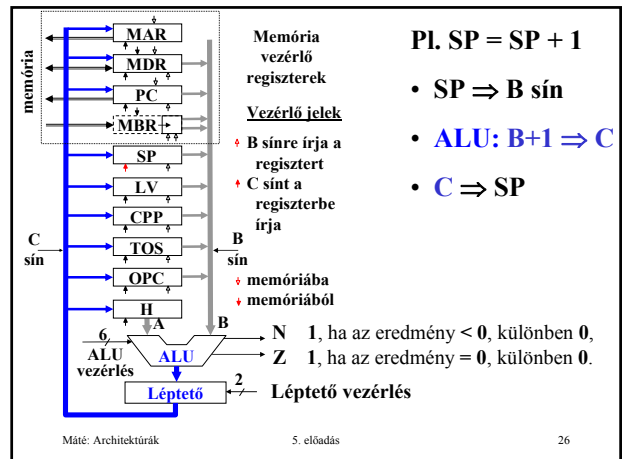
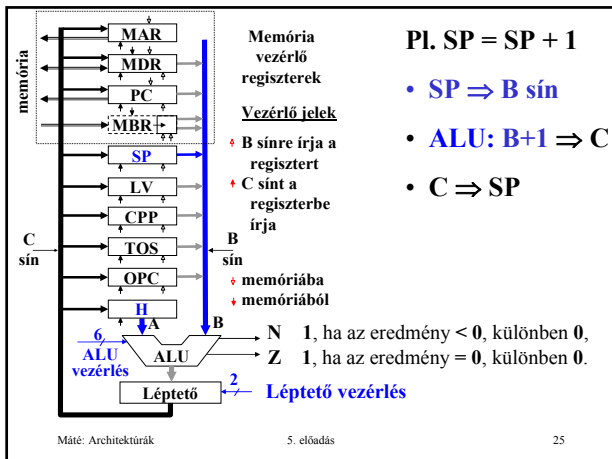
Léptető

Máté: Architektúrák 5. előadás 23

Pl. $SP = SP + 1$

- $SP \Rightarrow B$ sín
- $ALU: B+1 \Rightarrow C$
- $C \Rightarrow SP$

Máté: Architektúrák 5. előadás 24



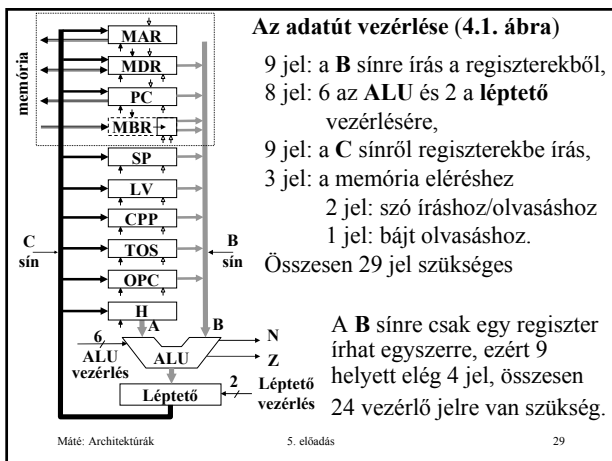
Memóriaműveletek (4.1. ábra)

- Szócímezés:** 32 bites adat írása, olvasása. szó cím = $4 * (\text{bájt cím})$, a túlszorduló bitek elvesznek
- MAR** (Memory Address Register)
- MDR** (Memory Data Register)
- Bájtcímzés:** gépi szintű utasítás bájt olvasás.
 - PC** (Program Counter): bájt cím,
 - MBR** (Memory Byte Register): bájt.

MBR kétfajta értelmezése (két vezérlőjel):

- MBR:** MBR előjel kiterjesztéssel kerül a B síre,
- MBRU:** MBR előjel nélküli kiterjesztéssel kerül a B síre.

Máté: Architektúrák 5. előadás 28



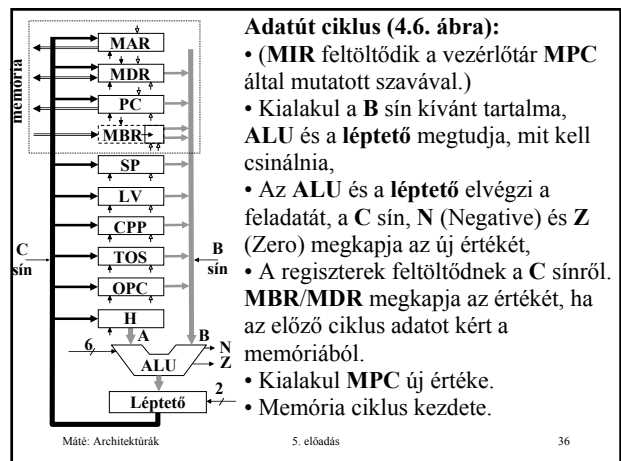
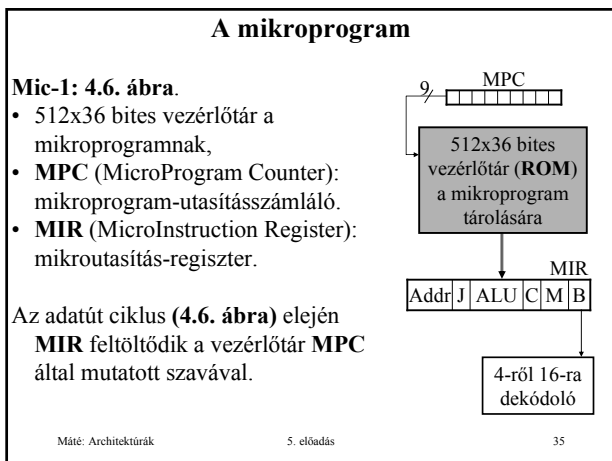
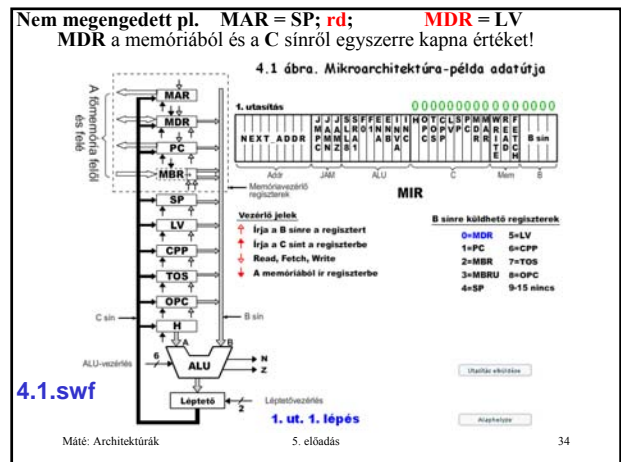
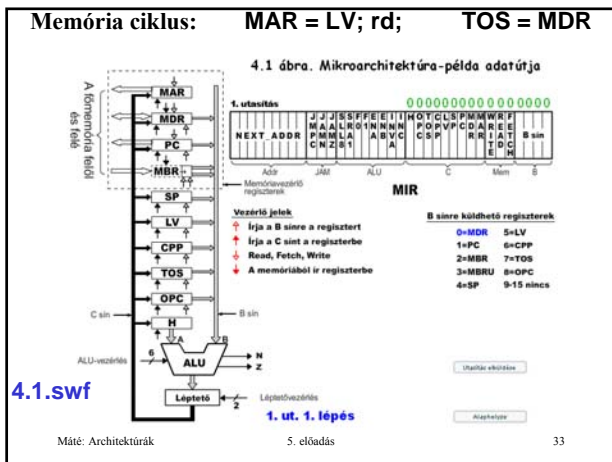
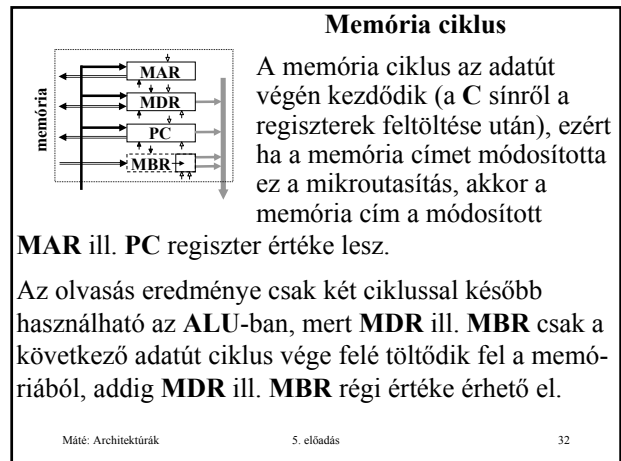
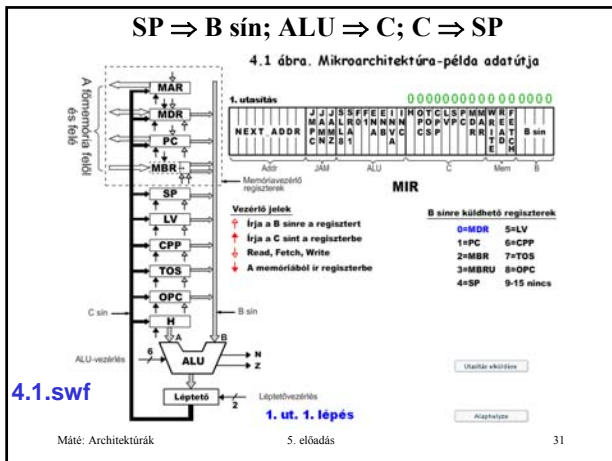
Mikroutasítások

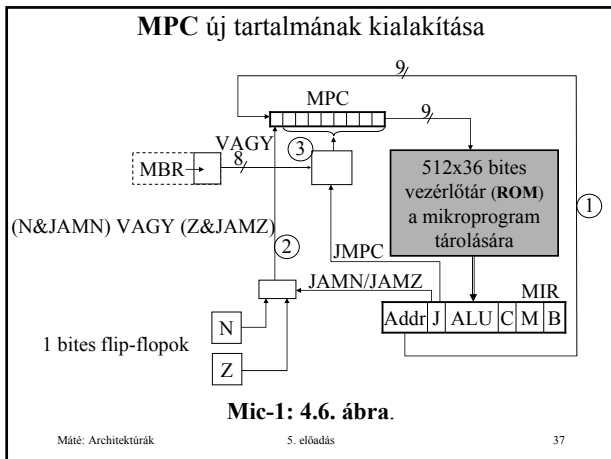
24 bit: az adatút vezérléséhez
 9 bit: a következő utasítás címének megadásához,
 3 bit: a következő utasítás kiválasztásának módjára.
 Ez adja a 36 bites mikroutasítást: **4.5. ábra.**

9	3	8	9	3	4
NEXT ADDRESS	JMPC JAMN JAMZ	SLL8 SRA1 F0 F1 ENA ENB INVA INC	H OPC TOS LV SP PC MDR MAR	WRITE READ FETCH	B sín
Addr	JAM	ALU	C	Mem	B

0 = MDR	1 = PC	2 = MBR	3 = MBRU	4 = SP
5 = LV	6 = CPP	7 = TOS	8 = OPC	9-15 semmi

Máté: Architektúrák 6. előadás 30





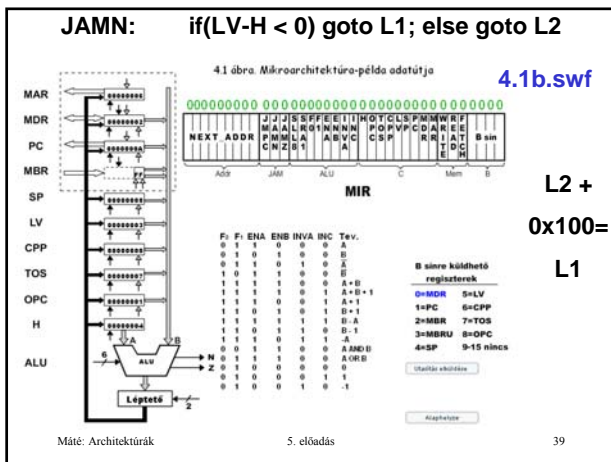
MPC új tartalma, JAMN/JAMZ

- A 9 bites következő cím (Addr) az MPC-be kerül.
- (JAMN ÉS N) VAGY (JAMZ ÉS Z) és MPC legmagasabb bitjének logikai vagy kapcsolata képződik MPC legmagasabb helyértékén. Pl.:

Cím	Addr	JAM	Adatút	vezérlő bitek
0x75	0x092	001	...	JAMZ = 1

esetén a mikroprogram a
0x092 címen folytatódik, ha **Z = 0**,
0x192 címen folytatódik, ha **Z = 1**.
 Feltételes ugrás – elágazás – a mikroprogramban.

Máté: Architektúrák 5. előadás 38



Feladatok

Milyen sín műveleteket ismer?
 Miért előnyös a blokkos átvitel?

Máté: Architektúrák 5. előadás 40

Feladatok

Milyen sít ismer?
 Ismertesse az **ISA** sít!
 Ismertesse az **EISA** sít!
 Ismertesse az **PCI** sít!
 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 sít a memória eléréséhez?
 Hogy illeszkedik a **PCI** sín a memória sínhez?

Máté: Architektúrák 5. előadás 41

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 5. előadás 42

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

5. előadás

43

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

5. előadás

44

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

5. előadás

45

Feladatok

Mi a mikroarchitektúra szint feladata?
Hogy valósítja meg a feladatát?
Mi az adatút?
Milyen sínjei vannak a Mic-1 -nek?
Mely regiszterek csatlakoznak az **A** sínhez?
Mely regiszterek csatlakoznak a **B** sínhez?
Mely regiszterek csatlakoznak a **C** sínhez?
Milyen jelek szükségesek az adatút vezérléséhez?
Mire szolgál az **SP** regiszter két vezérlő bemenete?
Mire szolgál az **MBR** regiszter két vezérlő bemenete?

Máté: Architektúrák

5. előadás

46

Az előadáshoz kapcsolódó

Fontosabb tételek

Sín műveletek.
Általános soros sín. PCI express
Parallel Input/Output (PIO), valódi és memóriára leképezett I/O eszköz. Teljes és részleges cím dekódolás
Mikroarchitektúra szint. Mikroutasítások: Mic-1. A Mic-1 működése. Memória műveletek. ALU és az adatút vezérlése.

Máté: Architektúrák

5. előadás

47

Az előadáshoz kapcsolódó

Fontosabb tételek

Sín műveletek.
Általános soros sín. PCI express
Parallel Input/Output (PIO), valódi és memóriára leképezett I/O eszköz. Teljes és részleges cím dekódolás
Mikroarchitektúra szint. Mikroutasítások: Mic-1. A Mic-1 működése. Memória műveletek. ALU és az adatút vezérlése.

Máté: Architektúrák

5. előadás

48