

Programozható logikai tömbök: PLA (3.15. ábra)
(Programmable Logic Array).

Ha ezt a biztosítékot kiégetjük, akkor nem jelenik meg B# az 1-es **ÉS** kapu bemenetén

Ha ezt a biztosítékot kiégetjük, akkor az 1-es **ÉS** kapu kimenete nem jelenik meg az 5-ös **VAGY** kapu bemenetén

24 bemenő vonal
12 bemenő jel
50 bemenő vonal
6 kimenet

Máté: Architektúrák 3. előadás 1

3.15. ábra. 4 bemenetű, 2 kimenetű programozható logikai tömb

3.15.swf

Máté: Architektúrák 3. előadás 2

Aritmetikai áramkörök

A kombinációs áramkörökön belül külön csoportot alkotnak.

Léptető:

3.16. ábra 1 bittel balra/jobbra léptető

C=1: jobbra,
C=0: balra léptet.

Máté: Architektúrák 3. előadás 3.16.swf 3

Összeadók:

3.17. ábra (a) 1 bites összeadás igazságtáblája. (b) Fél összeadó áramkör

A	B	Összeg	Átvitel
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Máté: Architektúrák 3. előadás 3.17. ábra, 3.17.swf 4

Összeadók:

3.18. ábra (a) A teljes összeadó igazságtáblája (b) Teljes összeadó áramkör

A	B	Átvitel be	Összeg	Átvitel ki
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Máté: Architektúrák 3. előadás 3.18. ábra, 3.18.swf 5

Aritmetikai-logikai egység: bitszelet (bit slice, 3.19. ábra), F0, F1 -től függően **ÉS, **VAGY**, **NEGÁCIÓ** vagy +**

3.19. ábra 1 bites ALU

Máté: Architektúrák 3. előadás 3.19.swf 6

• átvitel továbbterjesztő összeadó (ripple carry adder):

4.2. ábra. Az ALU jelenlegi hozzászólású és az elvégzett tevékenység

F0	F1	ENA	ENB	INVA	INC	Tevékenység
0	1	1	0	0	0	A
0	1	0	1	0	0	B
0	1	1	0	1	0	A
1	1	1	0	0	0	B
1	1	1	1	0	0	A+B
1	1	1	1	0	1	A+B+1
1	1	0	0	1	0	A-1
1	1	0	1	0	1	B+1
1	1	1	1	1	1	B-A
1	1	0	1	1	0	B-1
1	1	1	0	1	1	-A
0	0	1	1	0	0	A AND B
0	1	1	0	0	0	A OR B
0	1	0	0	0	0	0
1	1	0	0	0	1	1
0	1	0	0	1	0	-1

Csak a „Tevékenység” oszlop ismerete szükséges.

Máté: Architektúrák 3. előadás 3.19b.swf 7

• átvitel továbbterjesztő összeadó (ripple carry adder):

Egy helyértéken csak akkor helyes az eredmény, ha az alacsonyabb helyértékeken már kialakult az átvitel értéke. Minél több bites az ALU, annál tovább tart az összeadás elvégzése.

Máté: Architektúrák 3. előadás 8

• átvitel kiválasztó összeadó (carry select adder) eljárás:

Máté: Architektúrák 3. előadás 3.19c.swf 9

Nem kombinációs áramkörök

Óra (clock, 3.21. ábra): ciklusidő (cycle time).
 Pl.: 500 MHz - 2 nsec.
 Finomabb felbontás készlettel.
 Aszimmetrikus óra.

Máté: Architektúrák 3. előadás 10

Memória: „Emléskiz” az utolsó beállításra.

Tároló: Szint vezérelt (level triggered).

SR tároló (Set Reset latch, 3.22. ábra, 3.22.swf).
 Stabil állapot: a két kimenet **0, 1** vagy **1, 0**.
S (set), **R** (reset) bemenet. ($Q\# \equiv \bar{Q}$)

3.22. ábra NEM-VAGY tároló

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

Máté: Architektúrák 3. előadás 11

3.23. ábra Időzített SR-tároló

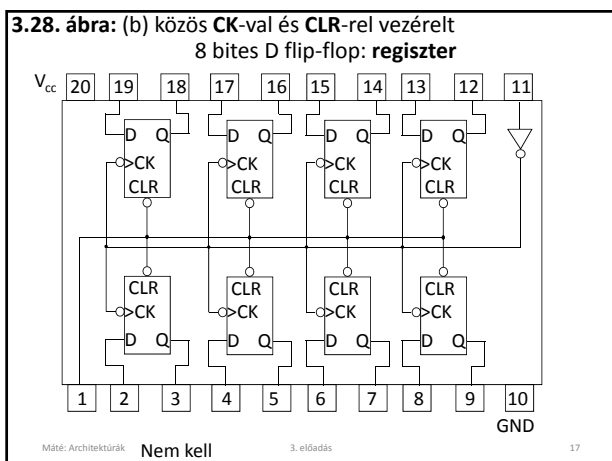
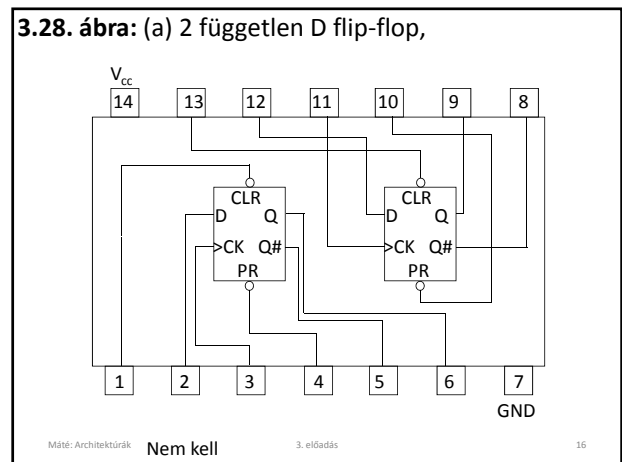
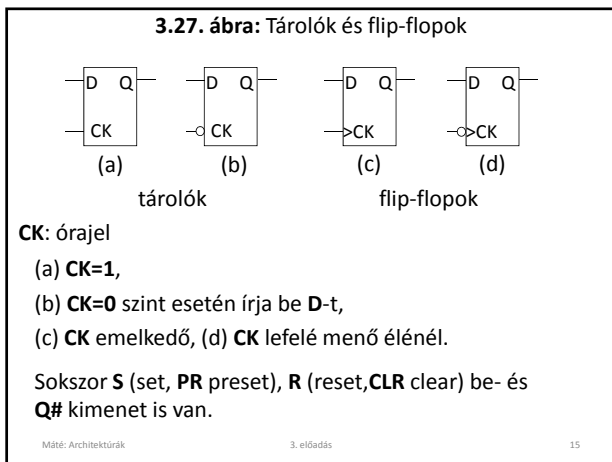
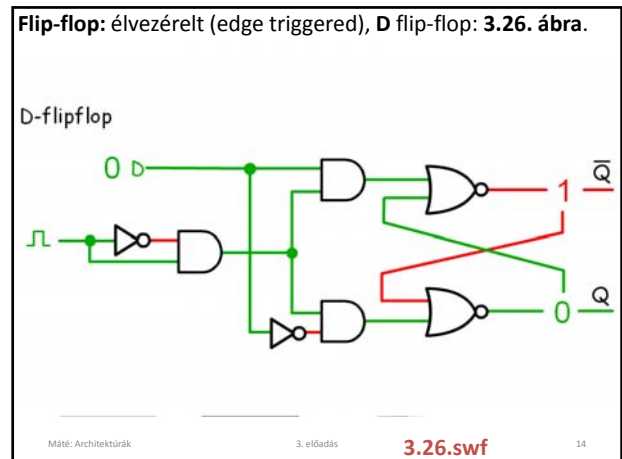
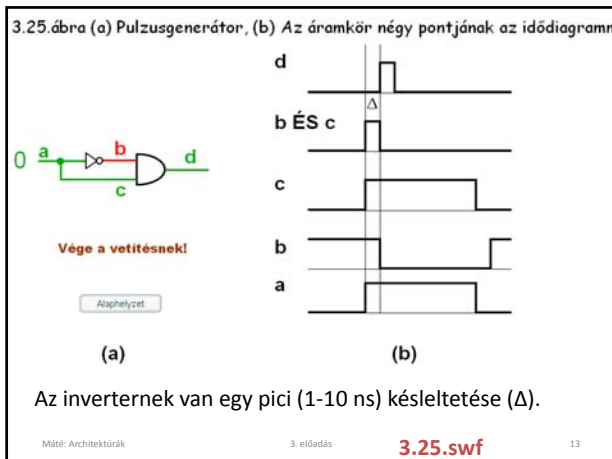
A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

3.24. ábra Időzített D-tároló

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

Mindkét SR tároló indeterminisztikussá válna, ha **S = R = 1** egyszerre fordulna elő.

Máté: Architektúrák 3.23.swf 3. előadás 3.24.swf 12



Alapvető digitális logikai áramkörök

Integrált áramkör (IC, Integrated Circuit, chip, lapka) 5x5 mm² szilícium darab kerámia vagy műanyag lapon (tokban), lábakkal (pins). Négy alaptípus:

- SSI (Small Scale Integrated 1-10 kapu),
- MSI (Medium Scale ..., 10-100 kapu),
- LSI (Large Scale..., 100-100 000 kapu),
- VLSI (Very Large Scale ..., > 100 000 kapu).

Máté: Architektúrák 3. előadás 18

Memória szervezése

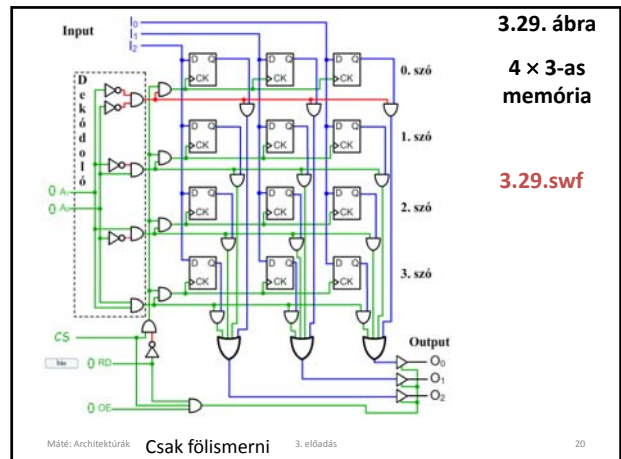
Elvárás: szavak címezhetősége.

3.29. ábra: Négy db három bites szó. Bemenetek: három a vezérléshez,
 – **CS** (Chip Select): lapka választás,
 – **RD** (ReaD): 1: olvasás, 0: írás választása,
 – **OE** (Output Enable): kimenet engedélyezése.
 kettő a címzéshez (dekódoló),
 három a bemenő adatoknak,
 három adat kimenet.

Máté: Architektúrák

3. előadás

19



3.29. ábra

4 × 3-as memória

3.29.swf

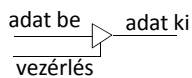
Máté: Architektúrák

3. előadás

20

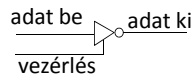
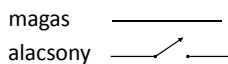
Memória szervezése

Az igazi memóriáknál a bemenet és kimenet közös (kevesebb lábra van szükség): Nem invertáló és invertáló pufferek (ezek három állapotú eszközök, **tri-state device, 3.30. ábra**).



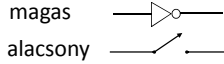
nem invertáló puffer

Ha a vezérlő jel



invertáló puffer

Ha a vezérlő jel



Máté: Architektúrák

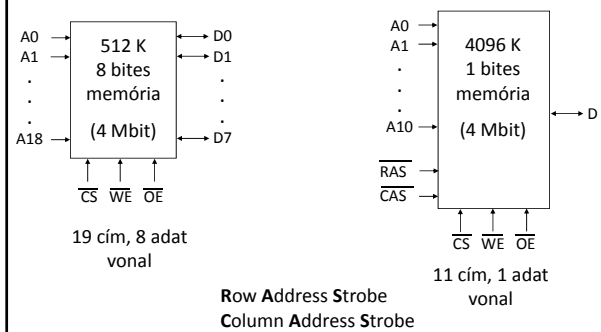
3. előadás

21

Memórialapok

Előnyös, ha a szavak száma **2 hatvány**.

4 Mbit-es memória kétféle szervezése: **3.31. ábra**.



Máté: Architektúrák

3. előadás

22

Memórialapok

A jel (bemenet) **beállított** (asserted) vagy **negált**.
CS beállított: **1**, de **CS#** beállított: **0**

- a) **512 K** bájtos elrendezés: **19** cím, **8** adat vonal.
- b) **2048*2048** bites elrendezés: **11** cím, **1** adat vonal:
 Bit kiválasztás sor- (**RAS**: Row Address Strobe) és oszlopindex **CAS** (Column ...) segítségével.
 Gyakran alkalmazzák nagyobb memóriáknál, bár a két cím megadása lassíthat.

Nagyobb memóriáknál **1, 4, 8, 16** bites kimeneteket is használnak.

Máté: Architektúrák

3. előadás

23

Központi memória (2.9. ábra)

A programok és adatok tárolására szolgál.

Bit: a memória alapegysége, egy 0-t vagy 1-et tartalmazhat.

Memória rekesz (cella): több bit együttese. Minden rekesz ugyanannyi bitből áll. Minden rekeszhez hozzá van rendelve egy szám, a **rekesz címe**. Egy rekeszre a címével hivatkozhatunk. A rekesz a legkisebb címezhető egység.

Máté: Architektúrák

3. előadás

24

Cím Rekesz/cella

0

--	--	--	--

 A rekesz hossza manapság legtöbbször 8 bit (byte ~ bájt).

1

--	--	--	--

 n a memória cellák száma

...

n-1

--	--	--	--

← Rekesz hossza →

Központi memória (2.9. ábra)

Máté: Architektúrák 3. előadás 25

A bitek száma rekeszenként néhány számítógép-történetileg érdekes, kereskedelmi forgalomba került gépen (2.10. ábra)

Számítógép	Bit
Burroughs B1700	1
IBM PC	8
DEC PDP-8	12
IBM 1130	16
DEC PDP-15	18
XDS 940	24
Electrologica X8	27
XDS Sigma 9	32
Honeywell 6180	36
CDC 3600	48
CDC Cyber	60

Máté: Architektúrák Nem kell 4. előadás 26

Bájtsorrend

A legtöbb processzor több egymás utáni bájtal is tud dolgozni (szó – word, ...).

A legmagasabb helyértékű bájt a szóban a legalacsonyabb címen: legmagasabb címen: **nagy (big) endian** **kis (little) endian**

MSBfirst (SPARC) **LSBfirst (Pentium)**

Most/Least Significant Byte first

Ha egy 32 bites szó bájtjainak értéke rendre: a, b, c, d , akkor a szó értéke: $a*256^3+b*256^2+c*256+d$ $a+b*256+c*256^2+d*256^3$

Máté: Architektúrák 3. előadás 27

Bájtsorrend (2.11. ábra)

A memória címek úgy vannak fölírva, hogy a legmagasabb helyértékű bájt van bal oldalon.

Cím	Nagy endian				Kis endian				Cím
0	0	1	2	3	3	2	1	0	0
4	4	5	6	7	7	6	5	4	4
8	8	9	10	11	11	10	9	8	8
12	12	13	14	15	15	14	13	12	12

← 32 bites szó → ← 32 bites szó →

Máté: Architektúrák 3. előadás 28

Bájtsorrend (12. ábra)

A szövegek karaktereit mindkét esetben növekvő bájt sorrendben helyezik el

Cím	nagy endian				kis endian			
0	0	1	2	3	3	2	1	0
4	4	5	6	7	7	6	5	4
	12	34	56	78	12	34	56	78

A TEXT szöveg és az 12345678 hexadecimális szám elhelyezése a két géptípuson

Problémák a gépek közötti kommunikációban!

Máté: Architektúrák 3. előadás 29

Kódolás: adat + ellenőrző bitek = kódszó.

Két kódszó Hamming távolsága: az eltérő bitek száma. Pl.: 11001 és 11011 (Hamming) távolsága = 1.

Hibaérzékelő kód: bármely két kódszó távolsága > 1: paritás bit.

d hibás bit javítása: a kódszavak távolsága > 2d.

Egy hibát javító kód (2.13. ábra):

m adat, r ellenőrző bit, összesen $n = m + r$. 2^m „jó” szó, + minden „jó” szónak n db „egyhibás” szomszédja van, ezért $(1+n)2^m \leq 2^n = 2^{m+r}$, 2^m -mel egyszerűsítve: $m+r+1 \leq 2^r$, vagy másképp: $m+r < 2^r$ szükséges.

Máté: Architektúrák 3. előadás 30

RAM (Random Access Memory)

- **Statikus RAM (SRAM).** D flip-flop elemekből épül fel. Amíg áram alatt van, tartja a tartalmát. Elérési idő: néhány nsec (cache-nek jók).
- **Dinamikus RAM (DRAM):** minden bit egy tranzisztor és egy kondenzátor: néhány msec-onként frissíteni kell, de nagyobb adatsűrűség érhető el. Elérési idő: néhány tíz nsec (főmemóriák).
 - régi: **FPM** (Fast Page Mode) sor-, oszlopcím.
 - újabb: **EDO** (Extended Data Output) lehet új memóriahivatkozás, mielőtt az előző befejeződik.
- **SDRAM** (Synchronous **DRAM**). A központi óra vezérli. Blokkos átvitel. Újabban: **DDR** (Double Data Rate). Az órajel föl- és lefutó élénél is van adatátvitel.

Máté: Architektúrák 3. előadás 31

ROM (Read-Only Memory)

ROM: gyárilag kialakított tartalom.

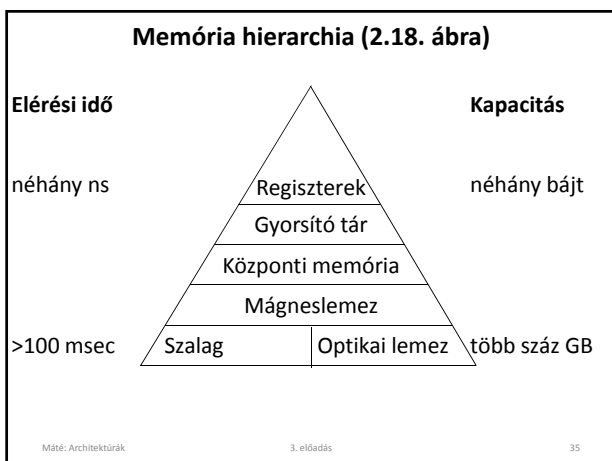
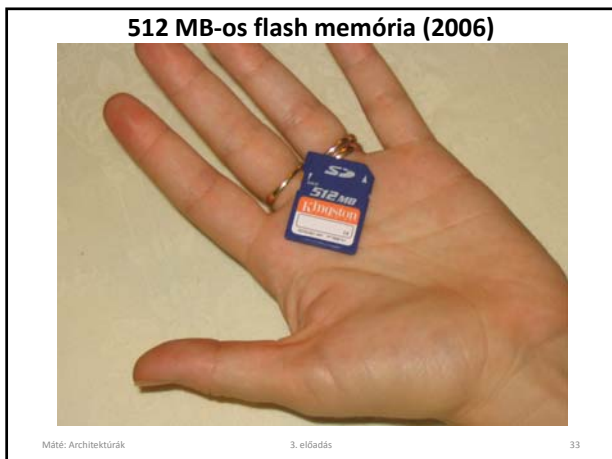
PROM (Programmable ROM): a tartalom biztosítékok kiegészítésével alakul ki (a **PLA**-khoz hasonlóan, **3.15. ábra**).

EPROM (Erasable PROM): a biztosítékok speciális fényvel kiolvaszthatók és „kijavíthatók”.

EEPROM: elektromos impulzusokkal.

Flash memória: törlés és újraírás csak blokkonként. Kb. 100 000 használat után „elkopnak”. Ilyen van a legtöbb MP3 lejátszóban, digitális fényképezőgépben ...

Máté: Architektúrák 3. előadás 32



Feladatok

Mit nevezünk kombinációs áramkörnek?
 Milyen kombinációs áramköröket ismer?
 Rajzoljon multiplexert! Hogy működik?
 Rajzoljon demultiplexert! Hogy működik?
 Rajzoljon dekódolót! Hogy működik?
 Mire használható a multiplexer, és hogyan?
 Mire használható a **PLA**, és hogyan?
 Milyen aritmetikai áramköröket ismer?
 Rajzoljon léptetőt! Hogy működik?
 Rajzoljon „fél összeadó”! Hogy működik?
 Mi indokolja a „fél összeadó” elnevezést?

Máté: Architektúrák 3. előadás 36

Feladatok

Rajzoljon teljes összeadót! Hogy működik?
 Milyen részei vannak az **ALU**-nak?
 Milyen be- és kimenetei vannak az 1 bites **ALU**-nak?
 Milyen műveletek végezhetők el az **ALU**-val?
 Az **ALU** milyen vezérlő bemenetei esetén lesz **1** az eredmény?
 Rajzoljon átvitel továbbterjesztő összeadót! Hogy működik?
 Hogyan csökkenthető az összeadásnál az átvitelekből származó idő?
 Rajzoljon átvitel kiválasztó összeadót! Hogy működik?

Máté: Architektúrák

3. előadás

37

Feladatok

Milyen nem kombinációs áramköröket ismer?
 Kombinációs áramkör-e az **ALU**?
 Hogy érhetünk el az órajelnél finomabb időzítést?
 Hány stabil állapota van az **SR** tárolónak?
 Rajzoljon **SR** tárolót!
 Rajzoljon időzített **SR** tárolót!
 Rajzoljon időzített **D** tárolót!
 Rajzoljon pulzusingenerátort! Mi a működési elve?
 Mi a különbség a tároló és a flip-flop között?
 Hogy működik az invertáló és a nem invertáló puffer?
 Miért használnak a memóriáknál invertáló vagy nem invertáló puffert?

Máté: Architektúrák

3. előadás

38

Feladatok

Hogy címezhető meg n címlábon 2^n -nél nagyobb memória?
 Mit jelent, hogy a **CS/CS#** bemenet beállított/negált?
 Mi a **RAM**?
 Jellemezze az **SRAM**-ot!
 Jellemezze a **DRAM**-ot!
 Mit jelent az **FPM** rövidítés?
 Mit jelent az **EDO** rövidítés?
 Jellemezze az **SDRAM**-ot!
 Mit jelent az **DDR** rövidítés?

Máté: Architektúrák

3. előadás

39

Feladatok

Mi a memória cella/rekesz?
 Mit jelent a big endian kifejezés?
 Milyen problémát okoz az eltérő bájtrend?
 Mi a Hamming távolság?
 Mekkora a hexadecimális E6 és C7 Hamming távolsága?
 Hány ellenőrző bit szükséges 256 kódszó 1 hibát javító kódolásához?

Máté: Architektúrák

3. előadás

40

Feladatok

Az alábbi memóriák közül melyik lehetséges, melyik ésszerű? Indokolja meg!

10 bites címek	1024 db	8 bites rekesz
10	1024	12
9	1024	10
11	1024	10
10	10	1024
1024	10	10

Egy régi gépnek 8192 szavas memóriája volt. Miért nem 8000?

Máté: Architektúrák

3. előadás

41

Feladatok

A memória 100-adik bájtjától a 01234567H 4 bájtos számot és – folytatólagosan – az abcd szöveget helyeztük el. Mi az egyes bájtok tartalma, ha a memória big/little endian szervezésű?

Máté: Architektúrák

3. előadás

42

Feladatok

Mit jelent a **ROM** rövidítés?

Jellemezze a **PROM**-ot!

Jellemezze az **EPROM**-ot!

Jellemezze az **EEPROM**-ot!

Milyen memória van a legtöbb fényképezőgépben?

Máté: Architektúrák

3. előadás

43

Az előadáshoz kapcsolódó**Fontosabb témák**

Programozható logikai tömbök

Aritmetikai áramkörök. Léptető, fél és teljes összeadó, ALU, az ALU-val végezhető műveletek, átvitel továbbterjesztő és kiválasztó összeadás

Nem kombinációs áramkörök.

Óra, tárolók, flip-flop-ok

Központi memória, bájtsorrend

Hamming távolság. Hibaészlelő, hibajavító kódok

Máté: Architektúrák

3. előadás

44