

Számítógép architektúrák

A memória

Tartalom

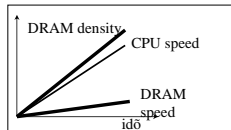
- Félvezető tárolók
- DRAM, SRAM
- ROM, PROM
- Töközások, memóriamodulok
- Lokális elve
- Gyorsítótárak (cache)

A memória

- Tár: programok és adatok tárolására. Címezhető cellák.
- Központi tár: (gyors) memóriabuszon v. a rendszerbuszon keresztül kapcsolódik a processzorhoz
- Memória a perifériavezérlőkön is! Ezek is címezhetőek! Néha címtartományuk egybeesik, néha nem.
- Később figyeljük meg a memória-hierarchiát!

A táruk implementációja

- Régebben ferritgyűrűs táruk: mágnesezhetőség-fluxusváltás elven. Nem felejtettek.
- Ma már magas-integráltságú félvezetős lapkák (millió számú tranzisztor[-kondenzátor] hálózata, tokban), memória modulra szerelve
 - ciklikus működés!
- Két trend:
 - kapacitás növelés,
 - elérési idő csökkentés.



© Vadász, 2005

Ea7 4

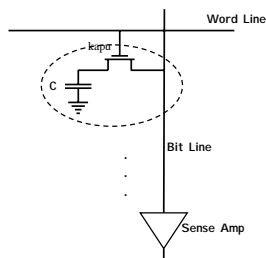
A félvezető tárolók

- **RAM: Random Access Memory**
 - random: egy cella elérése nem függ a többbitől, akár “véletlenszerűen” bármelyiket címezhetjük.
 - Sorokból és oszlopokból álló háló, elemei a cellák
- **DRAM: Dynamic RAM**
 - Egy cella egy tranzisztor-kapacitor pár, egy bithez.
 - Dinamikus: a kiolvasás-beírás, a “frissítés” is dinamikus, időt igénylő.
 - Írható-olvasható,
 - táp megszűnésével “felejt”.
 - MOS, CMOS, NMOS technológiák.

© Vadász, 2005

Ea7 5

Egy cella ...



- **Írás:**
 - Állíts a bit vonalra magas v. alacsony szintet (a beírni kívánt bit szerint);
 - Nyisd a kaput: megfelelően feltöltődik a C
- **Olvásás**
 - Állítsd a bit vonalat „fele” feszültségre;
 - Nyisd a kaput;
 - A C szintjétől függően a Sense Amp érzékel ...

© Vadász, 2005

Ea7 6

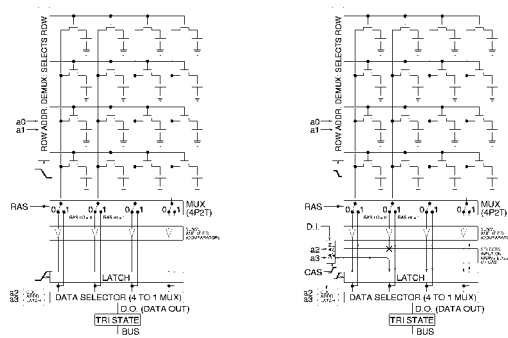
DRAM áramkörei ...

- Sorokból és oszlopokból álló háló, elemei a cellák
- A lapkában további speciális áramkörök segítenek
 - Cellák sor/oszlopainak kiválasztására (r/c address select/decoder, sor/oszlop cím bufferek)
 - A cellákból „kiolvasott” jelek tárolására (sens amplifiers: elérési tranzisztorok, output buffers)
 - Frissítési szekvenciák nyomonkövetésére (counter)
 - A cellák írására, töltésük „megemelésére” (write enable)
- „Kívülről” segíthet a memória controller (esetleg CPU)
 - Memória típus, sebesség, mennyiség azonosítás, hibakezelés

DRAM logikai felépítés



Egy 4 x 4-es mátrix

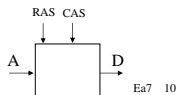


Olvasás

Írás

DRAM operációk

- **Tipikus memóriaelérés (olvasás):**
 - Sorcím a cím-lábakon → RAS jel leesik: sorcím rögzül a sor-cím-bufferben és az elérési tranzisztorok (sens amps) aktiválódnak;
 - a RAS jel stabilizálódásával a teljes sor celláinak értékét felveszik az elérési tranzisztorok;
 - Oszlopcím a cím-lábakon → CAS jel leesik: oszlop cím rögzül az oszlop-cím-bufferekben; a CAS stabilizálódásakor a kiválasztott rész töltődik az output bufferba.



© Vadász, 2005

Ea7 10

Technológiák

- **Fast Page Mode RAM** (egy sorcímzés mellett több oszlopcímzés), **Extended Data Out (EDO)**
- **Burst Extended Data Output (BEDO) RAM** (egy sorcímzés, egy oszlopcímzés mellett 4 adatcella) (dual bank, de az EDO „halála” miatt elavult)
- **SDRAM: Synchronous DRAM** (mehetünk a 66-100MHz fölé)
A mem „zárolja” a CPU-tól jövő cím, adat és kontrol infókat, autonóm módon dolgozik (a system clock kontrollja alatt), mialatt a CPU csinálhat más. Valamennyi idő után az outputon ott az eredmény...
- **ESDRAM (Enhanced ...):** a szokásos SDRAM lapkán kisebb SRAM cache is van (akár 200 MHz is)

© Vadász, 2005

Ea7 11

Roadmap ...

Bemutató év	Technológia	„Sebesség” határ	Max Bps
1987	FPM	50 ns	176 MBps
1995	EDO	50 ns	186 Mbps
1997	PC66 SDRAM	60-66-83 MHz	240 MBps
1998	PC100 SDRAM	100 MHz	400 MBps
1999	RDRAM	800 MHz	1,6 GBps
1999/2000	PC133 SDRAM	133 MHz	532 MBps
2000	DDR SDRAM	266 MHz	1064 MBps

© Vadász, 2005

Ea7 12

A piaci arányok ...

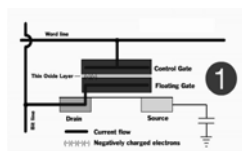
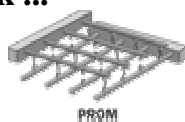
	SDRAM	DDR	RDRAM	EDO	DDR2
2002	55%	39%	5%	1%	
2003	13%	81%	3%		3%
2004	8%	83%	2%		9%
2005	5%	58%	2%		35%

Félvezető tárolók ...

- **ROM: Read Only Memory: csak olvasható.**
 - Ez is cellák hálózata, oszlop-sor tömbbe rendezve,
 - diódák a cellákban, összekötést adnak (1 bit), nincs összekötés (0 bit).
 - Kikapcsolva nem felejt a lapka.
 - Elérések itt is random jellegűek. (oszlop-sor kiválasztás).
 - Címtartományuk, címezhetőségük egybeeshet a DRAM-okkal: a (központi) memória részben ROM-okból, részben DRAM-ból állhat.
- **PROM, EPROM: Beégethető a tartalom, EPROM-nál törölhető és újraírható ...**

Félvezető tárolók ...

- **PROM (Programmable ROM)**
 - Sor-oszlop háló,
 - a cellákban „olvadó biztosíték” (fuse),
 - Beégethető a tartalom.
- **EPROM (Erasable PROM) törölhető és újraírható ...**
 - Cellákban két kapus tranzisztor
 - Floating gate
 - Control gate
 - Köztük oxidréteg
 - 1 bit: „összekötte” a bit- és word line ...
 - 0 bit: a kapu „zárva”



Féltrevezető tárolók ...

- **EEPROM (Electronical Erasable PROM)**
 - Mint az EPROM, de nem UV fény „töröl”.
 - Lassú, mert 1 bájtot lehet egyszerre törölni, újraírni ...
- **FLASH memory**
 - Ez valójában EEPROM, de egyszerre blokkot (512 bájt) lehet újraírni.
 - Már elég gyors.

Féltrevezető tárolók ...

- **SRAM: Static Random Access Memory**
 - ezek is írhatók, olvashatók,
 - random elérésűek,
 - kiolvasási idejük hallatlanul gyors (szinte 0),
 - de drágák és
 - energiaigényesek (ezért melegednek, hűtendőek!)
 - Gyorsító-tárakhoz (cache) használják.
 - Egy cellájuk 4-6 ternárisztorból álló flip-flop áramkör ...
nincs bennük kapacitor ... (olyanok, mint a CPU-k regiszterei)

Memória (összefoglalás)

- **RAM**
 - **DRAM (Töközött magas integráltságú áramköri lapkák)**
 - FPM, EDO, BEDO DRAM
 - SDRAM, ESDRAM
 - DDR, DDR2
 - RDRAM
 - **SRAM**
 - **ROM**
 - PROM
 - EPROM, EEPROM
- **FLASH**

A DRAM lapkák tokozása

Dual In-Line Package



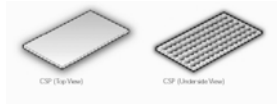
Small Outline J-lead



Thin Small Outline Package

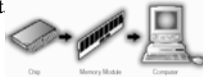


Chip Scale Package



Memória modulok

- Memória modulok hordozzák a lapkákat Szabványok. Az alaplap(ok) foglalatába helyezhetők.



- Single In-line MM (SIMM)

- 32 bites CPU-khoz
- 72 érintkező a modulon



- Dual In-line MM (DIMM)

- 168 láb, 64 bites CPU-khoz is

DDR400 (800 MHz-es
frontside bus chipset-
hez)

- Small Outline DIMM (SODIMM)

- 144 (72) érintkező, sokkal kisebb, notebook-okhoz

- Rambus (RIMM, SORIMM)

- 16 bites adatösvény, pipelining, gyors

Ismét a hozzáférési idő csökkentés

- Szélességnövelés, ciklusnövelés, pipelining, bursting
- Memória időzítés (amit lehet állítani)

2,5-2-2-5 (t_{CL} - t_{RCD} - t_{RP} - t_{RAS})

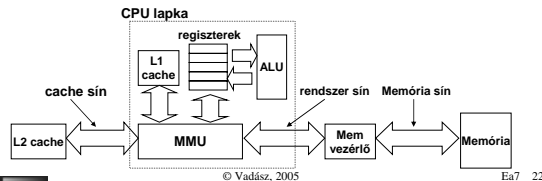
t_{CL}	CAS Latency: a CAS leesése utáni várakozás ideje, szokásosan 2, 2.5, 3 ciklus. Miután ez letelik, az adat a DQ lábakra került.
t_{RCD}	RAS to CAS Delay: RAS leesés utáni várakozás, amíg a CAS jelet lehet küldeni. Szokásosan 2, 3 ciklus
t_{RP}	RAS Precharge: Ezalatt az idő alatt a vezérlő ismét deaktiválja a sort.
t_{RAS}	Active to precharge delay: ezalatt a sor aktív kell legyen. Csak ezután lehet deaktiválni. 5 – 8 ciklus szokott lenni. Teljesülnie kell: $t_{RCD} + t_{CL} < t_{RAS}$

<http://www.tomshardware.hu/mainboard/04q1/040119/index.html>

- Gyorsítótárak (cache) alkalmazásával
- Asszociatív tárák alkalmazásával

A gyorsítótárak (caches)

- A programok lokalitása: tapasztalat
- Kisebb kapacitású, de a CPU-hoz “közelebbi” és gyorsabb (SRAM) memória, amiben
- a központi memória tartalom egy része szintén megvan.
- A CPU “egyszerre” címzi a cache-t is, a központi memóriát is:
 - ha találat van a cache-ben, csak onnan vesz!
 - gond lehet az adat-konzisztencia!



A lokalitás elve

Principle of Locality

- Processzek statisztikailag megfigyelhető tulajdonsága, hogy egy idő-intervallumban címtartományuk egy szűk részét használják ...
 - Időbeli lokalitás
 - Hivatkozott címeket újra ...
 - Térbeli lokalitás
 - Közeli címeket ...
- Az elv érvényesülése miatt van értelme kisebb, de gyorsabb átmeneti tárolók használatának ...
 - Gyorsítótár; munkakészlet; TLB; diszk buffer cache ...

A 80/20 szabály

Lokalitás elve (illusztráció)

- Processz futása során instrukciók sorozatát hajtja végre (instrukció hivatkozás sorozat),
- Az instrukciókban memória-hivatkozások lehetnek (adat hivatkozás)
- A hivatkozások sorozata a hivatkozási lánc (Reference String):
- $\omega = r_1, r_2, \dots, r_t, \dots, r_T$ # r_i : instr. vagy adat hiv.

for(i=1; i<=n; i++) a[i]=b[i]+c[i]; // n=1000 legyen
A gépi nyelvű program regiszteres gépen kb:

**Lokalitás elve
(illusztráció)**

Cím	Kód/adat	Megjegyzés	Cím	Kód/adat
4000	LOAD (R1), ONE	R1 index reg inicializálás	6000-6999	Tárolóhely A számára
4001	LOAD (R2), n	R2 határ reg inicializálás	7000-7999	Tárolóhely B számára
4002	COMP R1, R2	i > n tesztelés	8000-8999	Tárolóhely C számára
4003	JG 4009	Feltételes ugrás	9000	ONE
4004	LOAD (R3), B[R1]	B[i] betöltés R3-ba	9001	Tárolóhely n számára
4005	ADD (R3), C[R1]	Összeadás		
4006	STOR A[R1], (R3)	A[i]-be tárolás		
4007	ADD (R1), ONE	I inkrementáció		
4008	JUMP 4002	Ciklus újra		
4009	...	Ciklus után		

Ekkor a hivatkozási lánc a következő (összesen 9 instrukció-hív. ebből 7-re 1000-szer:
4000, 9000, 4001, 9001,
(4002, 4003, 4004, 7001, 4005, 8001, 4006, 6001, 4007, 4008)¹⁰⁰⁰
4002, 4003, 4009

© Vadász, 2005 Ea7 25

Lokalitás elve (illusztráció)

- Ugyanez virtuális memória menedzselésnél, lapozós rendszerben ...
- Legyen a lapméret 1000, a virtuális cím: v=(p, o)
(pl a[16] 6015 címe ekkor v=(6, 15))
- Ekkor a lapok hivatkozási lánc (összesen 5 lap):
4, 9, 4, 9,
(4, 4, 4, 7, 4, 8, 4, 6, 4, 4)¹⁰⁰⁰
4, 4, 4
- Kisméretű munkakészlet esetén sem valószínű a laphiba ...

© Vadász, 2005 Ea7 26

Gyorsítótárak szintjei

A gyorsítótár koncepció minden két szint között lehetséges, ahol a felsőbb szint gyorsabb, bár kisebb kapacitású.
Az alsóbb szinthez fordulás gyakorisága csökkenni fog.

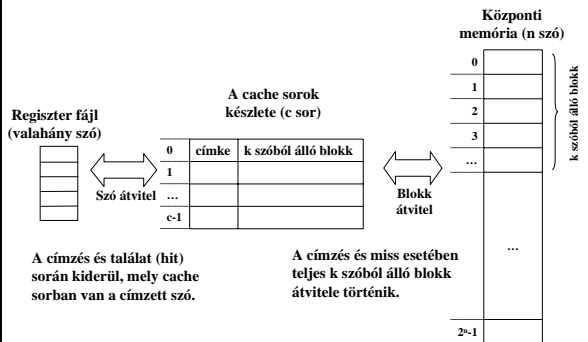
Level 1	CPU-n belül	~ 10 ns	4 – 16 KB
Level 2	SRAM cache sínen	~ 20 -30 ns	128 – 512 KB
Központi memória	DRAM memória sínen	~ 60 - 195 ns	128 – 512 MB
Diszkek	Mágneses+mech I/O sínen	~ 10 - 12 ms	32 – 90 GB
Internet	Hálózaton	~ 1 s -	határtalan

© Vadász, 2005 Ea7 27

A klasszikus gyorsítótár

- A CPU és a központi memória között
- Hardveres megoldás, még az OS számára is láthatatlan
- Mostanában két szintes (L1, és L2 szint), néha három
- Gyakori a szétválasztott adat- és instrukció gyorsítótár megoldás (Neumann elvet sért, de gyorsít!)
- A cache tervezés során ügyelni kell az adat konzisztenciára (ugyanaz a tartalom mind gyorsítótárban, mind a központi memóriában)
 - Címzés során mind a cache, mind a M címződik.
 - Ha találat van a cache-ben (cache hit). Szó transzfer.
 - ha nincs találat (cache miss). Cache miss penaly fogalom. Blokk transzfer.
 - Memória írás (store) probléma: konzisztencia biztosítás

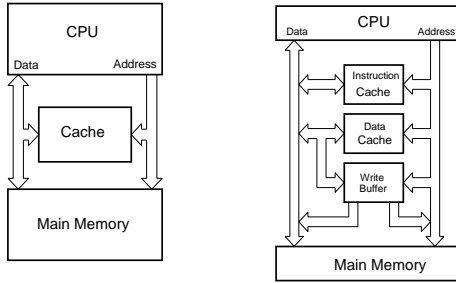
A regiszter, a cache és a M struktúra



A cache tervezés

- A cache mérete
- A blokk mérete
 - Újjonnan felhozott adat használatának valószínűsége
 - A cache-ből kicsorduló adat újrafelhasználásának valószínűsége
- Mapping: mely cache helyet foglaljon el egy-egy blokk
- Replacement algoritmus: mely blokk kerüljön ki a cache-ből, ha egy újnak kell a hely (LRU: Least Recently Used)
- Write Policy (store során)
 - Memóriába is írjunk, ha cache blokkba írtunk,
 - Csak akkor írjunk a memóriába, amikor a blokk helyére új kerül (replaced). Write Buffer-es cache.

Egyszerű és Write Bufferes Cache



Az asszociatív tár

- Tartalom szerint címezhető tár
- Translation Lookaside Buffer
- A CPU-hoz közeli tár, a memóriamenedzselést segíti
- Majd az OS tárgyban vesszük.

A memória hierarchia

