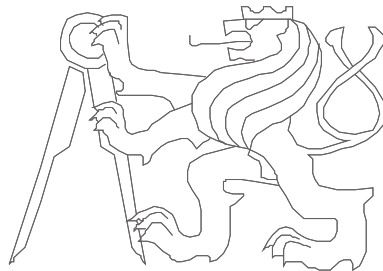


Architektury počítačů

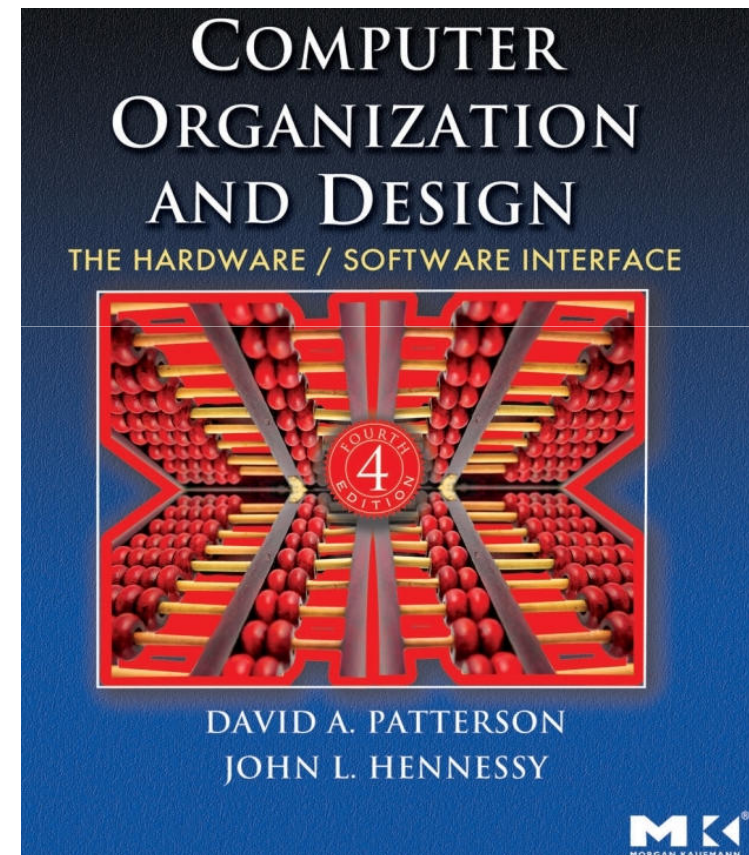
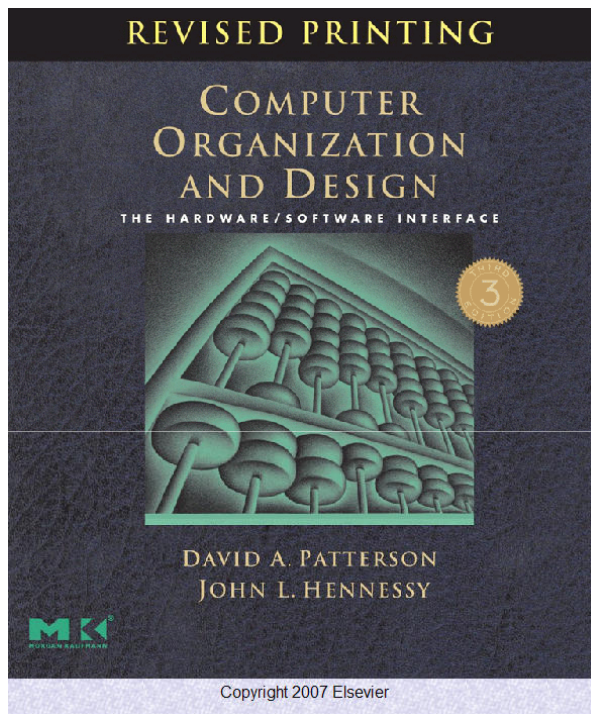
Paměť

Hlavní inspirace a zdroj: Patterson



České vysoké učení technické, Fakulta elektrotechnická

Literatura k předmětu



Osnova přednášky

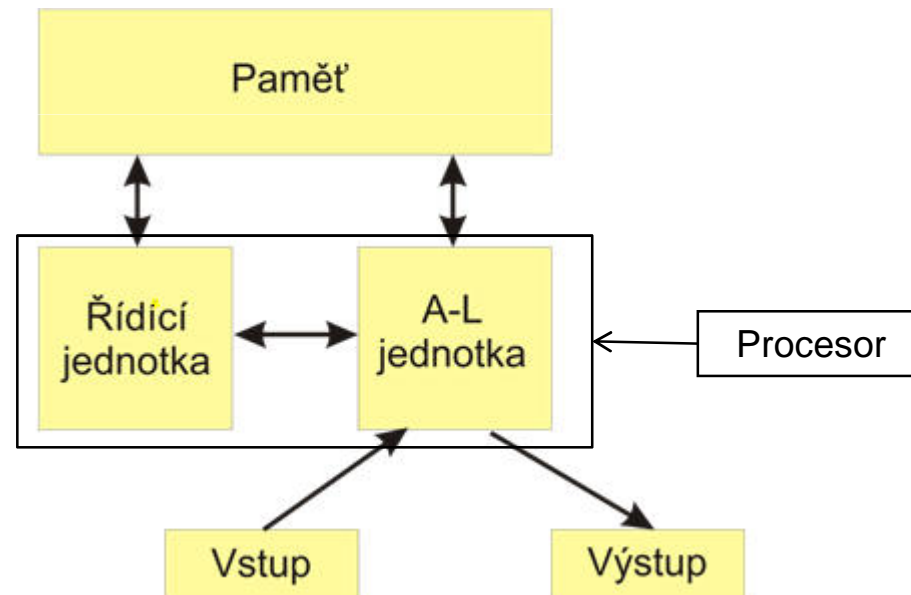
- Paměťová terminologie
- Hierarchie pamětí
- SP, resp. cache, keš
 - Více realistické SP
 - Víceúrovňové SP
- Virtuální paměť
- Problémy hierarchických pamětí
- Realizace pamětí - paměťové čipy
- Jiné principy vedlejších pamětí

John von Neumann, maďarský fyzik

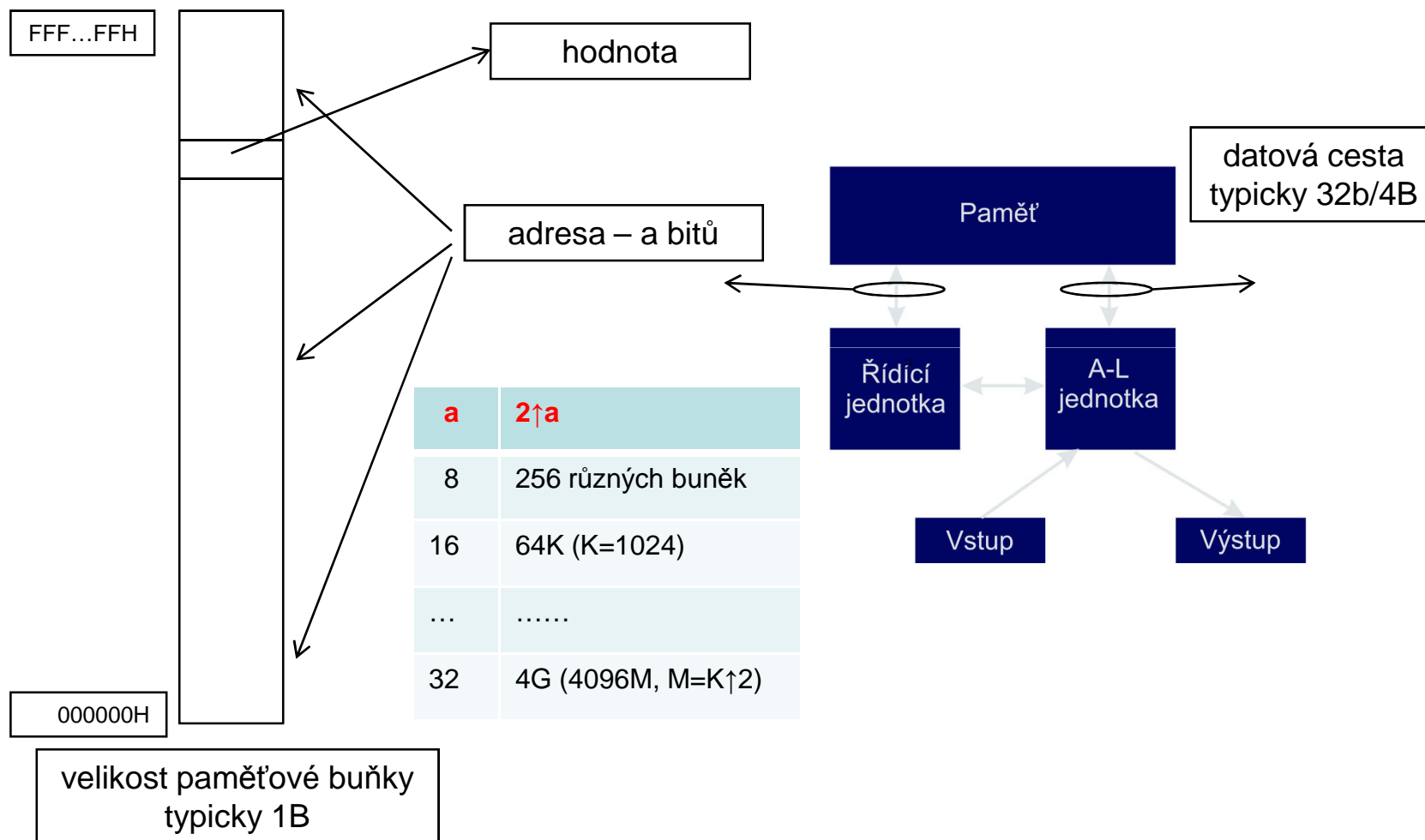
Architektura počítače podle JvN



28. 12. 1903 - 8. 2. 1957



Co je to adresní prostor? Adresa/hodnota



Terminologie kolem pamětí

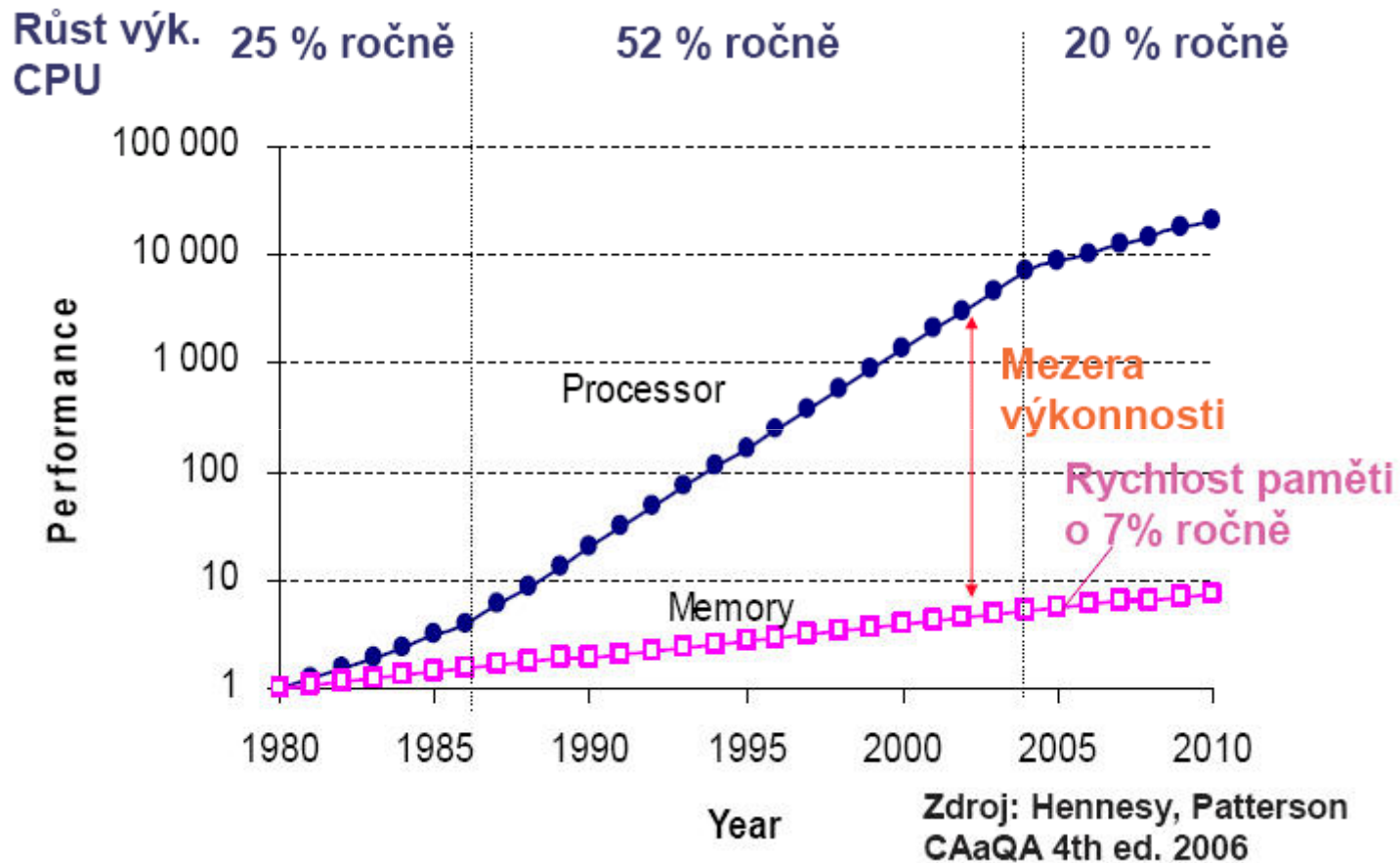
- Adresa, pojem snad není třeba vysvětlovat.
- Hodnota, vlastní informace. Paměťová buňka však může obsahovat i další informaci (třeba o platnosti hodnoty, apod.).
- **Parametry pamětí:**
 - Vybavovací doba paměti, kritický parametr. Délka časového intervalu mezi objevením se požadavku a okamžikem, kdy jsou data k dispozici.
 - Doba přístupu, zastaralý parametr; vybavovací doba + obnovení obsahu po destruktivním čtení. Význam měl u starých pamětí.
 - Propustnost, výkonový parametr. Schopnost zpracovat uvedené množství za jednotku času.
 - Latence = zpoždění, podobně jako vybavovací doba.

Terminologie kolem pamětí

- Typy pamětí RWM (RAM), ROM, FLASH,
- Provedení pamětí SRAM, DRAM.

- Občerstvování pamětí (refresh). Nezbytná pracovní fáze dynamické paměti. Negativně ovlivňuje (prodlužuje) průměrnou vybavovací dobu.

Disproporce ve výkonu proc x pam, Moorův zákon



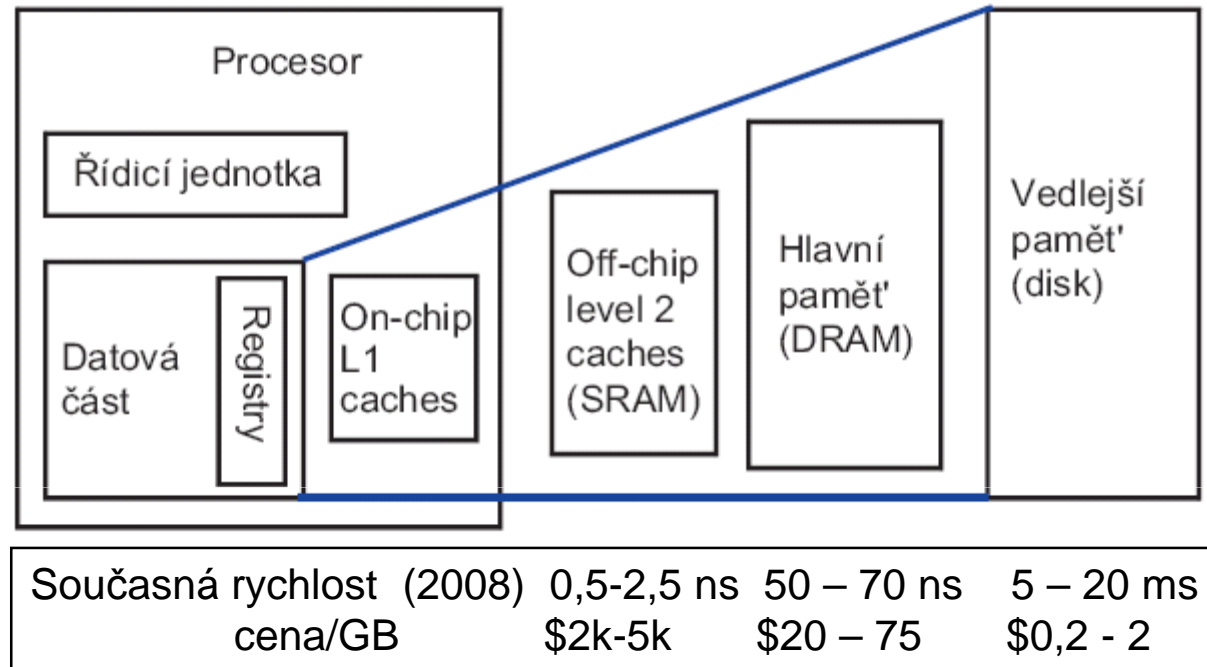
Paměťová hierarchie – základní principy

- Programy/procesy přistupují v daném okamžiku jen k malé části svého adresového prostoru
- Časová lokalita
 - Položky, ke kterým se přistupovalo nedávno, budou zapotřebí brzy znovu.
 - Příklad: programová smyčka, proměnné instrukcí.
- Prostorová lokalita
 - Položky poblíž právě používaným budou brzy zapotřebí také.
 - Příklad: sekvenční přístup ke kódu (paměť programu), datová pole (paměť dat).

Co z uvedeného plyne?

- Je výhodné uspořádat paměťový prostor hierarchicky – paměťová hierarchie.
- Všechny potřebné informace uchovávejte v sekundární paměti.
- Položky nedávno používané a blízké zkopírujte do (menší) paměti implementované DRAM.
 - Operační paměť.
- Položky ještě častěji používané (i ty jim blízké) zkopírujte do menší a rychlejší SRAM.
 - Skrytá paměť.

Paměťová hierarchie



- To se jedna a tatáž informace může objevit na více místech hierarchické paměti? Ano.

Jak a kde pak ale hledanou informaci najdeme?

- Podle adresy a případně dalších informací (např. o platnosti).
- Hledat začneme v paměti nejvyšší hierarchické úrovni (nejblíže procesoru).
- Požadavky:
 - Paměťová konzistence.
- Prostředky:
 - Virtualizace adresy,
 - Mechanizmy uvolňování místa a migrace informace mezi paměťovými úrovněmi.
 - Hit, miss.

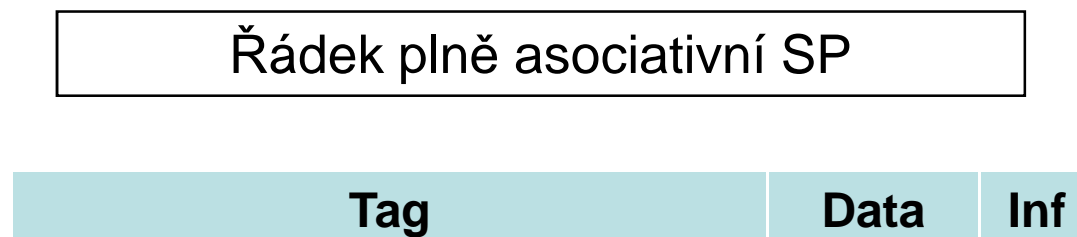
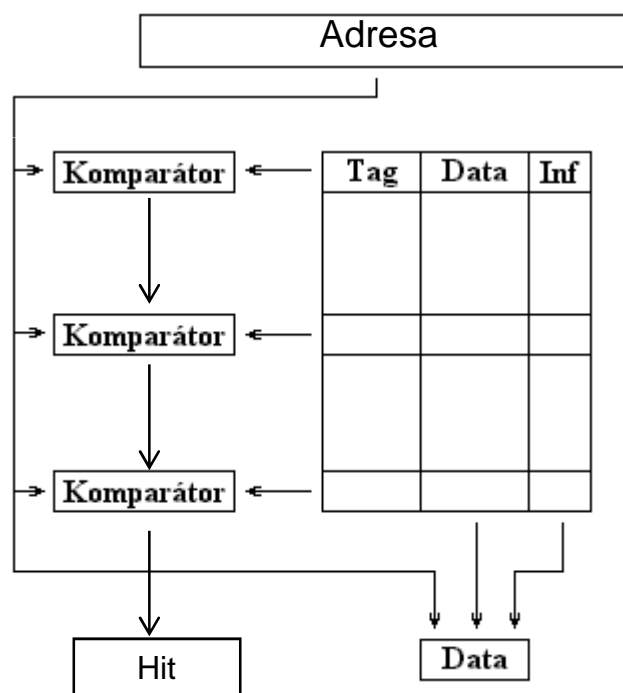
Řešení disproporce rychlosti? Skrytá paměť, SP.

Cache, česky keš

- Nebo-li **skrytá paměť**, SP,
- je označení pro vyrovnávací paměť používanou ve výpočetní technice.
- Zařazujeme ji mezi dva subsystémy s různou rychlostí. Vyrovnává se jí rychlost přístupu k informacím.
- Účelem skryté paměti je urychlit přístup k často používaným datům na „pomalých“ médiích jejich překopírováním na média rychlá.

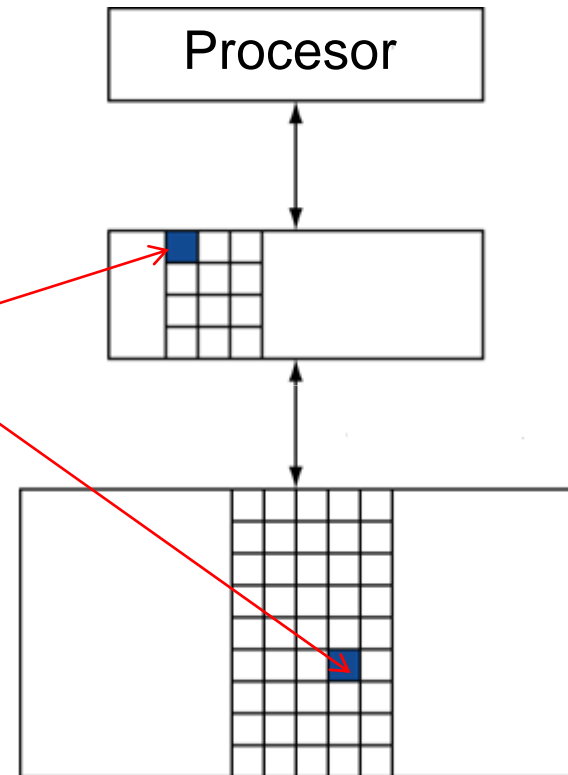
První přiblížení SP – **plně asociativní paměť**

- **Tag** - klíč pro vyhledávání informace ve SP. V tomto případě její adresa v paměti. Šířka pole odpovídá šířce adresy, typicky 4B.
- **Data** – ve SP uložená informace. Šířka pole je typicky 1B.
- **Inf** – maximálně několikabitové pole pro případnou další info o datech.



Terminologie kolem skryté paměti

- **Cache hit** pojmenování situace, kdy požadovaná hodnota ve skryté paměti (cache) **je**.
- **Cache miss**, opak. **Není** tam.
- **Cache line** nebo **Cache block** – základní kopírovatelná jednotka mezi hierarchickými úrovněmi.
- V praxi se velikost Cache Line pohybuje od 8B do 1KB.



Terminologie kolem skryté paměti II.

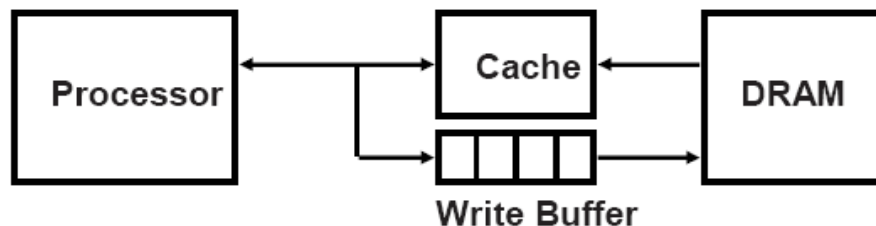
- **Hit Rate** - podíl počtu paměťových přístupů, které byly úspěšné (nalezla své údaje) při přístupu do té- které úrovně paměťové hierarchie.
- **Miss Rate** – podobně pro neúspěšný přístup.
- **Miss Penalty** – čas potřebný pro načtení bloku (údajů) z paměti nižší hierarchické úrovně.

Diskuze SP

- Šířka pole Tag odpovídá šířce adresy,
 - Každý řádek SP obsahuje tolik jednobitových komparátorů, kolik je šířka adresy,
 - Počet řádků SP určuje její kapacitu,
 - SP musí mít strategii uvolňování obsahu (migrace dat mezi hierarchickými úrovněmi) v případě vyčerpání její kapacity.
-
- Takováto SP je ale velmi drahá.
 - Proto existují a používají se SP
 - Přímě mapované,
 - S omezeným stupněm asociativity.

Řešení situace **Zápis dat** procesorem do paměti

- Na cestě je i SP!
- **Konzistence dat** – samozřejmý požadavek na shodu obsahu stejných adres na různých médiích.
- **Write through** – současně se zápisem do SP se data zapíše do zápisové fronty a pak asynchronně do paměti.
- **Write back** – data se do SP zapíše s poznámkou Dirty (D bit Inf pole). Ke skutečnému zápisu dat do hlavní paměti dojde až v okamžiku případného rušení příslušného řádku SP, kdy hrozí jejich ztráta.
- **Dirty bit** – rozšiřující pole v obsahu paměti. Indikuje, že v SP (cache) je **jiná hodnota**, než v paměti hlavní.

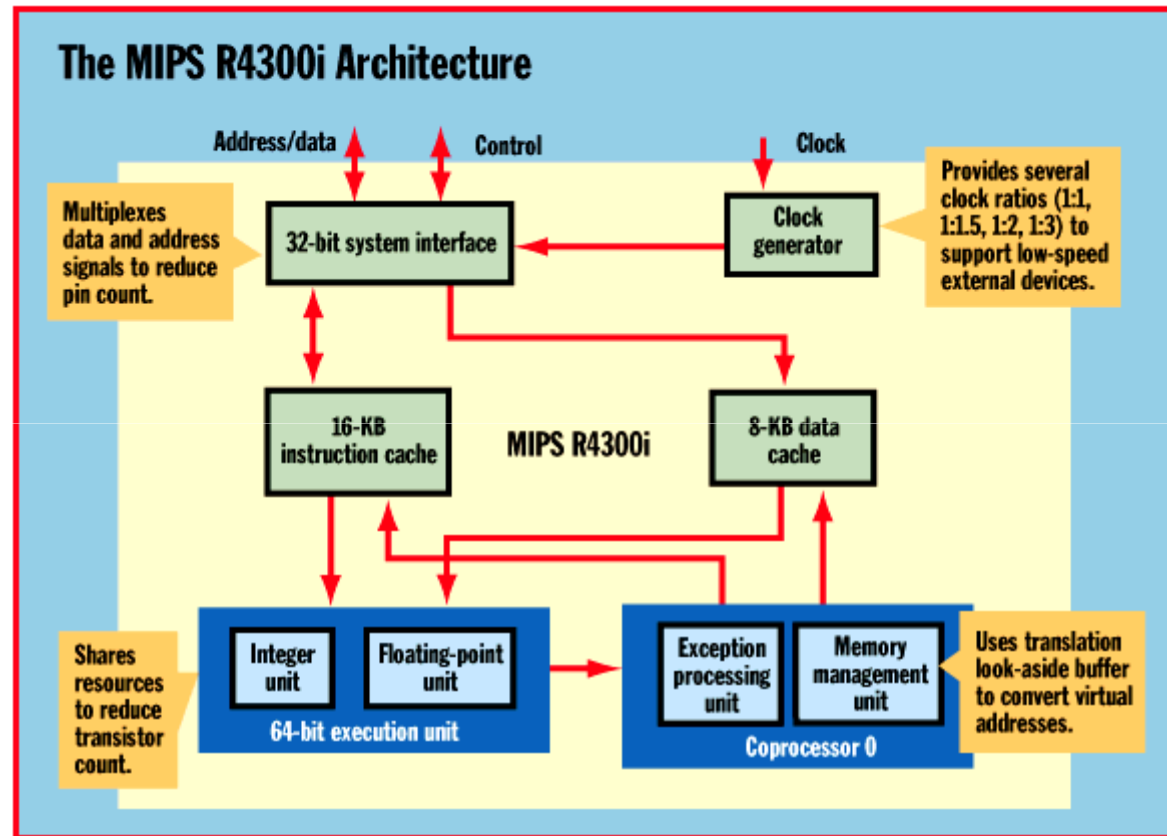


V	Další bity, např. D	Tag	Data
---	---------------------	-----	------

Řešení situace **Cache Miss**, data v SP nejsou

- Data se nejprve musí z hlavní paměti přečíst. Jenže: co když je SP plná?
- Strategie uvolňování bloků/řádek SP
- **Náhodná** (Random) – vybere se libovolný blok. Snadné, ale hloupé.
- **LRU** (Least Recently Used) musíme znát informace o posledním použití tohoto bloku (jedná se o celé číslo).
- **LFU** (Least Frequently Used), ke každému bloku si pamatujeme informace o tom, jak často byl blok požadován.
- **ARC** (Adaptive Replacement Cache), ve které se vhodným způsobem kombinuje strategie LRU a LFU.
- **Write-back**. Zároveň musíme obsah uvolňovaných řádek SP do hlavní paměti zapsat (D bity označené řádky). Zajištěno automaticky.

Příklad procesoru včetně SP – harvardská architektura



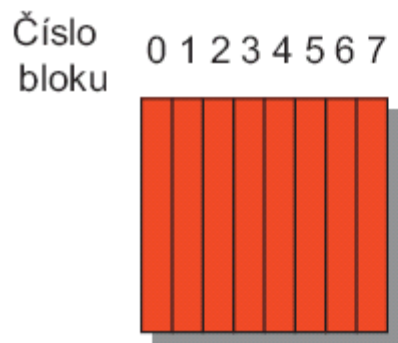
Oddělené SP dat a instrukcí

Více realistické SP

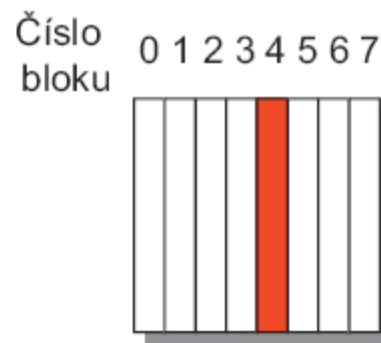
Příklad podle R.Lórenc, X36APS

- Mějme 8-mi blokovou skrytou paměť. Kam se do ní umístí blok 12 v případě
 - Plně asociativní.
 - Přímě mapované, nebo
 - S omezeným stupněm asociativity $N=2$ (2-cestná, 2-way cache).

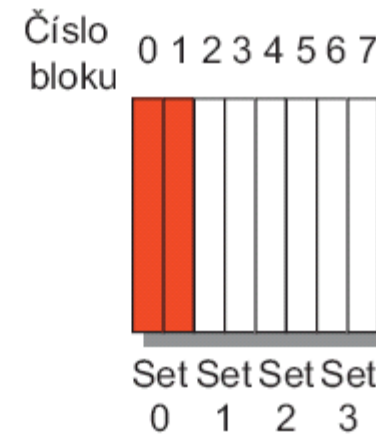
Plně asociativní:
blok 12 umístěn
kdekoliv



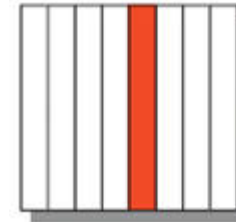
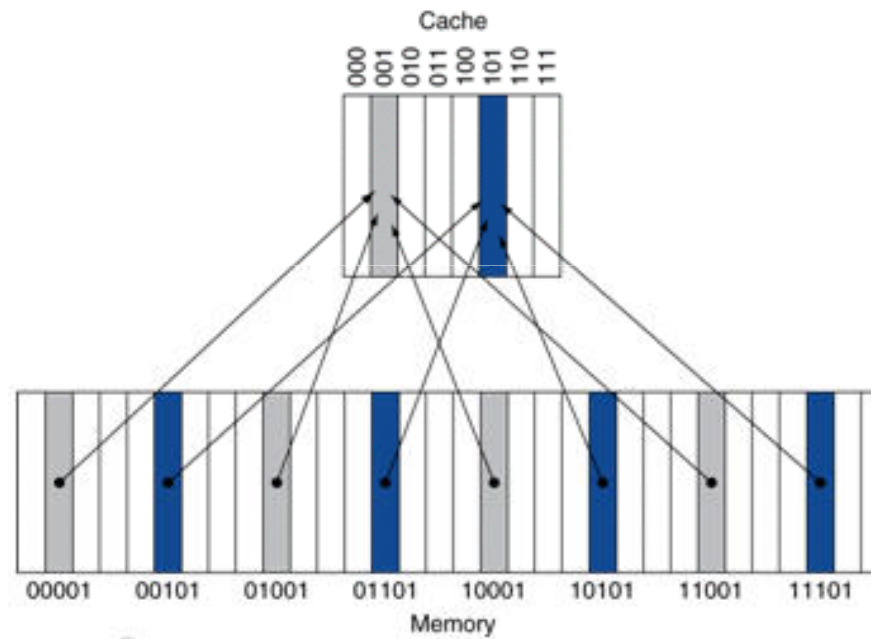
Přímě mapovaná:
blok 12 umístěn jen
do bloku 4 (12 mod 8)



Omezený stupeň:
blok 12 umístěn
do setu 0 (12 mod 4)



Přímo mapovaná SP

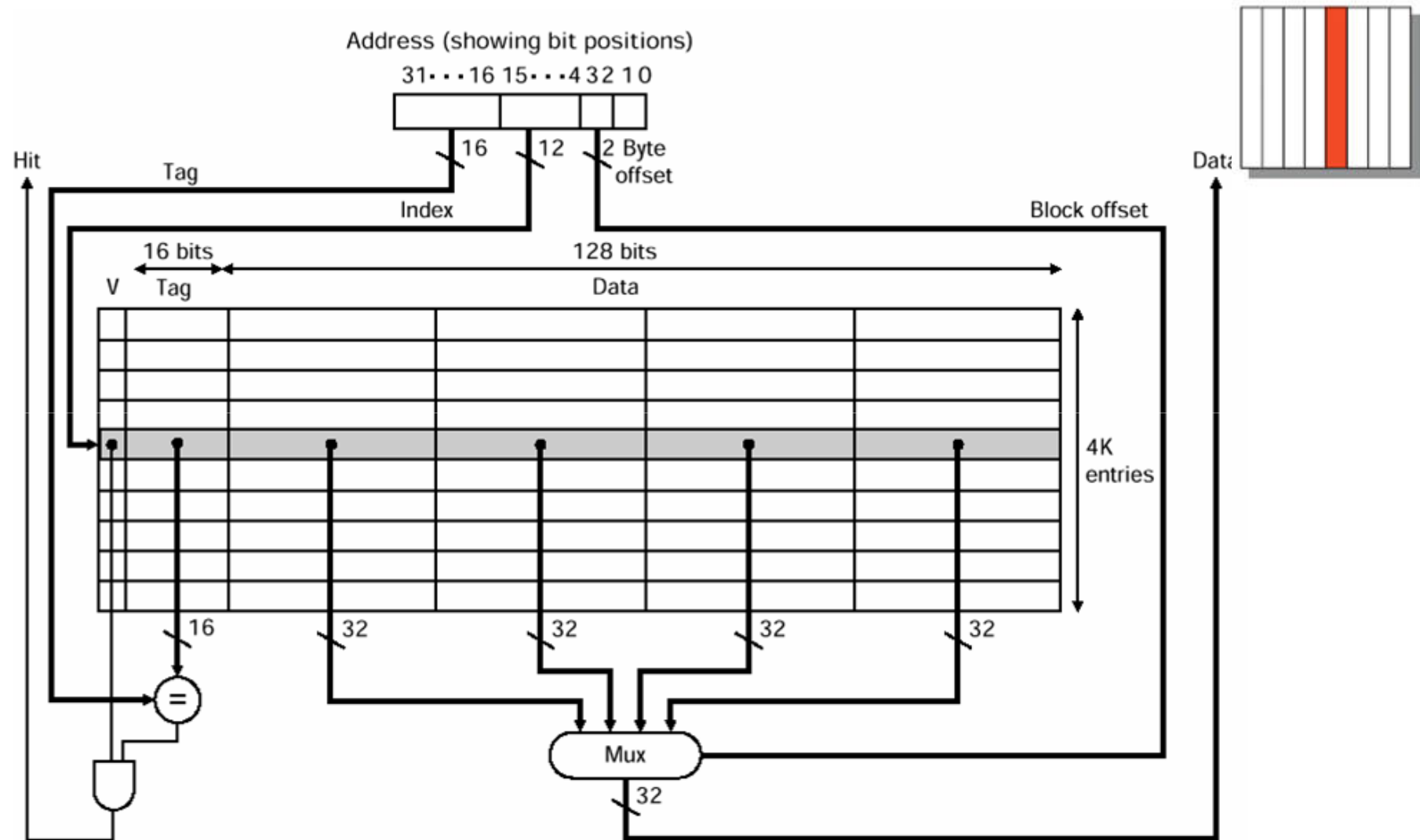


Realističtější formát řádky SP

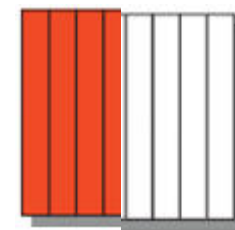
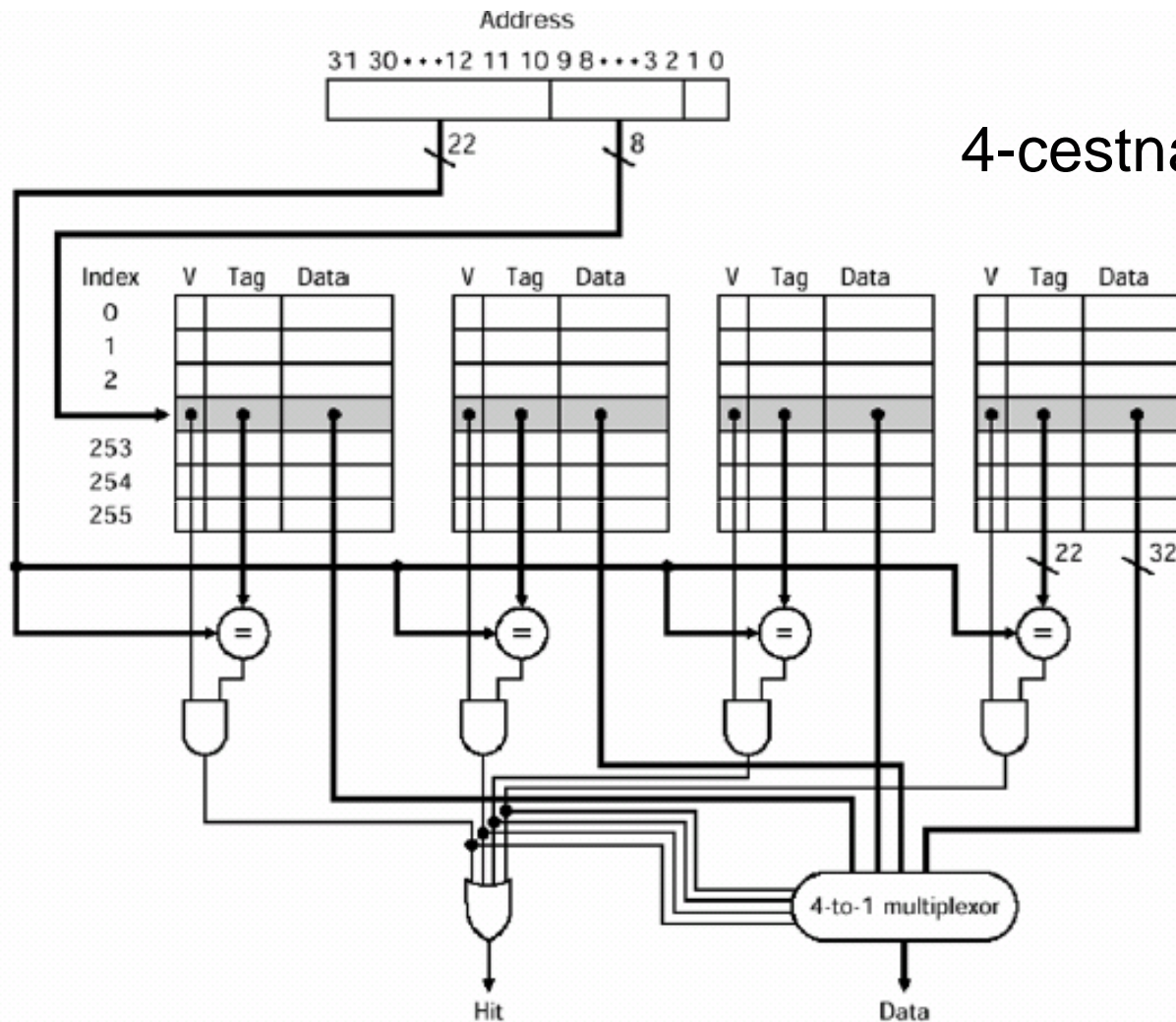
- **Tag** je index odpovídajícího bloku v operační paměti (v podstatě se jedná o hodnotu ukazatele/adresy dělenou délkou bloku).
- **Data** pole obsahující vlastní hodnoty na příslušné/ných adrese/ách.
- **Validity bit** – bit platnosti. Indikuje, zda je obsah pole Data vůbec platný.
- **Dirty bit** – rozšiřující pole v obsahu paměti. Indikuje, že v SP (cache) je **jiná hodnota**, než v paměti hlavní.

V	Další bity, např. D	Tag	Data
---	---------------------	-----	------

Přímo mapovaná skrytá paměť - implementace podrobněji



SP s omezeným stupněm asociativity N=4 podrobněji



Víceúrovňové SP

Trend - Víceúrovňové SP

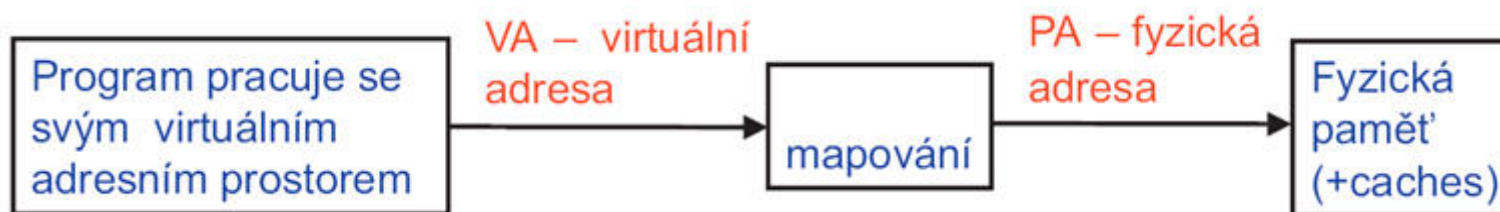
- Primární SP je bezprostředně připojena k procesoru
 - Rychlá, malá. Nejdůležitější: minimální Hit Time
- L2 SP ošetřuje výpadky primární SP
 - Větší, pomalejší, ale stále rychlejší než hlavní paměť.
Nejdůležitější: low Miss Rate
- Hlavní paměť ošetřuje výpadky L2
- Současné nejvýkonnější systémy mají i L3

	Typicky pro L1	Typicky pro L2
Počet bloků	250-2000	15 000-250 000
KB	16-64	2 000-3 000
Velikost bloku v B	16-64	64-128
Miss penalty (v hod)	10-25	100-1 000
Miss rates	2-5%	0,1-2%

Virtuální paměť

Virtualizace paměti

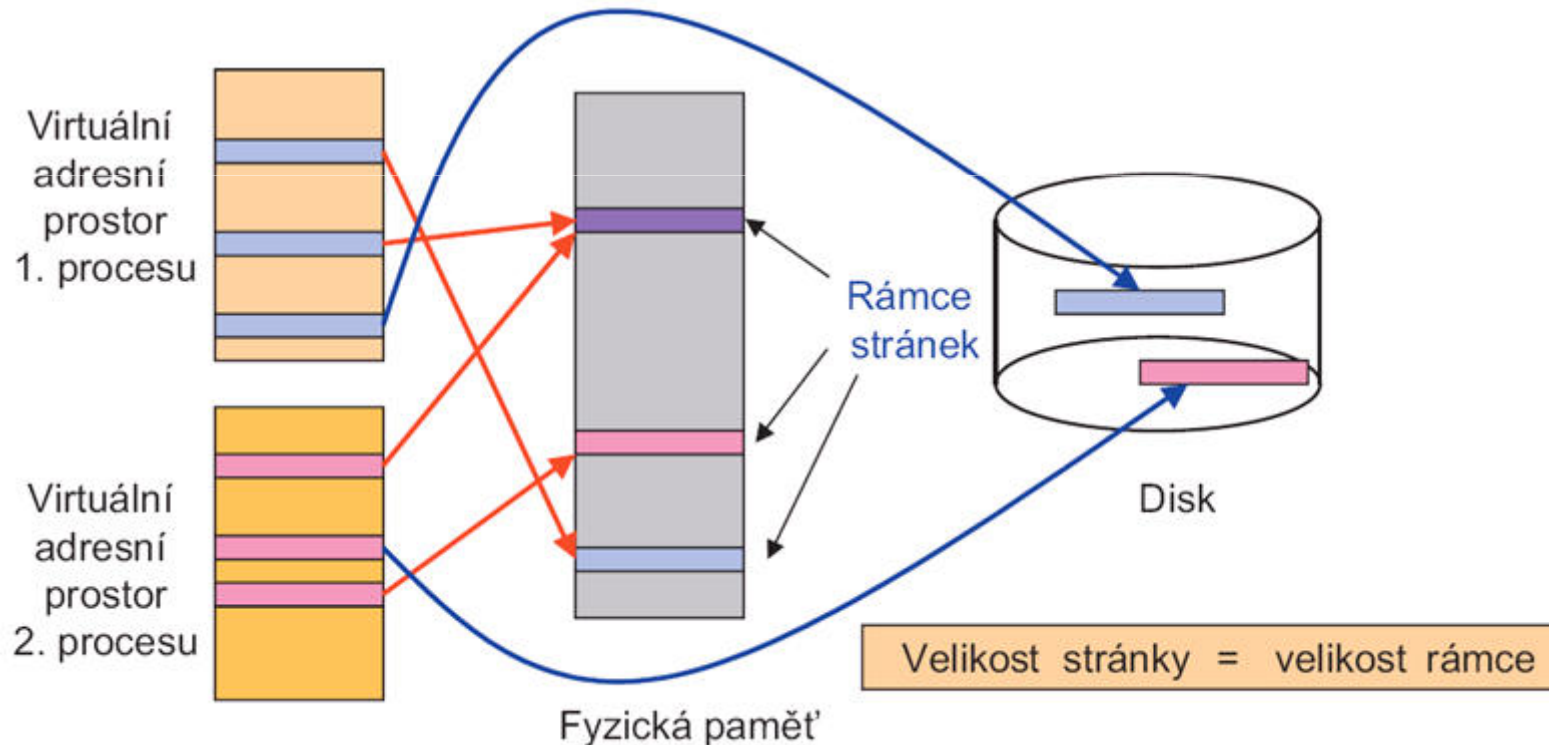
- **VP** je způsob správy operační paměti umožňující běžícímu procesu zpřístupnění paměťového prostoru, který je uspořádán jinak, nebo je dokonce větší, než je fyzicky připojená operační paměť.
- Převod mezi virtuální **VA** a fyzickou **PA** adresou může podporovat procesor (HW mapováním TLB, viz dále).
- V současně běžných operačních systémech je virtuální paměť implementována pomocí stránkování paměti spolu se stránkováním na disk, které rozšiřuje operační paměť o prostor na disku.



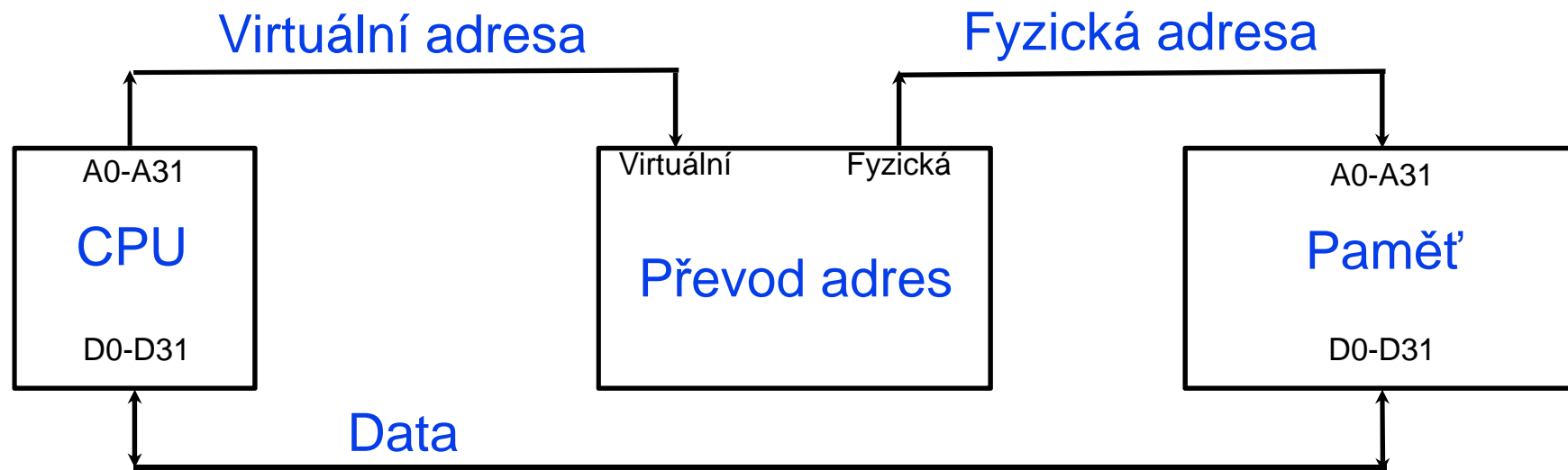
* R. Lórenc, X36APS, 2005

Virtuální paměť - stránkování

- Virtuální prostor tvoří stejně velké stránky (pages), které se přiřazují jednotlivým běžícím procesům.
- Fyzickou paměť tvoří stejně velké rámce (frames).
- Zde jen poznámka: moderní přístupy nevyžadují stejně velké stránky.



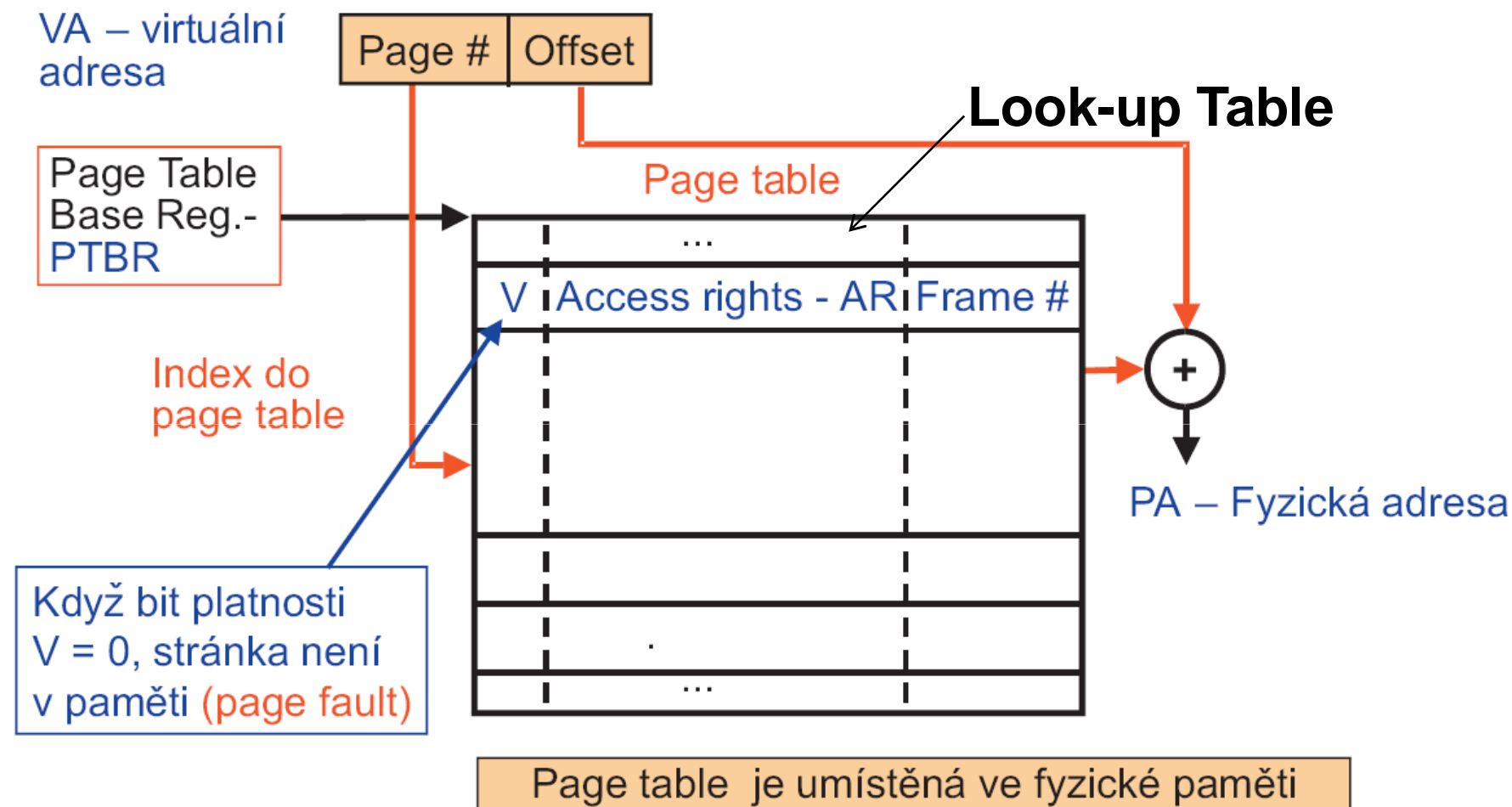
Virtuální a fyzické adresování



Realizace převodu adres?

- Tabulka stránek, Page Table.
- Jednotkou mapování jsou stránky,
- Stránka je také jednotkou přenosu mezi vedlejší a hlavní paměti.
- Mapovací funkce se nejčastěji implementuje Look-up Table (vyhledávací tabulkou).
- Příklad:

Tabulka stránek – význam položek

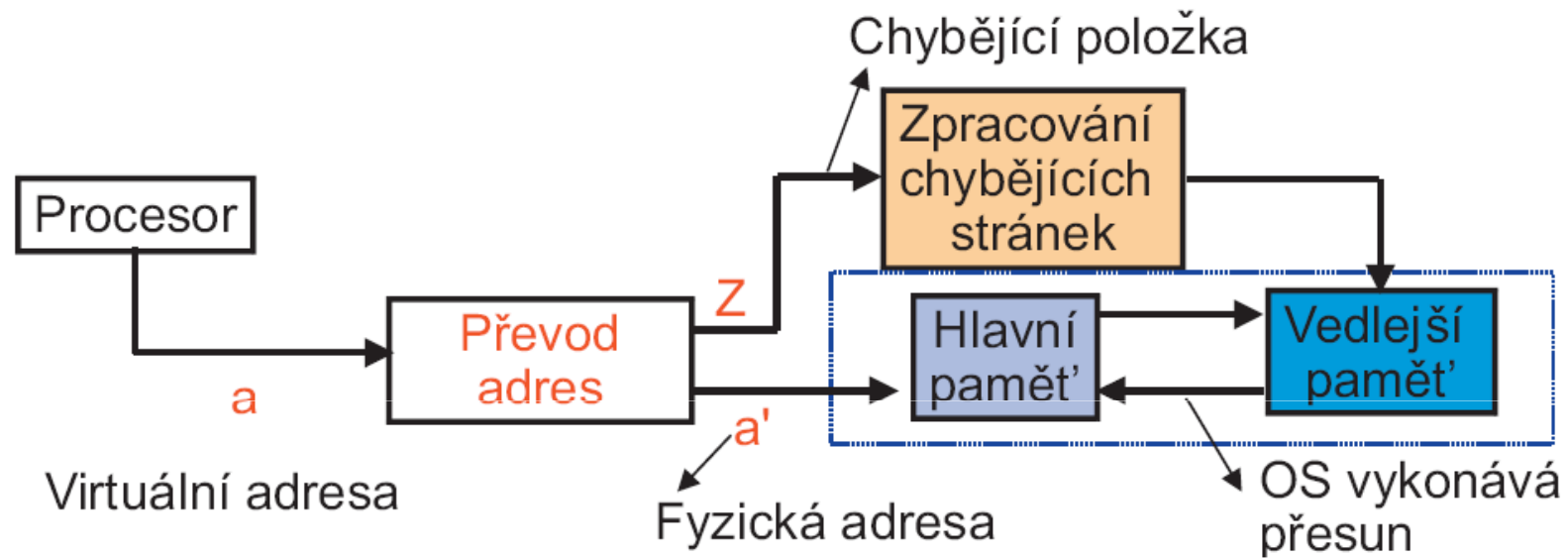


Poznámky

- Každý proces má svou Tabulku stránek,
- Tedy i svou hodnotu PTBR (bázového registru).
- To, mimochodem, zajišťuje paměťovou bezpečnost procesů.
- Formát položky Tabulky stránek
 - V – Validity Bit. V=0 Stránka není platná.
 - AR – Access Rights. Přístupová práva (Read Only, Read/Write, Executable, apod.),
 - Frame# - číslo rámce,
 - Popřípadě další, např. Modified/Dirty, apod. (budeme dále podle potřeby doplňovat).



Virtuální paměť: spolupráce HW a SW



Co dělat, když je výpadek stránky – Page Fault?

- **Fyzická paměť je volná, ale**
 - Rámec je prázdný, data jsou ve vedlejší paměti (na disku).
 - Požadovaná stránka se „nějak“ (DMA, Direct Memory Access, přímým přístupem do paměti, ale to zde neřešíme) načítá do prázdného rámce. Přepne se na případně čekající proces, který může probíhat.
 - Po dokončení DMA přenosu se vyvolá přerušení, aktualizuje se Tabulka stránek procesu.
 - Přepne se zpět na původní proces.
- **Paměti je nedostatek**
 - Pomocí LRU najdeme rámec, který můžeme uvolnit.
 - Má-li nastaven Dirty bit, zapíšeme stránku do vedlejší paměti (na disk).
 - Aktualizuje se Tabulka stránek procesu.

Pro vaší představu: typické hodnoty

	Typicky pro stránkované paměti	Typicky pro TLB
Velikost v blocích	16 000-250 000	40-1024
Velikost	500-1 000 MB	0,25-16 KB
Velikost bloku v B	4 000-64 000	4-32
Miss penalty (v hod)	10 000 000 – 100 000 000	10-1 000
Miss rates	0,00001-0,0001%	0,01-2

Problémy hierarchických pamětí

Některé problémy hierarchických pamětí?

- Koherence pamětí. Definice viz další slajd
- Jednoprocesorové (jednojádrové) stroje.
 - Řeší D-bit a migrační strategie Write-back.
- Multiprocesory se společnou i sdílenou pamětí – řešení je mnohem složitější. Používá se mj.
 - Společná sběrnice: Snooping (s odposlechem, slíděním), MESI protokol,
 - Broadcast (s rozesíláním),
 - Directories (adresáře).
- Je obsahem předmětu A4M36PAP.

Definice koherence

- Řekneme že multiprocessorový **paměťový systém je koherentní** jestliže výsledek jakéhokoli provádění programu je takový, že pro každé paměťové místo je možné sestavit myšlené sériové pořadí čtení a zápisů k tomuto paměťovému místu a platí
 - 1. Paměťové operace k danému paměťovému místu pro každý proces jsou provedeny v pořadí, ve kterém byly spuštěny tímto procesem.
 - 2. Hodnoty vracené každou operací čtení jsou hodnotami naposledy provedené operace zápis do daného paměťového místa vzhledem k sériovému pořadí.

Srovnání VPxSP, nenechte se zmást

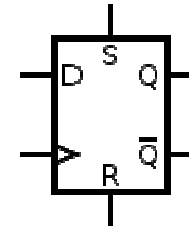
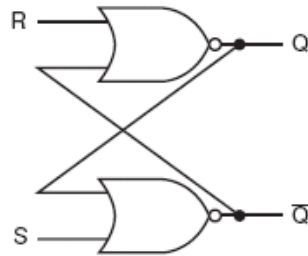
Virtuální paměť	Skrytá paměť
Stránka	Blok/řádek
Page Fault	Read/Write Miss
Velikost stránky: 512 B – 8 KB	Velikost bloku: 8 – 128 B
Asociativní	DM, N-cestná, asociativní
Výběr oběti: LRU	LRU/Náhodný
Write Back	Write Thru/Write Back

- Pozn.: TLB virtuální paměti může být plně asociativní, ale pro větší TLB typicky bývá jen 4-cestná.
- Rozumíte pojmům?
 - Co je oběť
- Závěr: každé adjektivum **V/SP** vyjadřuje něco jiného...

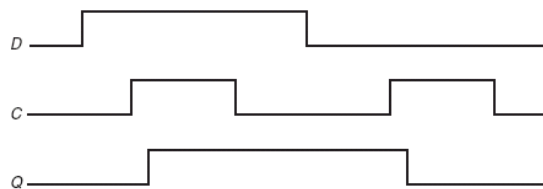
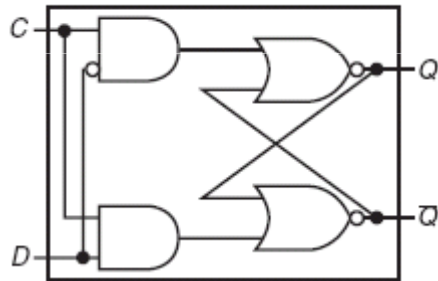
Realizace pamětí - paměťové čipy

Připomenutí začneme od lesa: klopné obvody

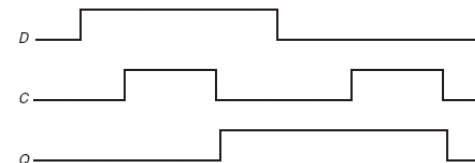
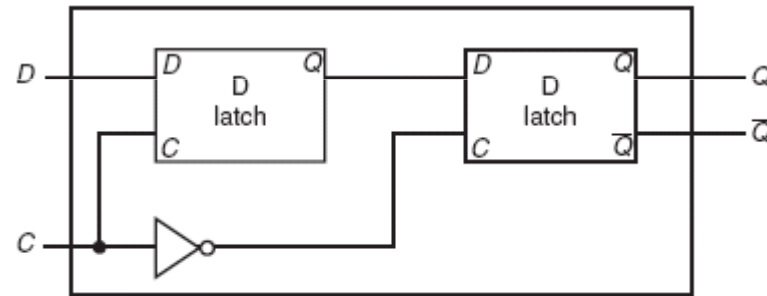
RS



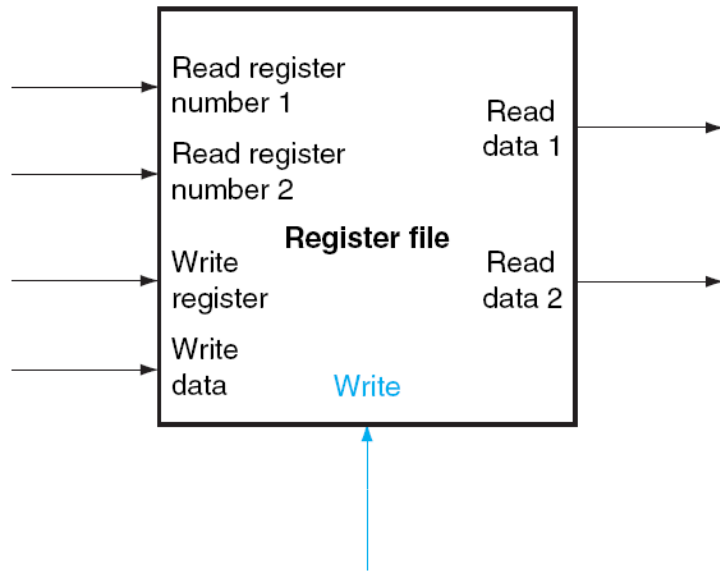
D latch, úrovnňový klopňý obvod



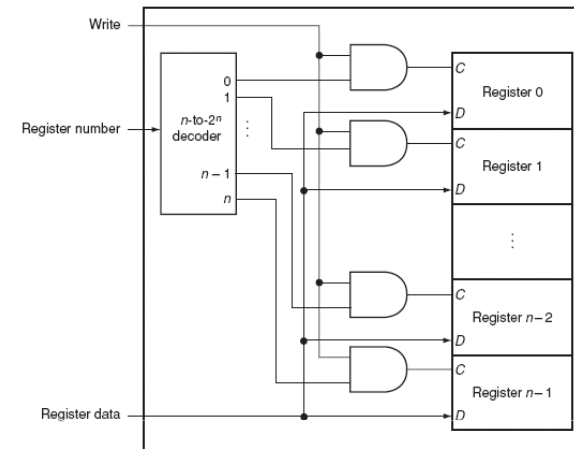
D flip-flop, hranový klopňý obvod



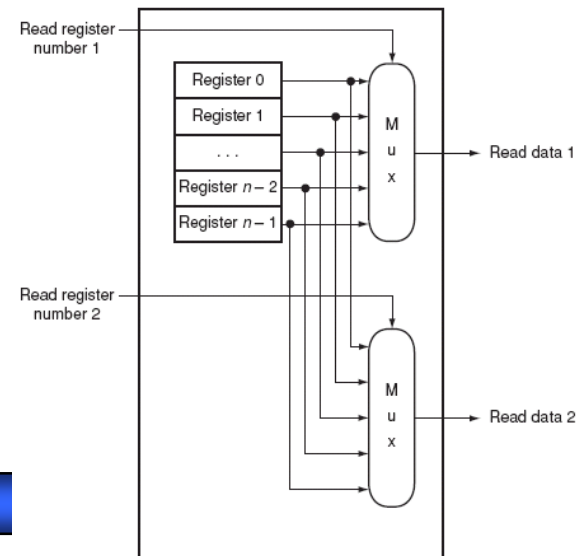
Příklad obvodu s dvěma sadami registrů



Zápis do registrů

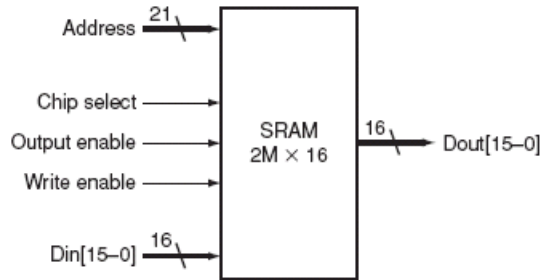


Čtení registrů

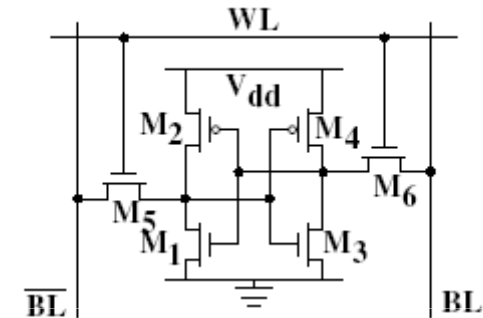


Typický čip a buňka SRAM

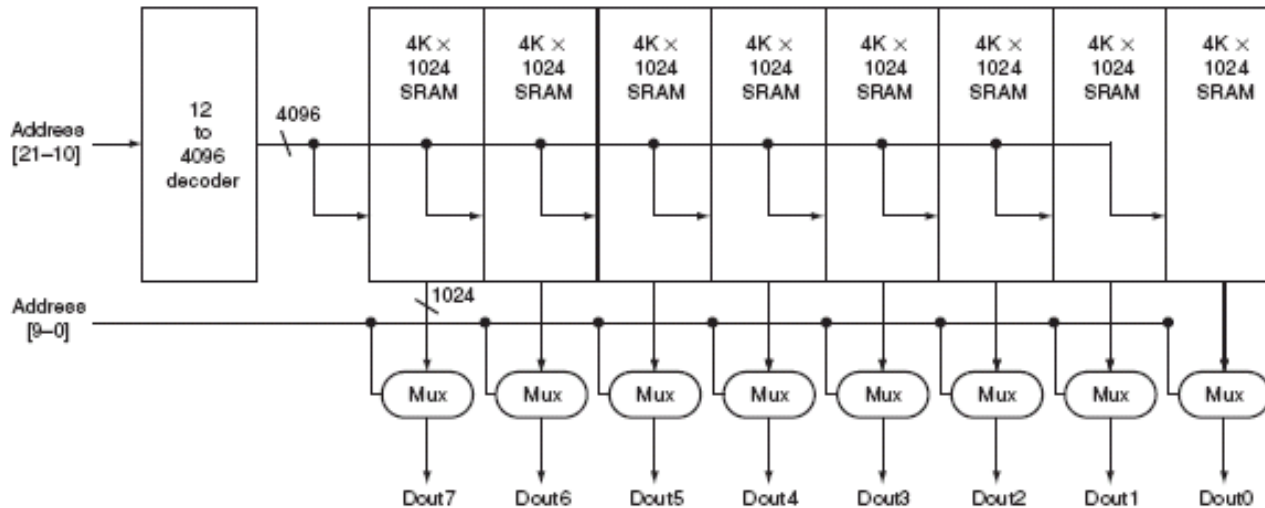
Typický SRAM čip



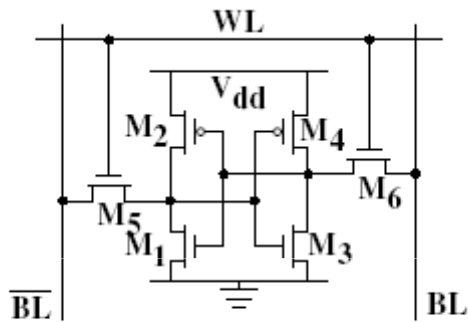
SRAM paměťová buňka
technologie CMOS



Větší paměť?

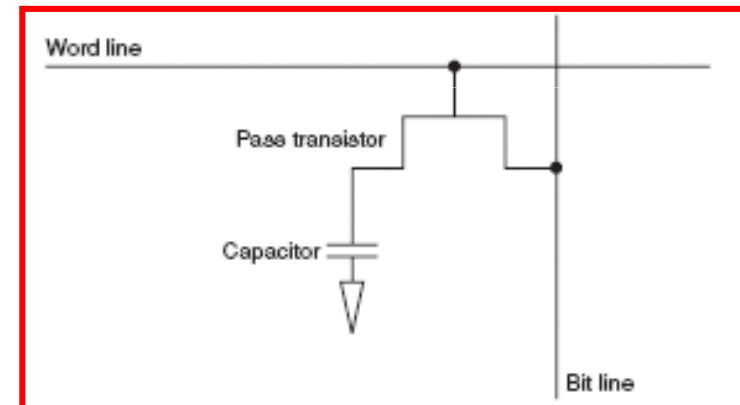


Detail paměťové buňky statické a dynamické paměti

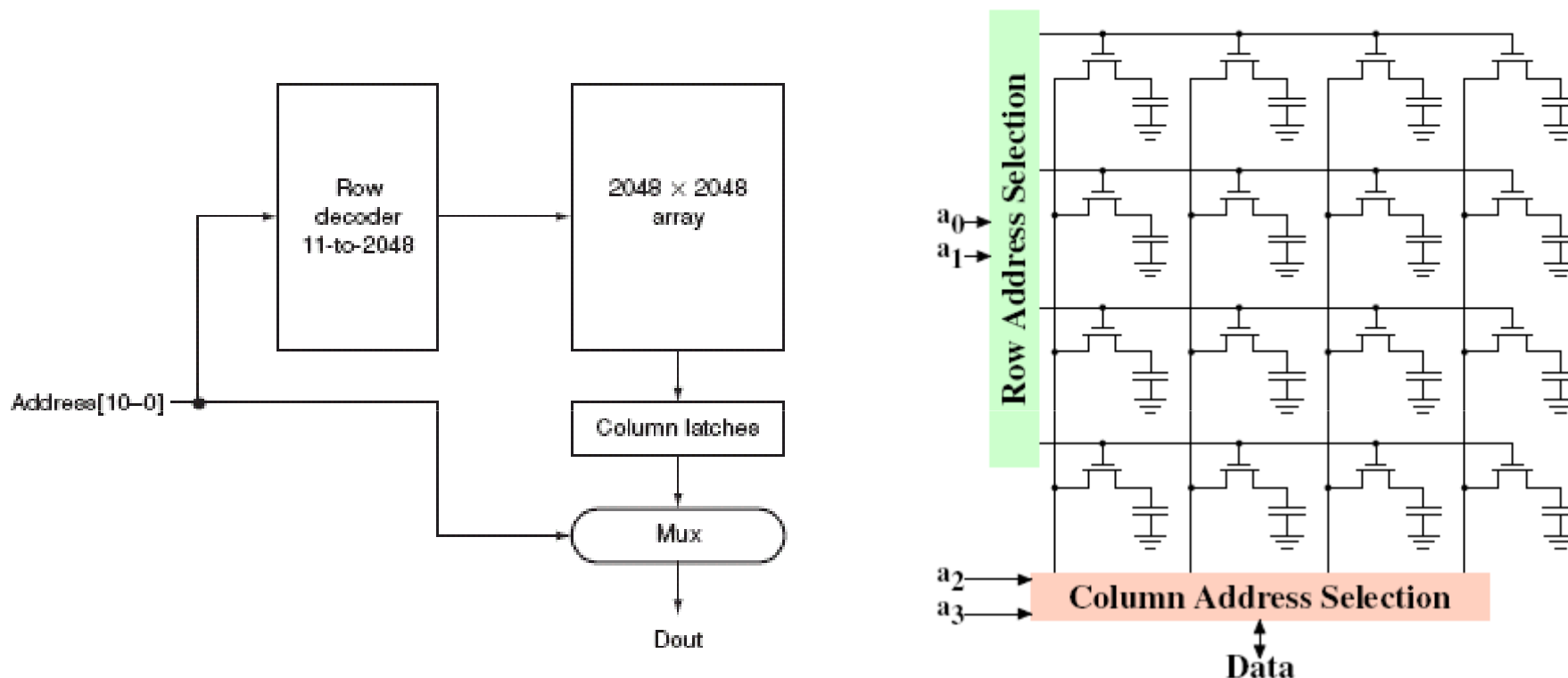


6-ti tranzistorová statická paměťová buňka

Jednotranzistorová dynamická paměťová buňka



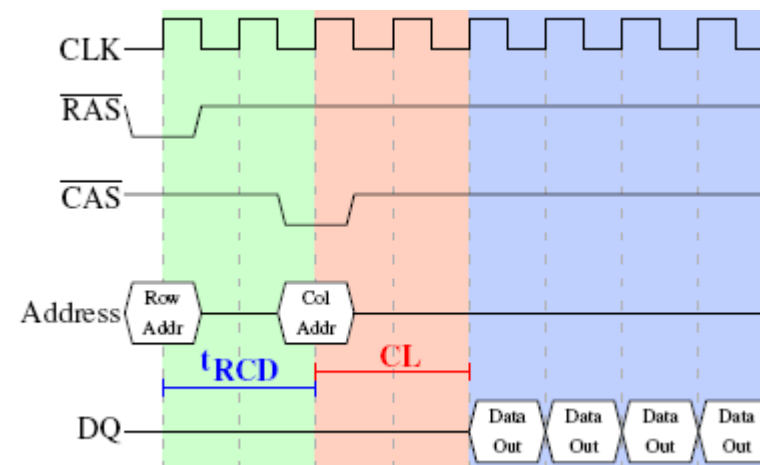
Vnitřní organizace čipu DRAM paměti



4M × 1 DRAM čip je uvnitř realizován
jako pole 2048 × 2048 1b paměťových buněk

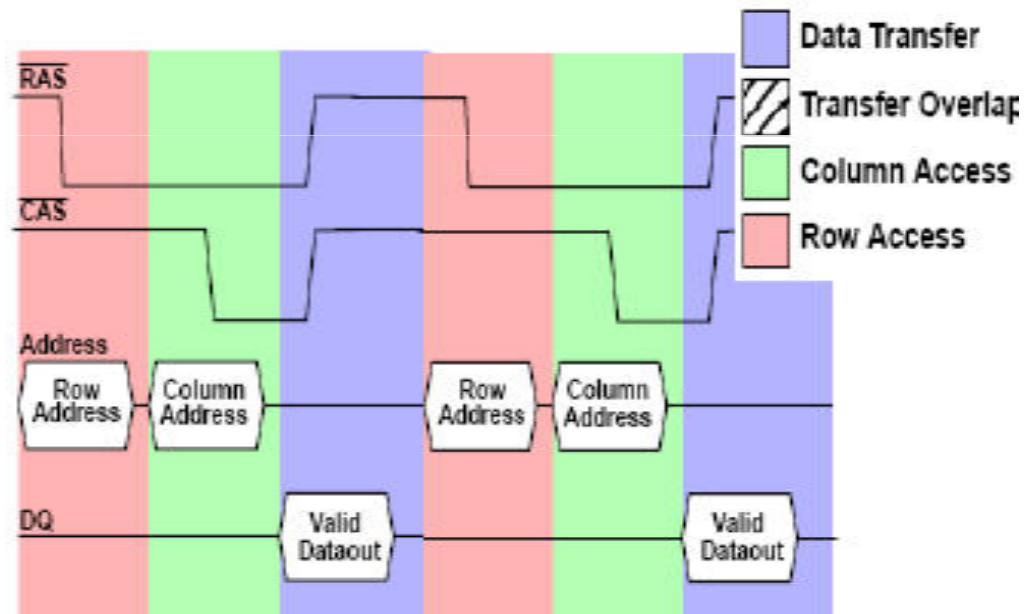
Vývoj DRAM paměťových čipů v čase

Rok	Kapacita	Cena[\$]/GB	Doba přístupu [ns]
1980	64 Kb	1 500 000	250
1983	256 Kb	500 000	185
1985	1 Mb	200 000	135
1989	4 Mb	50 000	110
1992	16 Mb	15 000	90
1996	64 Mb	10 000	60
1998	128 Mb	4 000	60
2000	256 Mb	1 000	55
2004	512 Mb	250	50
2007	1 Gb	50	40



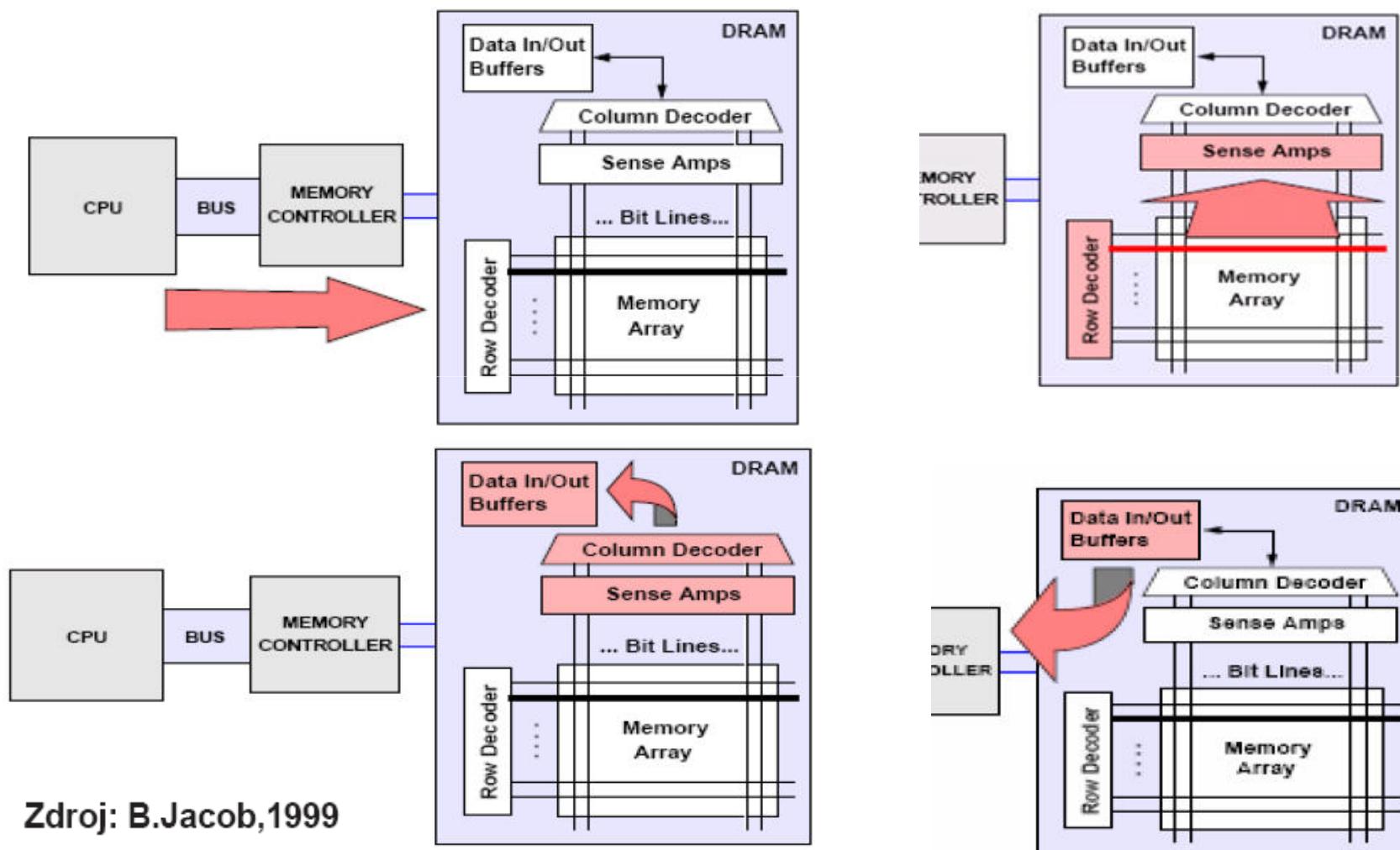
Klasická DRAM – asynchronní rozhraní

- Důvod rozdělení adresy na 2 části byl dán malým počtem pinů původních DRAM pouzder.
- Toto rozdělení se dodnes zachovává, ačkoli pouzdro už není problém. Uneslo by více vývodů...



RAS – Row Address Strobe,
CAS – Column Address Strobe

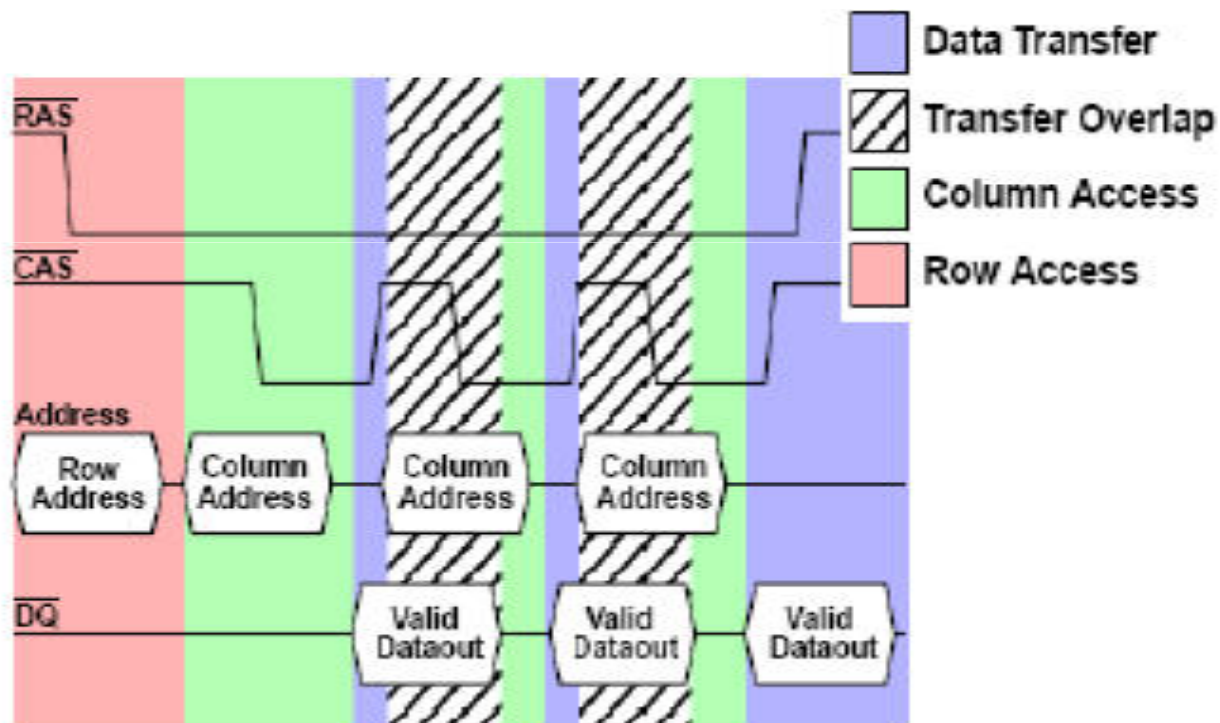
Fáze přístupu k datům v DRAM čipu



Zdroj: B.Jacob,1999

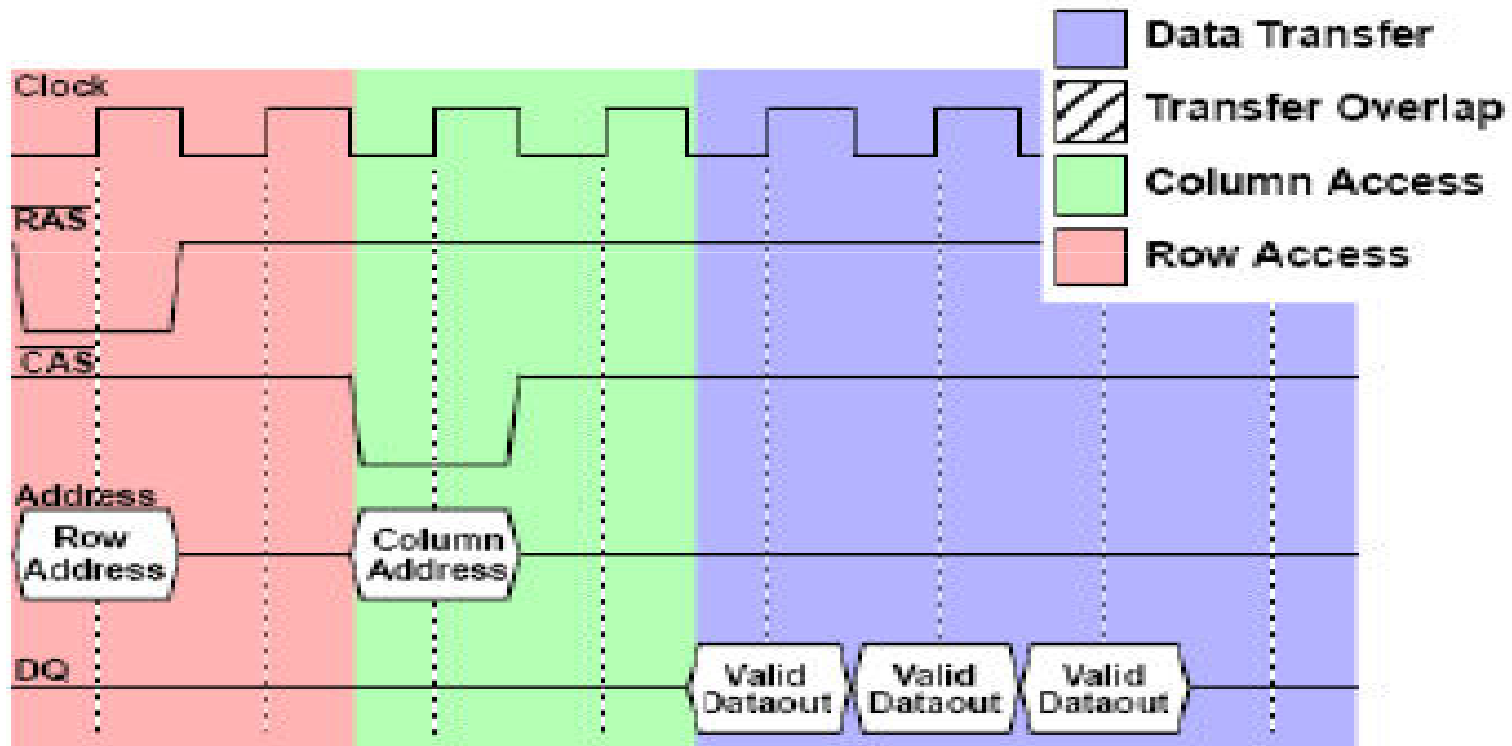
EDO DRAM – cca 1995

- EDO DRAM má registr na výstupu, což umožní překrýt následující CAS s čtením předchozích dat.



SDRAM – konec 90.let – synchronní DRAM

- SDRAM čip obsahuje čítač, který umožňuje nastavit délku souvislého (burst) čtení dat.



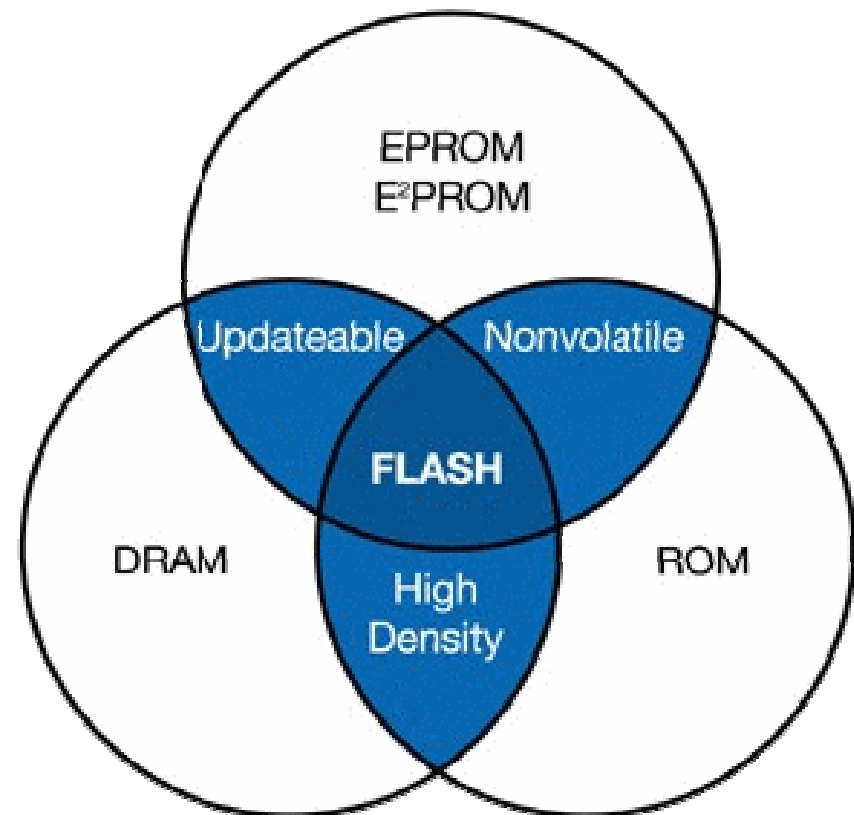
SDRAM – paměť současnosti

- **SDRAM** – frekvence až 100 MHz, 2.5V.
- **DDR SDRAM** – použití obou hran CLK při přenosu dat, 2.5V.
- **DDR2 SDRAM** – snížení spotřeby použitím napětí 1.8V, frekvence až 400 MHz.
- **DDR3 SDRAM** – snížení spotřeby použitím napětí 1.5V, frekvence až 800 MHz.
- **DDR4 SDRAM** ...
- Dále ještě existují paměti a moduly **RAMBUS**, které ovšem mají zcela odlišné rozhraní i způsob použití.
- **Všechny tyto inovace vylepšují propustnost čtení dat, nikoli latenci čtení první položky.**

Jiné principy vedlejších pamětí

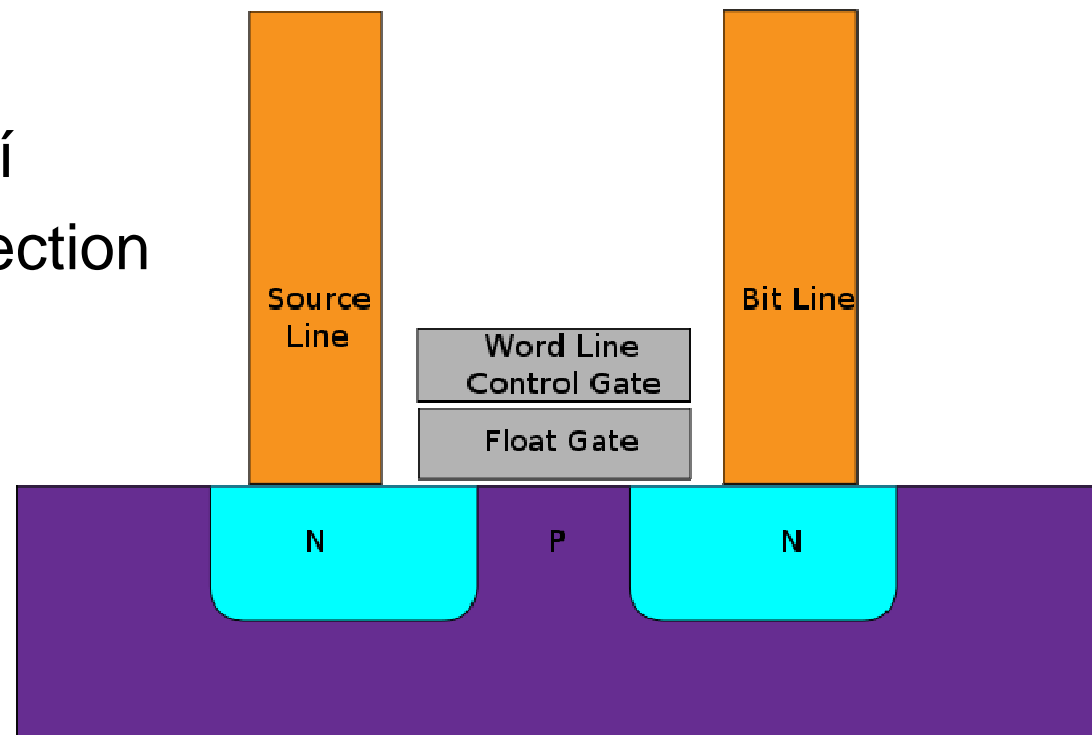
Flash

- Kombinace vlastností E²PROM, DRAM, ROM
- Data jsou ukládána v poli tranzistorů (v buňkách)
- Lze programovat každý blok samostatně
- Typu RAM (s náhodným přístupem)
- Nevolatilní počítačová paměť
- Přístupová doba (50 - 110 ns)
- Řádově stovky tisíc přemazání
- Uchová informaci několik let
- Využití: paměťové karty
USB flash disky
paměťové obvody
SSD disky



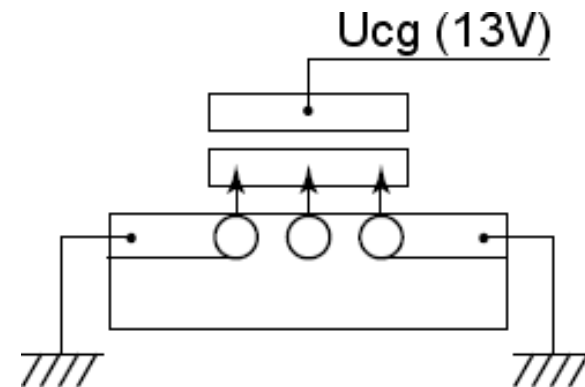
Flash paměťová buňka

- Modifikovaný transistor MOSFET rozšířený o elektricky izolované **plovoucí hradlo**
- Práce s buňkou:
 - Programování
 - F-N tunelování
 - Hot-carrier injection
 - Mazání
 - Čtení



Programování Fowler-Nordheimovo tunelování

- Na řídicí hradlo se přivede napětí U_{cg}
- Toto napětí vytvoří elektrické pole, které vytvoří potenciální bariéru
- Tato bariéra zjednoduší elektronům v substrátu cestu do plovoucího hradla
- Alternativou k programování pomocí Fowler-Nordheimova tunelování je Drain-side tunelování

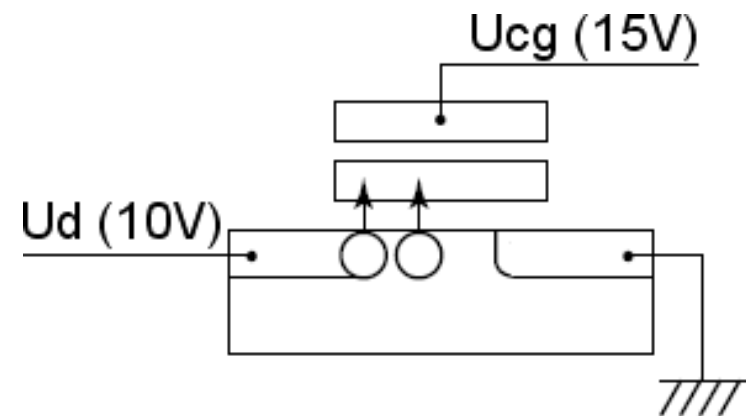


Programování Hot-carrier injection

- Existují dvě metody Hot-carrier injection
 - a) hot-electron injection (pro typ N postavený na P-substrátu) – používá se kvůli vyšší rychlosti v praxi
 - b) hot-hole injection (pro typ P postavený na N-substrátu)

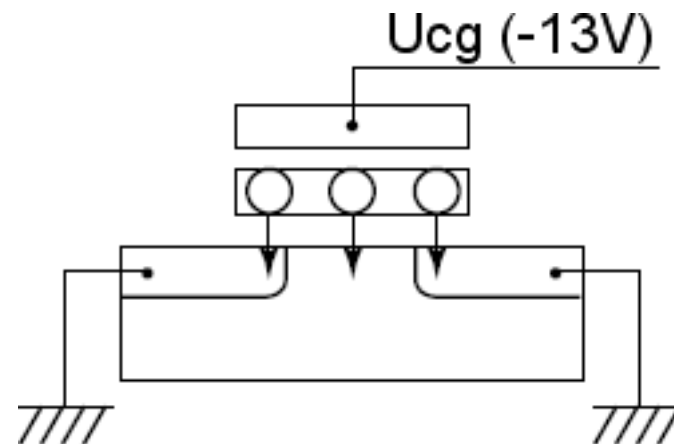
Hot-electron injection :

- 1) Napětí U_{cg} a U_d vytvoří v polovodiči elektrické pole
- 2) Toto pole urychluje elektrony z elektrody source k elektrodě drain
- 3) K elektrodě drain, ale nedoputují protože mají dostatek kinetické energie pro přechod přes vrstvu oxidu do plovoucího hradla směrem k vyššímu napětí



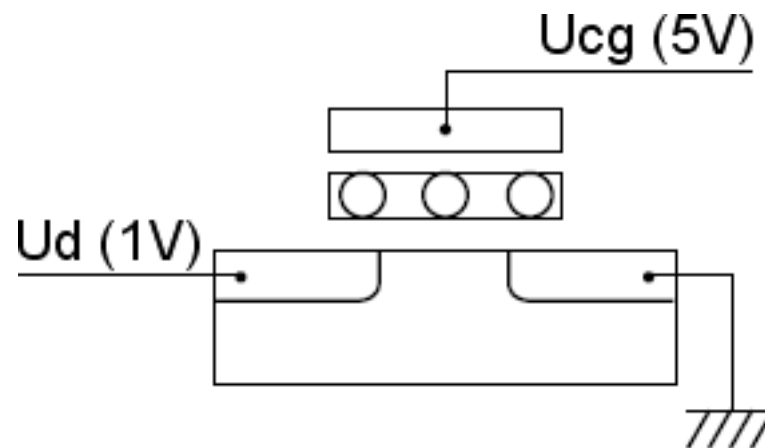
Mazání

- Mazání se provádí taktéž za použití Fowler-Nordheimova tunelování
- Elektrony jsou vypuzeny z plovoucího hradla opačnou hodnotou U_{cg} než byla použita při programování



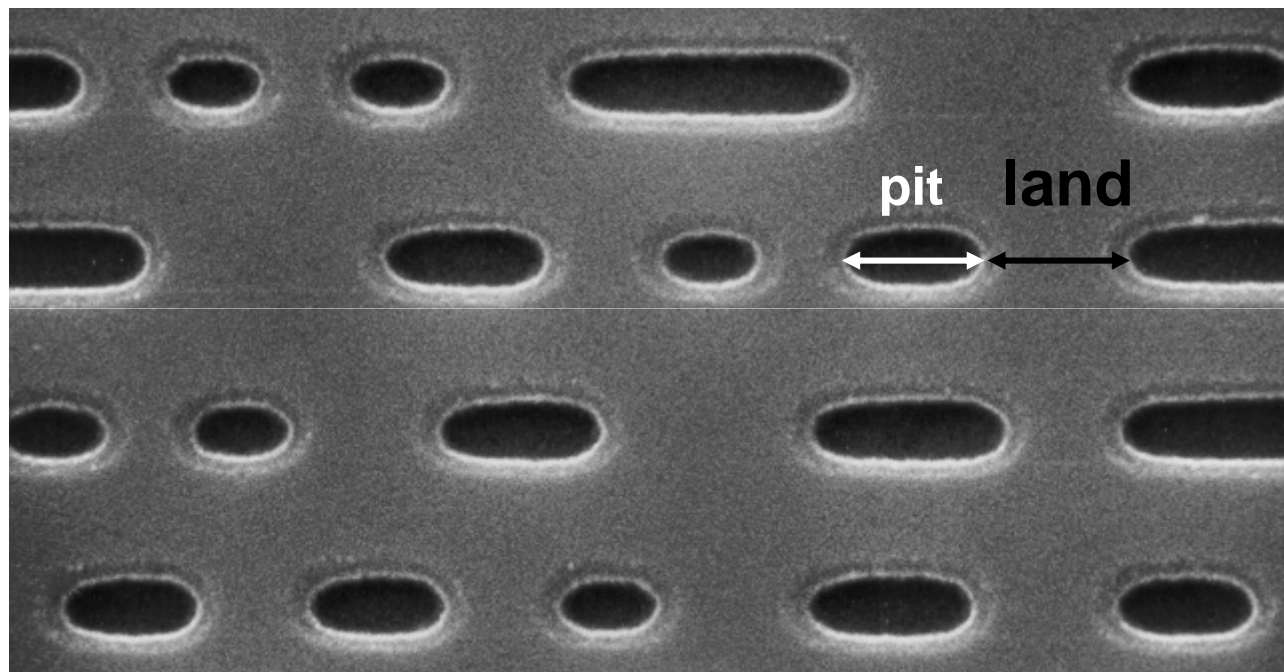
Čtení

- V závislosti na velikosti náboje v izolované elektrodě se mění elektrický proud procházející tranzistorem, který je zesílen čtecím zesilovačem a poté převeden na binární hodnotu.



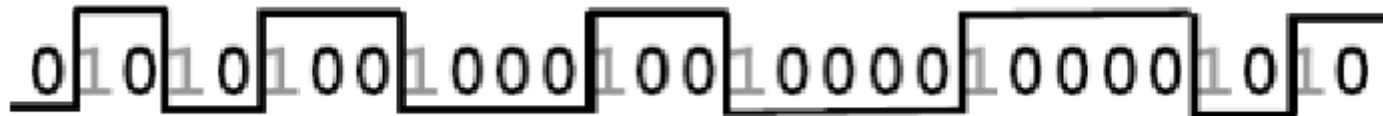
Optické paměti – detail záznamu

stopa
→



A jak se zaznamenávají „0“ a „1“?

NRZ
(d,k)



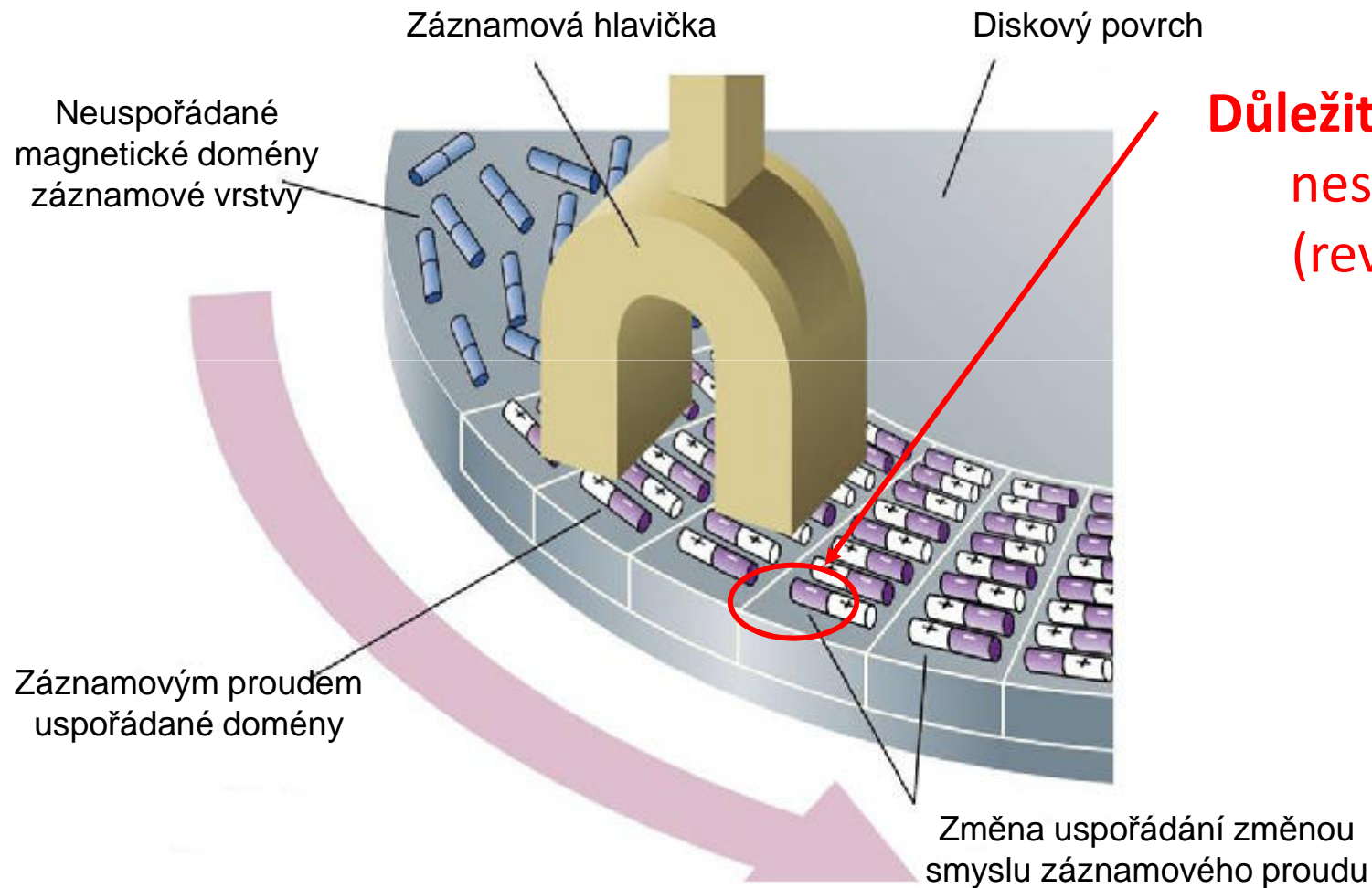
Zakódo-
vaná
data



Záznam
na médiu
(jedna
stopa)

Reverzacemi! V tomto případě optickými.

Fyzikální pozadí mg. záznamu



Otázky pro vás

- Je synonymem pro keš asociativní paměť?

What Every Programmer Should Know About Memory

Přečtěte si:

<http://www.akkadia.org/drepper/cpumemory.pdf>