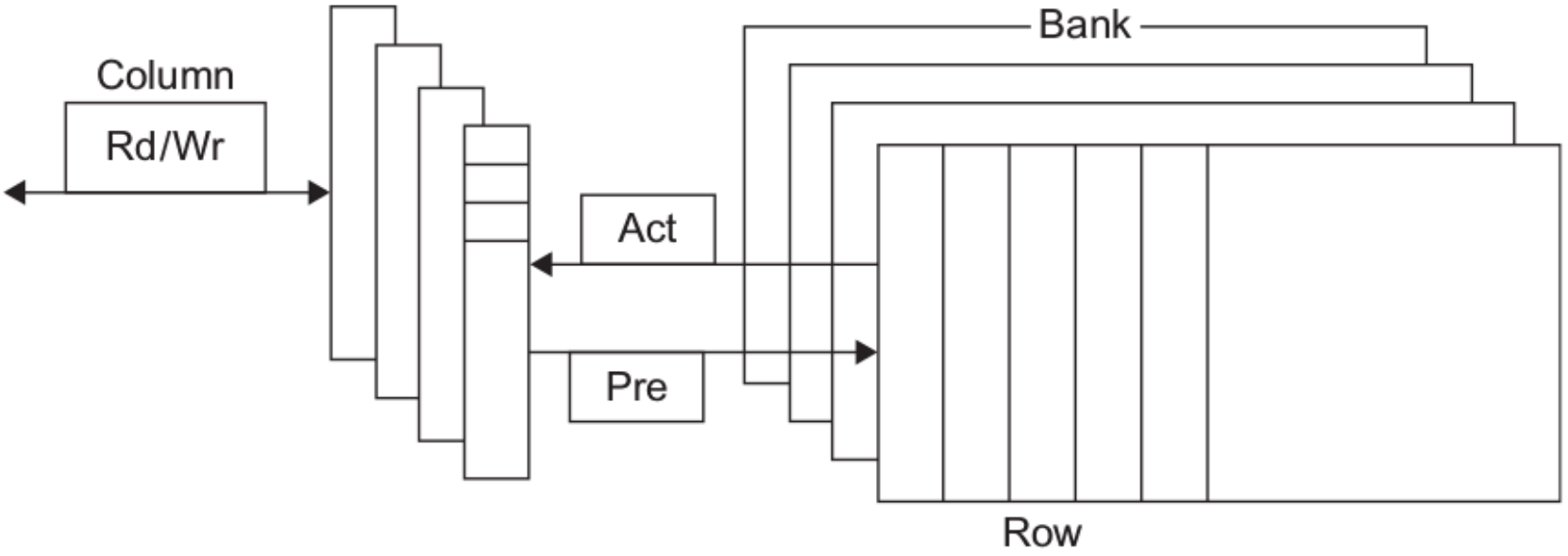
- Memorijske tehnologije
  - Osnovne tehnologije: SRAM, DRAM, Flash
  - SRAM - Static RAM, upotreba za keš
    - ne treba osvežavati sadržaj
    - vreme pristupa blisko vremenu kloka
    - tipično 6 tranzistora po bitu
    - širina linija podataka obično odgovara veličini bloka (linije), oznake se nalaze uz svaki blok
    - vreme pristupa je proporcinalno broju blokova
    - potrošnja energije proporcinalna ukupnom broju bita (statička snaga) i broju blokova (dinamička snaga)

- Memorijske tehnologije
  - DRAM - Dynamic RAM, upotreba za glavni RAM
    - multipleksiranje adresnih linija radi jednostavnijeg i jeftinijeg pakovanja, organizacija u obliku matrice gde svaka lokacija ima svoj red i kolonu
    - slanje adrese iz dva dela:
      - RAS (row access strobe)
      - CAS (column access strobe)
    - bafer koji sadrži celu kolonu (zgodno za sukcesivna čitanja)



- Memorijske tehnologije
  - DRAM - Dynamic RAM, upotreba za glavni RAM
    - Jedan tranzistor za 1 bit, čitanje je destruktivno, pa zahteva ponovni upis
    - Zbog struje curenja, napon na tranzistoru opada, pa sadržaj treba periodično osvežavati (što se svodi na čitanje i ponovni upis)
    - Istorijat
      - prvi DRAM čipovi su bili asinhroni, odnosno moralo se čekati na sinhronizaciju sa kontrolerom memorije
      - 1990 se pojavio sinhroni DRAM - SDRAM, uvođenje burst režima gde se pošalje jedna adresa reda, pa se čita sve iz zadate kolone (sukcesivne lokacije)
      - 2000 se pojavio double data rate - DDR, slanje podataka i na rastuću i na opadajuću ivicu kolka
      - uvođenje banki, odnosno više čipova na jednom modulu, što je omogućilo preplitanje memorije (interleaving) - sukcesivne grupe lokacija se nalaze u različitim čipovima, RAS se deli na adresu banke i dresu reda

- Memorijske tehnologije
  - DRAM - Dynamic RAM, upotreba za glavni RAM
    - Rast brzine i kapaciteta se znatno usporio posle 2010
    - Danas dolaze u DIMM (dual inline memory module) pakovanjima, sa 4-16 DRAM čipova
    - Radi smanjenja potrošnje, smanjivao se napon, pa danas DDR4 radi na naponu od oko 1.2V
    - GDRAM - prilagođen većoj propusnosti koju zahtevaju grafičke kartice
      - imaju širi data bus
      - rade na višim frekvencijama, direktno je povezan na GPU (zalemljen na istoj štampanoj ploči)
      - GDDR5, zasnovan na DDR3, ima 2-5 veći propusni opseg od DDR3
    - HBM (high bandwidth memory) - slaganje DRAM pločica jedna na drugu (3D matrica), zasad uglavnom kod grafičkih kartica

- Memorijske tehnologije
  - Flash
    - Vrsta EEPROM-a (electronically erasable), danas se najviše koristi NAND varijanta
    - Danas osnovna masovna memorija u prenosnimn uređajima
    - Karakteristike
      - Čitanje je sekvencijalno i po blokovima (512B-4KiB), za prvi bajt se čeka malo duže, ostali stižu dosta brže. Oko 150 puta sporija od DDR4, ali i 200-500 puta brža od hard diska
      - Pre ponovnog upisa se mora obrisati. Brisanje se radi po blokovima. Upis je oko 1500 puta sporiji nego DDR4, ali 8-15 puta brži od hard diska
      - Sadržaj čuva i bez napajanja, te ne troši puno kada se ne koristi
      - Blok se može brisati ograničen broj puta, oko 100k. Kontroler Flash-a se brine o tome da se blokovi ravnomerno pišu-brišu
      - Jefininija od DRAM-a, skuplja od hard diska po jedinici kapaciteta
      - Postojanje rezervnih blokova koji mogu zameniti defektne

- Memorijske tehnologije
  - Povećanje pouzdanosti memorije
    - hard errors, permanent faults - defekti u integrisanim kolima koje su stalnog karaktera i koji uvek izazivaju grešku
      - svaki čip ima jedan deo rezervnih ćelija koje se mogu koristiti umesto defektnih
    - soft errors, transient faults - nenamerne promene u sadržaju ćelija
      - mogu se detektovati parity bitima ili kodovima za korekciju grešaka (EEC - error correcting codes), i jedno i drugo zahteva redundantne ćelije
      - EEC memorija - uz dodatak od 8 bita na svaka 64, omogućava se detekcija greške na 2 bilo koja bita, kao i korekcija greške na 1 bitu

- Memorijske tehnologije
  - Povećanje pouzdanosti memorije
    - u veoma velikim sistemima verovatnoća grešaka u memoriji značajno raste
      - Korišćenje Chipkill tehnologije - slično RAID-u, ali za RAM
    - IBM-ova analiza za period od 3 godine, za sistem za 10k procesora i 4GB RAM po procesoru, broj nepopravljivih grešaka:
      - korišćenjem samo parity bita: oko 90k grešaka, ili 1 svakih 17 minuta
      - korišćenjem ECC memorije: oko 3500 grešaka, ili jedna svakih 7.5 sati
      - Chipkill: otprilike jedna na svaka 2 meseca

- Optimizacija keš memorije
  - Šta se može popraviti?
    - smanjenje vremena pogotka
    - povećanje propusnosti
    - smanjenje vremena promašaja
    - smanjenje frekvencije promašaja
    - upotreba paralelizma za smanjenje vremena i/ili frekvencije promašaja
    - smanjenje potrošnje energije
  - Kako se može popraviti?
    - unapređenje tehnologije integrisanih kola
    - unapređenje organizacije memorije i njenog korišćenja
    - korišćenje boljih kompajlera

- Optimizacija keš memorije

Technique	Hit time	Band-width	Miss penalty	Miss rate	Power consumption	Hardware cost/ complexity	Comment
Small and simple caches	+			–	+	0	Trivial; widely used
Way-predicting caches	+				+	1	Used in Pentium 4
Pipelined & banked caches	–	+				1	Widely used
Nonblocking caches		+	+			3	Widely used
Critical word first and early restart			+			2	Widely used
Merging write buffer			+			1	Widely used with write through
Compiler techniques to reduce cache misses				+		0	Software is a challenge, but many compilers handle common linear algebra calculations
Hardware prefetching of instructions and data			+	+	–	2 instr., 3 data	Most provide prefetch instructions; modern high-end processors also automatically prefetch in hardware
Compiler-controlled prefetching			+	+		3	Needs nonblocking cache; possible instruction overhead; in many CPUs
HBM as additional level of cache		+/–	–	+	+	3	Depends on new packaging technology. Effects depend heavily on hit rate improvements

- Virtuelna memorija
  - Ima ulogu sličnu ulozi keša kod RAM memorije: držanje u RAM-u neophodnih stranica za izvršavanje programa, čije se kompletne slike nalaze na masovnoj memoriji
  - Multiprogramiranje - više procesa (programa) istovremeno dele resurse računara
    - zahtev za međusobnom izolacijom procesa
      - zašto?
  - Operativni sistem i hardver moraju ograničiti šta korisnički procesi mogu da koriste, dok s druge strane moraju omogućiti i rad sistemskih procesa

- Virtuelna memorija
  - Arhitektura mora obezbediti:
    - bar dva režima izvršavanja, jedan za korisničke procese (user), jedan za sistemske (kernel, supervisor) procese
    - obezbedi read pristup korisničkim programima za neke delove procesora
      - user/supervisor bit
      - podaci o zaštićenoj memoriji (npr. šta je dodeljeno procesu)
    - obezbedi mehanizme za prelazak procesora sa user na supervisor režim rada i nazad
      - sistemski pozivi
      - povratak iz sistemskog poziva
    - obezbedi mehanizme za ograničavanje prisupa memoriji
  - Najčešće korišćena tehnika zaštite memorije je dodavanje podataka o ograničenjima za svaku stranicu virtuelne memorije
    - read/write/execute prava

- Virtuelna memorija
  - Informacije o tome koja virtuelna stranica odgovara kojoj fizičkoj se takođe nalaze u RAM-u
    - što znači da je svaki pristup memoriji duplo duži nego obično - prvo se pristupa tabeli stranica da bi de dobila adresa fizičke stranice, pa tek onda adresi u fizičkoj stranici
    - uvođenje keš memorije za ubrzanje prisupa fizičkim adresama
      - TLB - translation lookaside buffer - tag je virtuelna adresa, a sadržaj je
        - fizička adresa
        - podaci o zaštiti - read/write/execution prava
        - protection bit - zabrana odlaganja stranice u masovnu memoriju
        - valid bit - da li se u sadržaju uopšte nalazi adresa
        - dirty bit - da li je stranica menjana
        - use bit - da li je stranica korištena u poslednje vreme

- Virtuelna memorija
  - Mehanizmi zaštite bi trebalo da omoguće poptunu međusobnu izolaciju procesa
    - Nažalost, nije tako, jer postoje greške i propusti kako u operativnim sistemima, tako i u arhitekturi i/ili hardveru. 2017-te su otkriveni meltdown i spectre propusti. Meltdown:
      - proces može čitati svu raspoloživu memoriju, čak i kada nema prava za to (Intel, AMD, Power, ARM)
      - napravi se niz od 256 stranica u memoriji (256*4KB), isprazni se keš čitanjem random lokacija po memoriji, a zatim se pokuša čitanje željene lokacije, a odmah zatim pristup elementu niza pomoću vrednosti lokacije pomnožene sa 4096
      - procesor će probati da izvrši prvu naredbu, ali će “pući” na proveri granica memorije/prava pristupa; međutim, takođe će i unapred izvršiti i drugu naredbu i keširati sadržaj jedne stranice niza
      - zatim se u petlji pokušava direktno pristupati prvom bajtu svake stranice niza i meri se vreme za koje se dobije podatak
      - kada se naleti na čitanje vrednosti koje se dobije brže (odnosno, iz stranice koja je keširana), redni broj stranice odgovara vrednosti lokacije koja je tražena
    - Spectre - slična stvar, samo generalnija i bazirana na spekulativnom izvršavanju posle naredbi skoka

- Virtuelne mašine
  - Postoje još od 60-tih godina na mainframe računarima, od 80-tih su izgubile na poluparnosti, ali su u poslednjih 10-15 godina ponovo aktuelne
    - veća izolacija procesa
    - izbegavanje propusta u operativnim sistemima
    - deljenje resursa računara na više korisnika
    - overhead upotrebe virtuelnih mašina se znatno smanjio
  - Najčešće je unutar VM podržana ista arhitektura kao na host računaru
    - Xen, KVM, VirtualBox, VMWare, ...
    - Ima i izuzetaka: QEMU, emulacija starih računara
  - VMM (Virtual Machine Monitor) - softver koji omogućava rad VM, mapira resurse guest sistema na host sistem

- Virtuelne mašine
  - Overhead zavisi od softvera koji se izvršava unutar guest sistema
    - ako je više zavisan od procesora, overhead je manji
    - ako je više zavisan od sistemskih poziva i generalno od I/O, overhead je veći
    - neke naredbe se mogu direktno izvršiti na host procesoru, neke se moraju emulirati (izvršiti programski kod koji daje efekat izvršavanja naredbe)
  - Dodatne mogućnosti VM
    - Korišćenje (starijeg) softvera - unutar VM se može instalirati kompletno okruženje potebno za neki zadatak, pa se multiplicirati bez uticaja na host sistem. Ovo uključuje i korišćenje softvera koji zahteva stariji OS i/ili biblioteke
    - Bolje iskorištenje hardvera - različiti delovi softvera koji zahtevaju različite verzije OS i/ili biblioteka se mogu pokrenuti unutar više VM na istom računaru, umesto na više računara