Tečajevi naprednog računarstva u Križevcima

HACK2020 | Hub for Advanced Computing Križevci

Osnove računarstva visokih performansi

Vol. 2: CPU & Memory

Andrej Dundović

Križevci, 25. 6. 2021.

Organizator:

udruga point križevci

Pokrovitelj:

Embassy of the United States
Zagreb - Croatia

Podsjetnik na motivaciju

Figure: Žene-računala

Arhitektura računala I

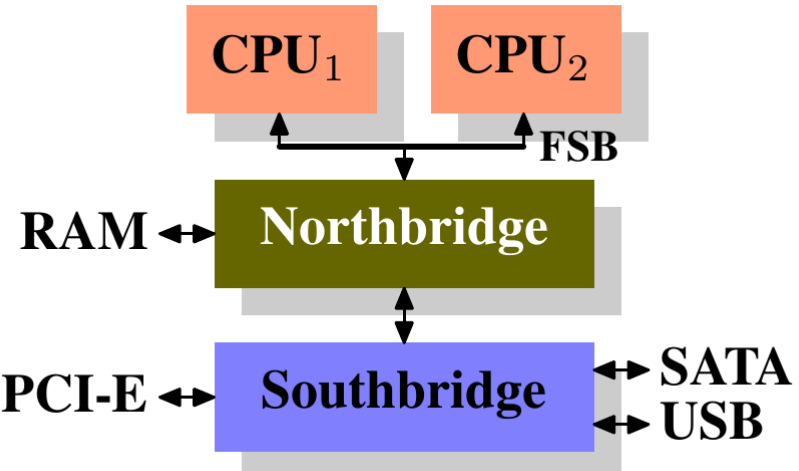


Figure: Komunikacija između osnovnih komponenti u računalu

Arhitektura računala II

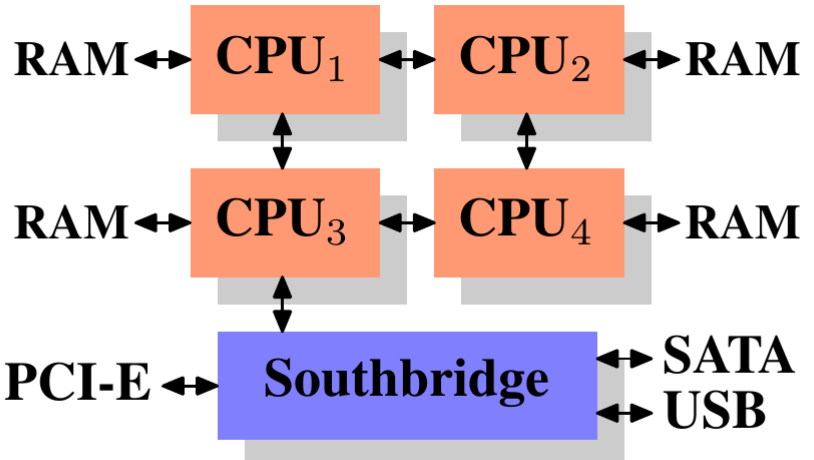
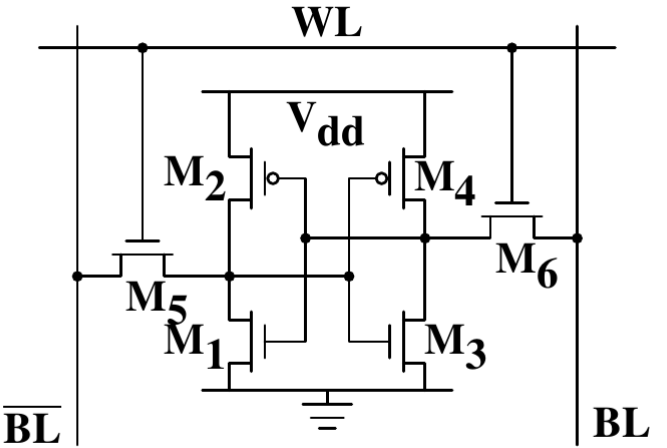
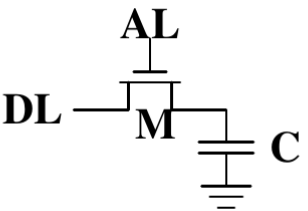


Figure: Komunikacija između osnovnih komponenti u računalu

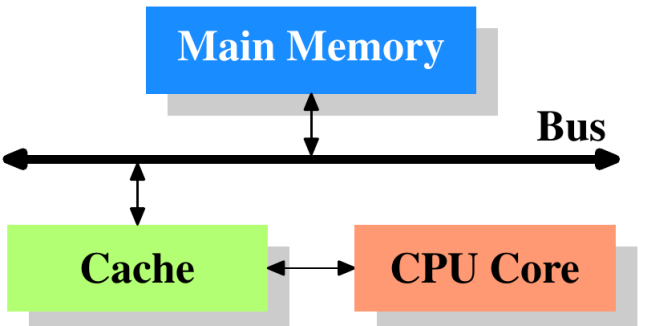
Static RAM



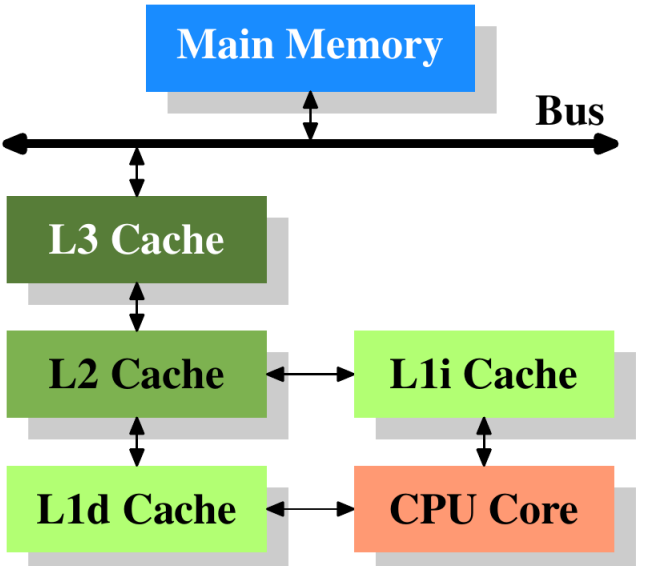
Dynamic RAM



Cache / Predmemorija



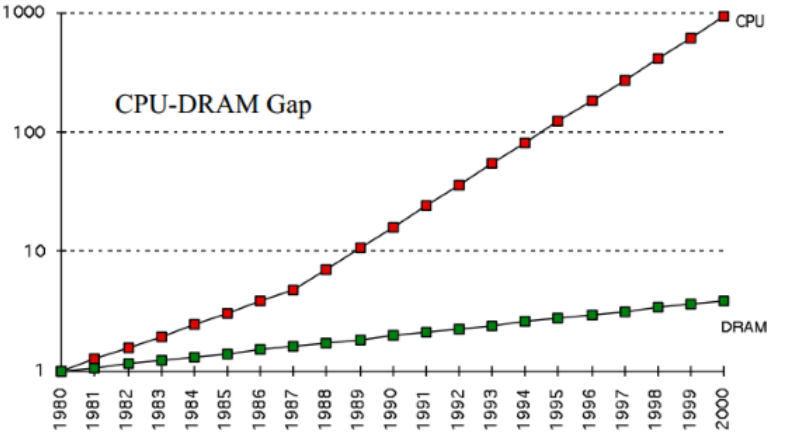
Cache / Predmemorija - L1, L2, L3



 andrej.dundovic@udruga-point.hr

Usporedba razvoja CPU / DRAM

- Processor vs Memory Performance



1980: no cache in microprocessor;

1995 2-level cache

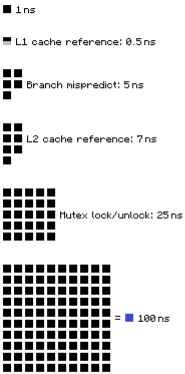
 andrej.dundovic@udruga-point.hr

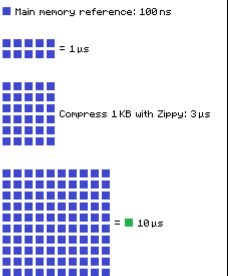
Odnosi brzina predmemorija i DRAM-a

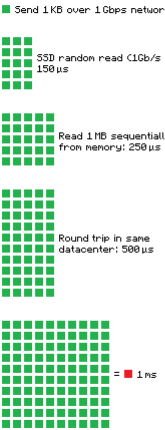
Primjer odnosa brzina predmemorija i DRAM-a za Core i7 Xeon 5500 Series Data Source Latency (aproksimativno)

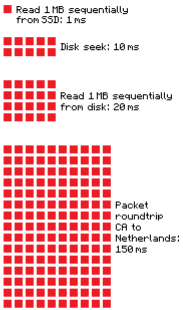
Naziv	broj ciklusa	latencija
local L1 CACHE hit	4	2.1 – 1.2 ns
local L2 CACHE hit	10	5.3 – 3.0 ns
local L3 CACHE hit	40	21.4 – 12.0 ns
...		
local DRAM		$\tilde{6}$0 ns
remote DRAM		$\tilde{1}$00 ns

Latency Numbers Every Programmer Should Know









Source: https://gist.github.com/2841832

Principi za bolju iskoristivost predmemorije

- Prostorna lokalnost – ako se program referira na podatke na određenoj adresi, vjerojatno je da će se referirati i na susjedne adrese
- Vremenska lokalnost – ako se program referira na podatke, vjerojatno je da će se referirati opet na iste podatke

Konkretnije?

- Izbjegavati algoritme i strukture podataka koji pristupaju memoriji po nepravilnim obrascima
- Valja koristiti manje podatkovne tipove i organizirati ih tako da nema praznina kod poravnavanja (*alignment holes*)
- Iskoristiti linearne podatkovne strukture (polja, vektori, stogovi nasuprot grafovima i stablima)