

# Implementacija disk keš kontrolera koji koristi periode „tišine“

Zlatković, S., Delečić, D., Lazarević, P., ☎ 011/699-911

Laboratorija za VLSI, Katedra za računarsku tehniku, Elektrotehnički fakultet u Beogradu

## 1. UVOD

U ukupnom vremenu rada jednog računarskog sistema, udeo rada disk podsistema veoma je značajan. Što su procesori brži, to je veći relativan udeo rada disk podsistema. Ovo povećanje nije posledica povećanog obima poslova disk podsistema, već zaoštavanja u poboljšanju njegovih performansi u odnosu na performanse ostatka računarskog sistema. Zbog toga se danas o disk podsistemu govori kao o uskom grlu računarskog sistema. Problem otklanjanja ovog uskog grla tema je mnogobrojnih studija i istraživanja.

### 1.1 Disk keševi

Uvođenje skrivene memorije (cache memory), čiji je zadatak da iskoristi prostornu i vremensku lokalnost podataka traženih sa diska, predstavlja suštinu rešenja navedenog problema. Jednostavno preslikavanje koncepta procesorskog keša u disk podsistem nije dalo maksimalne rezultate. Takvi rezultati mogu se dobiti jedino pristupom problemu disk keša koji bi uvažavao sve specifičnosti disk keša, uz primenu novih metoda i algoritama.

Istraživanje, za koje je opis poslednje faze dat u ovom radu, predstavlja konkretizaciju novog pristupa disk keš memoriji (Lazarević/Milutinović pristup), kao delu disk podsistema (vidi i [1], [3]). Faze istraživanja koje su prethodile implementacionoj analizi (pregled postojećih rešenja, analitička analiza i simulaciona analiza) dale su punu potvrdu takvom pristupu.

### 1.2 PL VLSI

Metodologije projektovanja VLSI sistema se dele u tri osnovne grupe:

- projektovanje na bazi geometrijskih simbola (*full custom VLSI*);
- projektovanje na bazi logičkih simbola, koje se dalje deli na tri podgrupe:
  - projektovanje na bazi standardnih ćelija (*standard cell VLSI*),
  - projektovanje na bazi gejitovskih matrica (*gate array VLSI*), i
  - projektovanje na bazi programabilne logike (*programmable logic VLSI - PL VLSI*);
- projektovanje na bazi funkcionalnog opisa korišćenjem jezika za opis hardvera (*silicon translation VLSI*).

Kako je za izradu prototipova, kao i relativno malih serija, najpogodnija metoda PL VLSI, implementacija dela disk keš kontrolera izvršena je baš ovom metodom.

Za ovu metodu se nakon završetka projektovanja konkretnog sistema na čipu, vrši samo aktiviranje potrebnih veza (vidi [4]); sve moguće veze su realizovane u postupku predfabrikacije, a aktiviraju se samo one koje su predviđene datim projektom. Ovo se postiže upisivanjem odgovarajućeg sadržaja u internu ROM ili RAM memoriju na samom čipu.

## 2. DEFINICIJA PROBLEMA

Implementacija celog disk keš kontrolera predstavlja ozbiljan posao koji je po pravilu povezan sa velikim ulaganjima. Pored toga, implementacija većeg dela disk kontrolera ne predstavlja značajan doprinos istraživanju u kome je implementaciona analiza jedna od faza. Zbog toga je ovaj rad posvećen delu disk keš kontrolera nove strategije, koji čini suštinu novog pristupa projektovanju disk keša, sa ciljem da se pokaže mogućnost njegove realizacije u okviru raspoložive tehnologije.

### 2.1 Analiza

Da bi se mogao napraviti dobar izbor među više kandidata za implementaciju, posebno je prethodno definisati kriterijume za odlučivanje. Kriterijumi za izbor dela (bloka) disk keš kontrolera koji će se implementirati, u ovom radu su bili: neophodnost implementacije u hardveru u odnosu na implementaciju u softveru, inovativnost, kvantitativna kompleksnost (u odnosu na resurse VLSI čipova) i kvalitativna (kognitivna) kompleksnost.

Na osnovu faza istraživanja koje su prethodile ovom radu, a posebno simulacione analize, definisano je nekoliko blokova disk keš kontrolera od interesa za realizaciju u hardveru:

- podrška algoritmu za asocijativno pretraživanje,
- podrška LRU algoritma,
- RISC procesor za podršku ažuriranja statistika, i
- podrška algoritmu za dohvatanje unapred.

Izabran je deo za podršku algoritma za dohvatanje unapred sadržaja disk keša (*Prefetch Algorithm Support - PAS*), koji radi u periodima „tišine“, korišćenjem statistika o najverovatnijim budućim obraćanjima disk podsistemu.

### 2.2 Važnost problema

Algoritam za dohvatanje unapred sadržaja disk keša (*prefetch algorithm*) predstavlja srž novog pristupa problemu disk keša koji do sada nije realizovan. Realizacija u hardveru neophodna je zbog toga što blok PAS radi u periodima tišine, pa treba da bude brz i nezavisan od ostatka kontrolera.

## 3. O METODAMA ČIJA SE IMPLEMENTACIONA KOMPLEKSNOŠĆ POREDI

Do sada najbolji disk keš kontroleri (jedan od njih je prezentovao Reddy) koristili su periode „tišine“ disk keša. Sve ostale aktivnosti obavljali su samo i isključivo prilikom opsluživanja zahteva upućenog disk podsistemu.

Disk keš kontroler, čija je implementacija prikazana u ovom radu, koristi u mnogo većoj meri specifičnosti disk keša i pomenute periode tišine, u kojima se, pored čišćenja, vrši i preuređenje sadržaja keša. Promena sadržaja keša vrši se na osnovu statistika - informacija o toku izvršenja aplikacija i odgovarajućim obraćanjima disku.

Simulaciona analiza pokazala je da korišćenjem statistika prosečne učestalosti novi disk keš kontroler ima performanse za 50% bolje od danas najboljih kontrolera [1].

#### 4. LATTICE PL VLSI TEHNOLOGIJA

Lattice pLSI (*programmable Large Scale Integration* - programabilna tehnologija visokog stepena integracije) i isplSI (*in-system programmable Large Scale Integration* - programabilna „u sistemu“ tehnologija visokog stepena integracije) jesu dve familije E<sup>2</sup>C<sup>2</sup>MOS programabilnih logičkih kola velike gustine pakovanja i izrazito dobrih performansi. Bazična arhitektura pLSI i isplSI kola je u osnovi matricna. Sa gustim PL gejtova u opsegu od 2000 do 8000, pLSI i isplSI familije obezbeđuju širok spektar programabilnih logičkih rešenja. Programabilnost u sistemu velika je prednost Lattice tehnologije u odnosu na konkurentne, u slučaju istraživanja i razvoja novih uređaja.

#### 5. STRUKTURA PROJEKTA

PAS blok predstavlja deo disk keš kontrolera koji celokupnu svoju aktivnost obavlja u periodima tišine. Prilikom zadovoljavanja zahteva upućenog disk podsystemu, PAS prati adrese zahtevanih klastera (zahteve). Nakon što se takav zahtev zadovolji, počinje izvršavanje aktivnosti perioda tišine, korišćenjem statistika i informacije o adresi poslednjeg zahtevanog klastera. Posebno razvijeni algoritmi za dohvatanje unapred sadržaja disk keša čine to moguće staze izvršavanja koda ravnopravnim [1], pa se „front“ popunjavanja keša na osnovu statistika širi ravnomerno, tj. istom brzinom po svim stazama - što je duži period tišine - to je front dalje od tekuće linije izvršavanja programa.

##### 5.1 Struktura bloka PAS

PAS blok disk keš kontrolera realizovan je kao mikrokontroler. Program, koji predstavlja realizaciju algoritma za dohvatanje unapred, PAS izvršava u koracima. U svakom koraku vrše se odgovarajuće operacije koje su implementirane u hardveru.

PAS se sastoji iz sledećih funkcionalnih celina:

- programskog brojača (*program counter* - PC) i dekoderske mreže,
- interfejsa ka ostatku disk keš kontrolera,
- interfejsa ka specijalizovanoj memoriji (statusnoj i sa statistikama), i
- pomoćne interne logike.

##### 5.2 Lattice realizacija

Da bi se implementirao blok PAS bilo je potrebno upotrebiti Lattice isplSI (ili pLSI) 1048 kolo (na slici). Ono sadrži 288 registara, 96 univerzalnih I/O pinova, deset specijalizovanih ulaznih pinova, četiri specijalizovana ulaza za takt i jednu globalnu spojnu mrežu koja obezbeđuje potpunu međusobnu povezanost svih navedenih elemenata.



Za realizaciju bloka PAS u Lattice čipu, kritičan resurs predstavljao je broj pinova, zbog povezanosti razmatranog bloka kao sa ostatkom kontrolera, tako i sa specijalizovanim memorijama.

#### 6. ZAKLJUČAK

U ovom radu predstavljena je implementacija dela poboljšanog disk keš kontrolera koji vrši dohvatanje podataka unapred, korišćenjem perioda tišine. Periodi kada nema eksternih zahteva disk podsystemu koriste se za inteligentnu popunu disk keša podacima za koje postoji najveća verovatnoća da će biti traženi u bliskoj budućnosti. Suština algoritama po kome se vrši dohvatanje, a koji je razvijen tokom ovog istraživanja, jeste u ravnopravnosti svih naleta zahteva koji su obuhvaćeni statistikama. Ovaj rad opravdava investiranje u dodatni hardver za postojeće disk keš kontrolere, jer pokazuje da je implementacija novog i sofisticiranog dela koji predstavlja suštinu novog pristupa u projektovanju disk kontrolera, moguća i to u samo jednom PL VLSI Lattice čipu.

Koristi od ovog rada imaju projektanti budućih disk podsystema, a takođe i svih sistema sekundarne memorije (CD ROM i CD WORM jedinice, RAID sistemi).

Novootvoreni problemi vezani su za budući rad koji je usmeren na realizaciju preostalog dodatnog dela disk keš kontrolera, kao i modifikaciju postojećih delova kontrolera.

#### 7. REFERENCE

- [1] Delečić, D., Lazarević, P., Milutinović, V., Zlatković, S., "Simulation Analysis of an Improved Disk Cache Controller For Utilization of 'Silence'." Technical Report, TI-ETF-RTI-93-031, School of Electrical Engineering, University of Belgrade, Belgrade, Serbia, Yugoslavia, October 1993.
- [2] *Lattice pLSI and isplSI Data Book and Handbook*, Lattice Semiconductor Corporation, Hillsboro, Oregon, U.S.A., 1992.
- [3] Lazarević, P., Milutinović, V., "Application of Trace Scheduling to Disk Cache Contents on the Silence," Technical Report, TI-ETF-RTI-91-010, School of Electrical Engineering, University of Belgrade, Belgrade, Serbia, Yugoslavia, June 1991.
- [4] Milutinović, V., *Projektovanje i arhitektura RISC procesora za VLSI*, Nauka, Beograd, Srbija, Jugoslavija, 1994.

Zahvalnica: Rad na ovom projektu delimično su finansirali Fond za nauku i Fond za tehnološki razvoj Republike Srbije, preduzeće IFACT, Beograd, Jugoslavija, a delimično i NCR Corporation, Augsburg, Nemačka.

**Abstract:** This work presents one implementation of a part of an improved disk subsystem controller, based on disk cache. A Prefetch Algorithm Support (PAS) block was implemented. Activities of the specified block are related to "silence" periods (time periods without external requests to disk subsystem). During the silence periods disk cache is filled with data most likely to be requested from the disk in the near future. This intelligent disk cache filling is done according to statistics, which contains possible successors of the last requested cluster from disk. The originally developed prefetch algorithm ensures the equality of all future bursts of accesses contained in statistics.

Implementation of the Disk Cache Controller For Utilization of the "Silence", Zlatković, S., Delečić, D., Lazarević, P.