Proiect Controller Cache 4-Way Set-Associative

# 1. Introducere

Acest proiect are ca obiectiv proiectarea, implementarea și testarea unui controller de cache de tip 4-way set-associative, cu o capacitate totală de 32KB, dimensiune a blocului de 64 bytes, și o dimensiune a cuvântului de 4 bytes.  
  
Controllerul este implementat în Verilog, fiind organizat în jurul unei mașini de stări finite (FSM) pentru a controla toate operațiile de citire, scriere, încărcare și evacuare. Proiectul reprezintă o aplicație practică a cunoștințelor dobândite în cadrul cursului de Calculatoare Numerice, consolidând înțelegerea arhitecturii memoriei cache.

# 2. Specificații Tehnice

- Tip cache: 4-way set-associative  
- Capacitate totală: 32KB  
- Dimensiune bloc (linie): 64 bytes  
- Dimensiune cuvânt: 4 bytes  
- Număr de seturi: 128  
- Politică de înlocuire: Least Recently Used (LRU)  
- Politică de scriere: Write-back cu write-allocate  
  
Cache-ul este organizat în 128 de seturi, fiecare conținând 4 linii (ways). Politica LRU gestionează selecția liniei pentru evacuare în cazul unui cache miss pe un set complet ocupat. Politica de scriere write-back implică scrierea datelor în memoria principală doar în momentul evacuării, iar write-allocate presupune aducerea datelor în cache înaintea unei scrieri.

# 3. Arhitectura Generală

Arhitectura este modulară, compusă din:  
  
- cache\_controller.v: FSM-ul care orchestrează comportamentul cache-ului  
- cache\_memory.v: implementarea memoriei cache, cu tag-uri, date, valid/dity/LRU bits  
- main\_memory.v: model de memorie principală simplificată  
- four\_way\_set.v: instanțiază cele 4 linii per set și logica de selectare  
- cache\_line.v: descrierea unei linii individuale din cache  
- fsm\_structural.v: structură de top-level pentru instanțierea modulelor  
- cache\_controller\_tb.v: testbench-ul pentru validare funcțională

A diagram of a computer

AI-generated content may be incorrect.

# 4. Mașina de Stări Finite (FSM)

Controllerul de cache utilizează o FSM pentru a gestiona ciclurile de operare. Stările principale sunt:  
  
- IDLE: Așteaptă o solicitare nouă  
- READ\_HIT: Operație de citire găsită în cache  
- READ\_MISS: Citirea nu este găsită – se inițiază încărcarea  
- WRITE\_HIT: Scriere directă în cache (flag dirty activat)  
- WRITE\_MISS: Scriere pe o linie inexistentă – se alocă bloc  
- EVICT: Se evacuează o linie (dacă este necesar) conform politicii LRU

A diagram of a diagram

AI-generated content may be incorrect.

## Structura Adresei – Cache 4-Way Set-Associative

Adresa de 32 de biți este împărțită în următoarele câmpuri:  
  
- Tag (21 biți): Identifică blocul de memorie univoc.  
- Index (5 biți): Selectează unul dintre cele 32 de seturi (2⁵ = 32).  
- Word Offset (2 biți): Selectează unul dintre cele 4 cuvinte (4 cuvinte/bloc).  
- Byte Offset (4 biți): Selectează un byte în cadrul unui cuvânt (4B × 2⁴ = 64B/bloc).  
  
Această împărțire permite adresarea precisă a fiecărui byte în cache și determinarea rapidă a poziției unei date într-un set și linie.

A diagram of a computer

AI-generated content may be incorrect.

# 5. Descrierea Modulelor

## 5.1 cache\_controller.v

Modulul central de control. Instanțiază FSM-ul și generează semnalele de control pentru operațiile cache.  
  
Funcții cheie:  
- Detectare hit/miss prin compararea tag-urilor  
- Gestionarea politicii LRU  
- Inițierea scrierii în memorie principală la evict dacă linia este dirty  
- Coordonarea încărcării datelor noi (read/write allocate)

A screen shot of a computer program

AI-generated content may be incorrect.

## 5.2 cache\_memory.v

Implementarea efectivă a memoriei cache.  
  
Structură internă:  
- Memorie pentru date și tag-uri (organizate pe seturi și căi)  
- Bit-uri valid și dirty  
- Vectori pentru politica LRU pe fiecare set  
  
Funcții cheie:  
- Citire/scriere în linia selectată  
- Selectare linie de victimă (cu prioritate pe invalid, altfel după LRU)  
- Actualizare LRU la access

A screen shot of a computer program

AI-generated content may be incorrect.

## 5.3 four\_way\_set.v

Organizează un set cache cu 4 căi. Selectează linia corespunzătoare pe baza adresei și controlează accesul sincronizat pe toate cele 4 căi.

A screen shot of a computer program

AI-generated content may be incorrect.

## 5.4 cache\_line.v

Reprezintă o linie individuală de cache – conține date, tag, semnale valid și dirty.

## 5.5 fsm\_structural.v

Modul top-level care leagă toate componentele într-o structură ierarhică. Este responsabil pentru instanțierea modulelor și conectarea semnalelor între ele.

## 5.6 main\_memory.v

Simulează o memorie principală simplificată. Include:  
- Array intern de date  
- Timp de răspuns simulat pentru acces  
- Interfață de citire/scriere utilizată de controller

## 5.7 cache\_controller\_tb.v

Testbench-ul pentru validarea funcționalității controllerului.  
  
Scenarii testate:  
- Citire READ\_HIT și READ\_MISS  
- Scriere WRITE\_HIT și WRITE\_MISS  
- Evacuare linie dirty (EVICT)  
- Verif icare actualizare LRU  
A computer screen shot of a program

AI-generated content may be incorrect.  
A close-up of a calculator

AI-generated content may be incorrect.A computer screen shot of a code

AI-generated content may be incorrect.

# 6. Probleme Tehnice și Soluții

- Implementarea politicii LRU: Provocarea principală a fost menținerea ordinii de acces pentru 4 căi pe fiecare set. S-a folosit un vector de priorități pe fiecare set pentru a determina rapid calea cea mai puțin recent utilizată.  
  
- Write-back corect: Controlul semnalului dirty a necesitat o logică atentă pentru a declanșa scrierea în memorie doar în starea EVICT, fără pierdere de date.

# 7. Rezultate și Concluzii

Simulările au confirmat comportamentul corect pentru toate operațiile de bază ale cache-ului:  
- Detectare hit/miss precisă  
- Politica LRU funcțională  
- Evacuări corecte ale liniilor modificate (dirty)  
- Performanță constantă în rularea testelor  
  
Proiectul respectă specificațiile inițiale și oferă o bază solidă pentru extinderi ulterioare precum cache multi-level sau optimizări energetice.