

Изследване на ефективността на кеш паметта при компютърни системи с архитектура симетричен мултипроцесор

1. Актуалност на проблема

Симетричните мултипроцесори са паралелни компютърни архитектури с обща памет, които притежават следните архитектурни аспекти:

- Симетрия.
- Едно адресно пространство.
- Кеширане.
- Кохерентност.
- Комуникацията се осъществява през паметта – ниска латентност, основана на прости инструкции load/store и информацията за кохерентността се генерира хардуерно.

Симетричните мултипроцесори се характеризират с унифициран достъп по общата памет (UMA – Unified Memory Access). Те се използват интензивно в комерсиалните приложения като бази данни и системи за он-лайн транзакции. От особена важност при симетричните мултипроцесори е всички процесори да имат равнопоставен достъп до общата памет, В/И устройства и услугите на операционната система.

Симетричността на тези системи дава възможност да се постигне висока степен на паралелизъм, което не е възможно при асиметричните (главен-подчинени) мултипроцесори. Главното ограничение се обуславя от централизираната обща памет, както и от общата шина или комутатора кросбар, които след като веднъж са изградени, трудно подлежат на модулно разширение.

Кеш паметите са високоскоростни буфери, които се разположени между процесора и основната памет и ускоряват обмена на данни между различните нива в йерархията на паметта. Ускоряването се постига чрез поддържане на копия от избрани части от данните върху носител с бързо действие, близко до това на горното ниво на паметта.

Особено важно при симетричните мултипроцесори е свойството кохерентност, което изисква една и съща информация да бъде идентична на различните нива от йерархията на паметта. Контрол на кохерентността се налага при запис на общи данни, миграцията на процеси и В/И операции в мултипроцесорни платформи.

Съществуват две стратегии за поддържане на кохерентността, *запис през кеша (write-through)*, където всяка операция за запис в кеша едновременно се повтаря за главната памет, като новите данни се записват едновременно в кеша и в главната памет и *запис при връщане на линията от кеша в основната памет (write-back)*, където в главната памет се

записват линиите, които подлежат на заместване с нови и в които е извършен запис, като тяхното съдържание е било променено.

Създадени са различни протоколи за поддържане на кохерентността като MSI, MESI, MOSI, MOESI, MERSI, MESIF, Synapse, Berkely, Firefly и Dragon, разделени в два класа:

- *Write – invalidate*: прави невалидни всички други кеширани копия, когато се модифицира локално кеширано копие;
- *Write – update*: изпраща (broadcasts) новото кеширано копие за актуализиране на всички други кеширани копия с един и същ адрес на линията.

Повечето симетрични мултипроцесори използват шина, като кохерентността на кешовете се осигурява от протокола MESI.

Протоколът MESI е известен още като Илинойски протокол. Той е широко използван протокол за поддържане на кохерентността в кеш памети. Най-популярният протокол, който поддържа стратегията „write-back“.

- *Модифицирано (M - Modified)* - линията на кеша е модифицирана поради осъществен запис в кеша.
- *Ексклузивно (E – Exclusive)* - съдържанието на линията в кеша е валидно, и невалидно във всички останали кешове. Съдържанието на паметта все още не е актуализирано.
- *Общо (S – Shared)* - линията е валидна в един или повече кешове или в паметта и всички копия са актуални.
- *Невалидно (I - Invalid)* - това е началното състояние след reset, или линията в кеша е станала невалидна от осъществен запис в кеша в друг кеш със същия адрес.

Кеш паметите, са обикновено от 5 до 10 пъти по-бързи от главната памет, те могат да намалят значително времето за достъп до паметта, ако внимателно се проектират. Успехът от използването на кеша зависи от характеристики на програмата и по-точно от нейната *степен на локалност*. Локалността на достъпите могат да бъдат:

- *Локалност във времето*, при нея има голяма вероятност да се извършат обръщения към определени позиции в паметта в близки бъдещи моменти.
- *Локалност в пространството*, при нея има голяма вероятност да се осъществи достъп до единици информация, чиито адреси са близки.
- *Локалност на последователността*, представлява отношението на изпълнението на последователност от инструкции и обработката на последователност от данни към това на последователни такива.

За да се използва локалността в пространството, информационният трансфер между кеша и главната памет включва не дума, а последователност от клетки, наричана блок или линия. За оптимална производителност, линията се предава между главната памет и кеша по

широка шина за данни. Броят на модулите памет се избират с оглед да се постигне оптимално съответствие между бързодействието на главната памет и това на кеша.

Ефективността на кеш паметите в симетричните мултипроцесори се определя от попаденията (hit rate) и липсите (miss rate) в кеша. Когато процесорът чете от паметта, първо се извършва обръщение към кеша. Ако търсената информация се съдържа в кеша, то тя се изпраща директно към процесора (попадение). Когато търсената информация не се съдържа в кеша (липса), се налага обръщение към главната памет за прехвърляне на данните в кеша, след което се налага отново процесорът да се обърне към кеш паметта. Времето за достъп се измерва в брой машинни цикли, като времетраенето на машинния цикъл е свързано с тактовата честота на системата.

2. Цел и задачи на експериментите

Целта на лабораторното упражнение е да изследва ефективността на кеш паметите в симетрични мултипроцесори, като се изучат и вземат под внимание архитектурните аспекти свързани със стратегиите и протоколите за поддържане на кохерентността на кеш паметите в симетричните мултипроцесори и принципите на локалността. На база на архитектурните конфигурации, описани в табл. 1.1, да се направи оценка на влиянието на размера на кеш паметта и броя на процесорите върху трафика по шината и липсите (miss rate) в кеша.

Таблица 1.1: Архитектурни конфигурации за изследване на кеш паметта в симетрични мултипроцесори

Параметър	Стойност
Процесори в симетричния мултипроцесор	1, 2, 4, 8
Протокол за кохерентност на кеша	MESI
Схема за арбитраж по шината	LRU
Дължина на думата	16
Думи в блок	32
Блокове в главната памет	524288 (= 32 MB)
Метод на съответствие (Mapping)	Set-Associative
Размер на кеша	Варира в зависимост от блоковете в кеша.
Блокове в кеша	16 (= 1 KB), 64, 256 и 2048 (= 128 KB)
„Memory trace“	COMP, WAVE, EAR

За постигането на целта е необходимо да се решат експериментално следните задачи:

1. Конфигуриране и симулации в съответствие с архитектурните конфигурации на симетричните мултипроцесори зададени в таб. 1.1.
2. Оценка на влиянието на размера на кеш паметта върху:
 - Трафик по шината.
 - Липсите (miss rate).
3. Оценка на влиянието на броя на процесорите върху:
 - Трафик по шината.
 - Липсите (miss rate).
4. Графично представяне на еволюцията на кеш паметта за всяка от изследваните архитектурни конфигурации.

3. Експериментална платформа

За провеждане на поставените задачи и експериментите в настоящето лабораторно упражнение ще се използва симулаторът SMPCache. Това е симулатор за изучаване и анализиране на кеш паметта в симетричните мултипроцесори с удобен графичен интерфейс, който работи на компютърни системи с операционна система MS Windows.

С този симулатор могат да се проведат експерименти и да се изучат принципите на локалността, влиянието на броя на процесорите, протоколите за кохерентност на кеша, схемите за арбитраж, методите на съответствие, политики на заместване и да се отчете влиянието на размера на кеша (броя на блоковете в кеш), броя на думите в блок (размера на блока в оперативната памет), дължина на думата и броя на процесорите върху ефективността на кеш паметта в компютърни системи с архитектура симетричен мултипроцесор.

4. Методика за провеждане на експериментите

За решаване на поставените задачи и реализиране на експериментите по зададените архитектурни конфигурации от табл. 1.1 е необходимо да се направи конфигуриране, настройка и тестване, като се създадат файлове с разширение “.cfg” за всяка архитектурна конфигурация. Възможните стойности на всеки параметър има свой код, показани в табл. 1.2.

Таблица 1.2: Възможни стойности и кодове на параметрите за изследване на кеш паметта в симетричните мултипроцесори

Параметър	Възможна стойност	Код
Протокол за кохерентност на кеша (Cache coherence protocol)	MSI	1
	MESI	2
	DRAGON	3
Схема за арбитраж по шината (Scheme for bus arbitration)	Random	1
	LRU	2
	LFU	3

Метод на съответствие (Mapping)	Direct	1
	Set-associative	2
	Fully-associative	3
Броя на групите в кеш (Cache sets)	0	NO
	1, 2, 4, ..., 2048	1, 2, 4, ..., 2048
Политики на заместване (Replacement policy)	NO	1
	Random	2
	LRU	3
	FIFO	4
	LFU	
Стратегии за запис (Writing strategy)	Write-through	1
	Write-back	2

Във всеки един от конфигурационните файлове могат да се запишат само по една възможност за стратегиите на запис (Write-back) и за нивата на кешовете (Level 1). Това позволява записаните файлове с различни архитектурни конфигурации да бъдат преизползвани за изготвяне на сравнителен анализ на получените експериментални данни.

За провеждане на изследванията е необходимо да се използват и файлове с разширение “.prg”, т. е. файлове с адресите от оперативната памет, към които ще се обръщат процесорите по време на изпълнение на дадена програма. Това са така наречените трейсове. Тези трейс файлове се състоят от редове, всеки от които има два номера: етикет (label) и стойност (value).

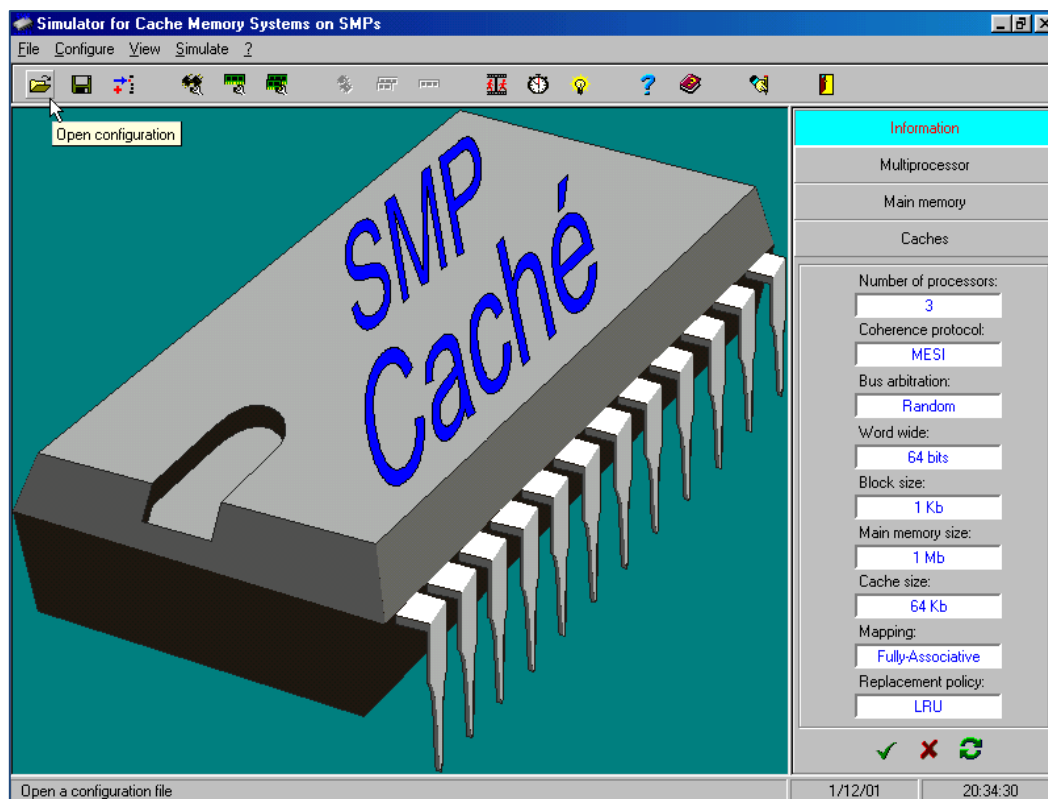
Етикетът представлява десетичен номер, който идентифицира типа на операцията за обръщение на процесора към адрес от оперативната памет в даден момент според програмата: *за извличане на инструкция (0)*, *за четене на данни от оперативната памет (2)*, *за запис на данни в оперативната памет (3)*. Стойността представлява номер, който показва адреса на думата от оперативната памет, към която ще се обърне процесора. На фиг. 1.1 е показан примерен трейс файл.

0	00001b08
0	00001ca5
2	00007951
0	00001d04
0	00001eb8
2	00007952
0	0000201c
2	00007c71
0	0000201f
3	00007b51

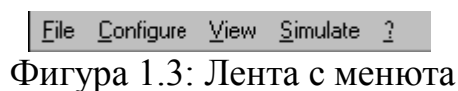
В примерния трейс файл се наблюдават шест обръщения към паметта за извличане на инструкция, три обръщения за четене на информация от оперативната памет и едно обръщение за запис. Общо десет обръщения към оперативната памет.

Фигура 1.1: Примерен трейс файл

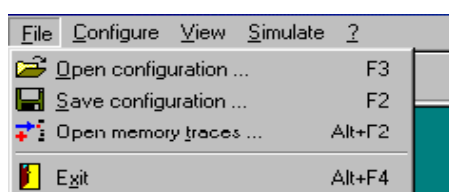
Конфигурирането, настройката и тестване в среда на SMP Cache е възможно посредством *лента с менюта*, *лента с инструменти*, *конфигурационен панел* и *статус бар*.



Фигура 1.2: SMPCache интерфейс



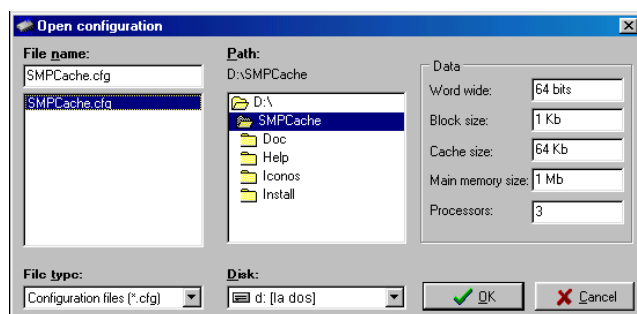
Фигура 1.3: Лента с менюта



Фигура 1.4: Бутон „Файл“

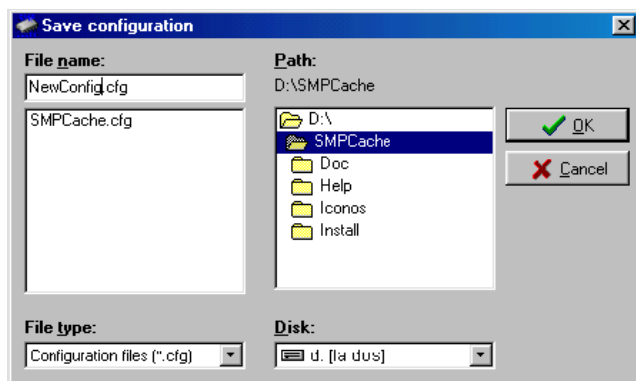
Лентата с менюта съдържа различни команди: *Файл*, *Конфигуриране*, *Изглед*, *Симулиране* и *Помощ*.

Това меню съдържа следните команди: *Отваряне* на конфигурация, *Запис* на конфигурация, *Отваряне* на трайс файл и *Изход*.



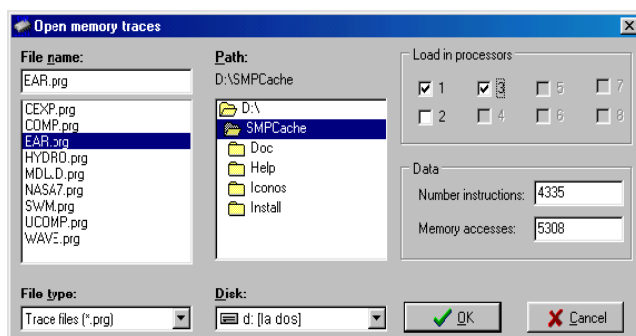
Фигура 1.5: Бутон „Отвори конфигурация“

Тази команда позволява да се отвори файл с разширение (".cfg"), с параметрите на желаната компютърна конфигурация.



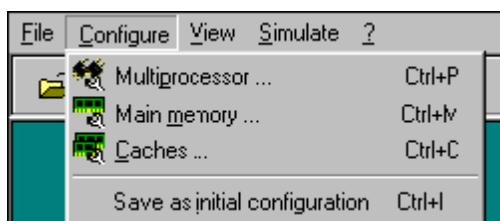
Фигура 1.6: Бутон „Съхрани конфигурация“

Тази команда позволява да се съхрани файл с разширение (“.cfg”), за да може текущата конфигурация да се използва за провеждане на изследванията в бъдеще.



Фигура 1.7: Бутон „Отваряне на трайс файл“

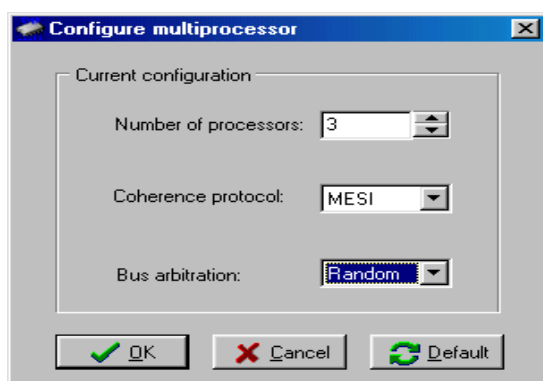
Тази команда позволява да се отвори файл с разширение (“.prg”), който трябва да се зареди в някои от активните процесори.



Фигура 1.8: Бутон „Конфигуриране“

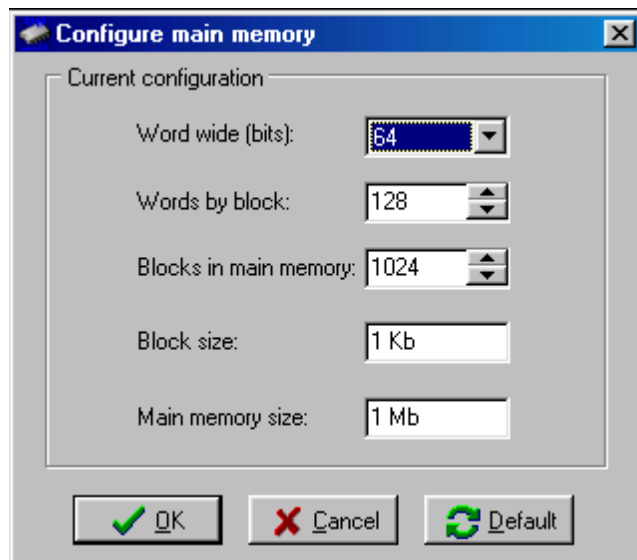
Този бутон съдържа следните команди за конфигуриране на желаната архитектура на симетричен мултипроцесор:

Мултипроцесор (Multiprocessor), Памет (Main memory), Кеш (Caches) и съхраняване на първоначалната конфигурация.



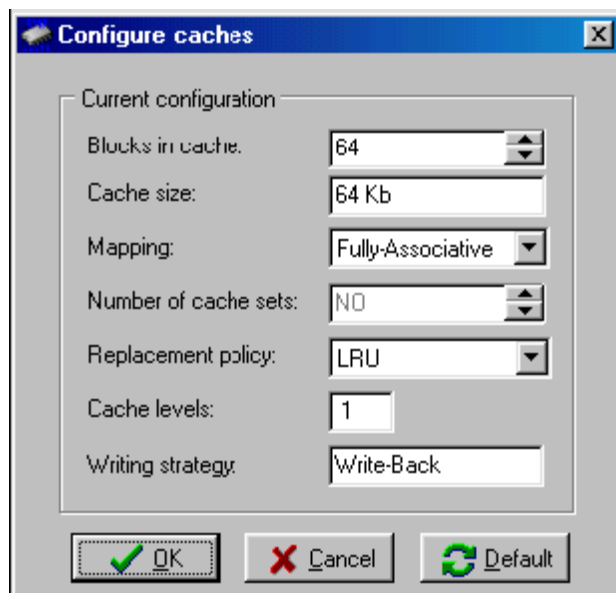
Фигура 1.9: Бутон „Конфигуриране на мултипроцесор“

Тази команда позволява да се направи нова конфигурация включваща брой процесори, протокол за кохерентност и арбитраж по шината.



Тази команда позволява да се направи нова конфигурация на паметта. Параметрите, които трябва да се зададат са стойности са: *дължина на думата, брой думи в блок, брой блокове в оперативната памет, размер на блока и размер на оперативната памет.*

Фигура 1.10: Бутон „Конфигуриране на главната памет“



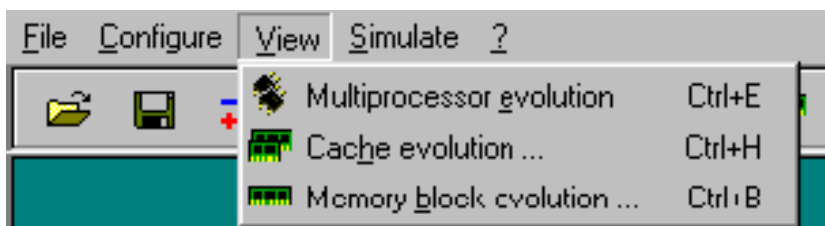
Тази команда позволява да се направи конфигурация на кешовете свързани с всеки процесор. Всички кешове ще имат една и съща конфигурация. Параметрите вързани с кешовете са: *брой блокове, размер на кеша, метод на съответствие, брой групи в кеш (при групово-асоциативните кешове), политика на заместване, ниво на кеша, стратегия за запис.*

Фигура 1.11: Бутон „Конфигуриране на кеш паметта“

В симулатора SMPCache има три варианта за протичане на симулацията:

- *Step by step* – ако се избере тази команда симулацията се спира след всяко обръщение към оперативната памет. Този вид симулация е по подразбиране.
- *With breakpoint* – ако се избере тази команда симулацията се спира след извършване на определен брой обръщения към оперативната памет.
- *Complete execution* – ако се избере тази команда симулацията се спира след извършване на всички обръщения към паметта (when all the memory traces are finished).

За да се получи изглед на направената конфигурация е необходимо да бъде зареден поне един трейс файл в някой от процесорите на симетричния мултипроцесор.



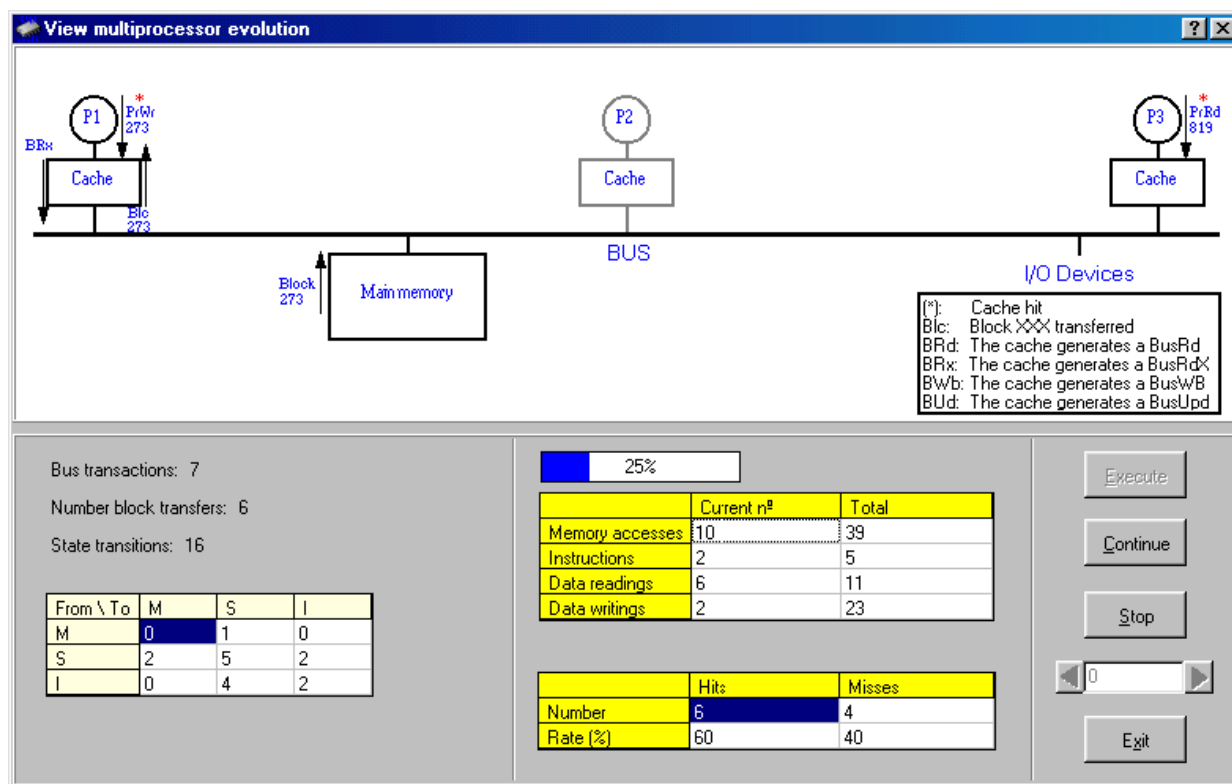
Фигура 1.12: Визуализация на изпълнението

Първият вариант на визуализация на изпълнението (Multiprocessor evolution) позволява да бъде направен глобален поглед върху развитието на мултипроцесорната система в зависимост от направената конфигурация. Симулаторът показва как реагира системата (responds) на обръщанията към оперативната памет генерирани от програмите (трейс файловете, по време на симулацията).

Само процесорите, в които е зареден трейс файл влияят на симулацията. При избирането на командата *Multiprocessor evolution* се отваря прозорец, на който е показана диаграма на симетричния мултипроцесор. На тази диаграма са показани трансфера по шината (BusRd, BusRdX, BusWB and BusUpd), инициализиране на запитвания от страна на процесора (PrRd XXX and PrWr XXX) и трансфера на блоковете (Block XXX). Със стрелка е указана посоката на трансфера по шината, запитванията на процесора или трансфера на блокове.

Трансферът по шината винаги са от даден кеш към шината. Когато процесорът има някакво запитване, той се обръща към неговия кеш. Да предположим, че процесорът инициализира две неща: четене (PrRd) и запис (PrWr). Извличането на инструкции и четенето на данни се считат за PrRd искане. Четене или запис може да се извършва в блок от оперативната памет, като не е задължително този блок да се намира и в кеш.

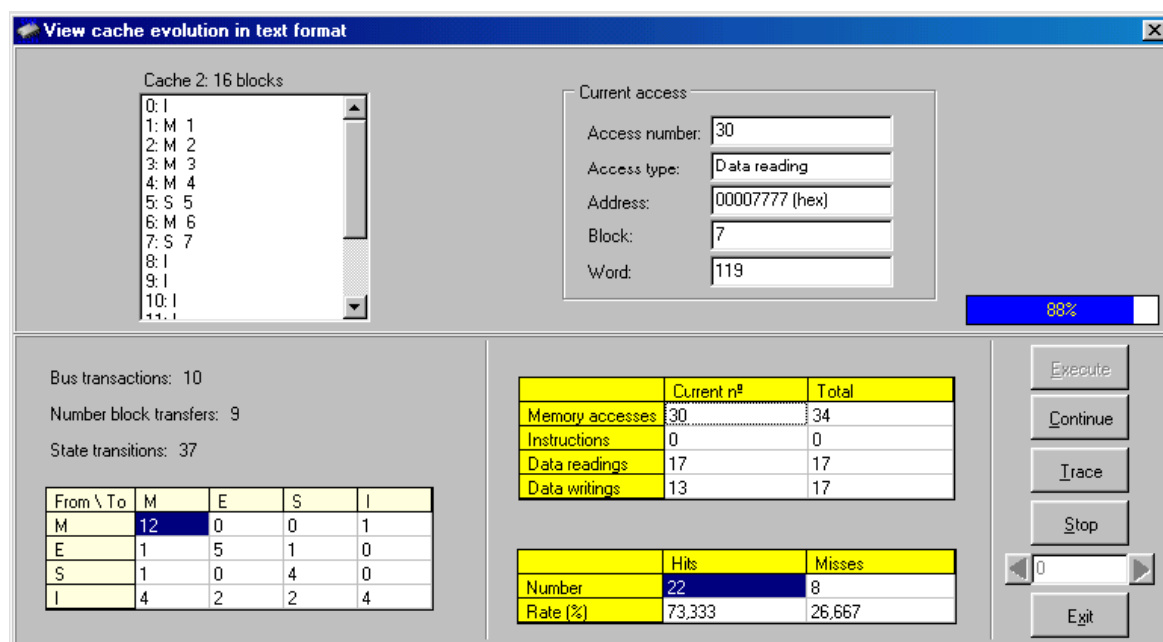
При този тип визуализация се показват също *броя на блоковете трансферирани по шината, общия брой обръщания към паметта, типа на обръщание (дали е за извличане на инструкция, за четене на данни или за запис на данни), броя на попаденията и липсите (степената на попадения и степената на липсите).*



Фигура 1.13: Визуализация на изпълнението „Multiprocessor evolution“

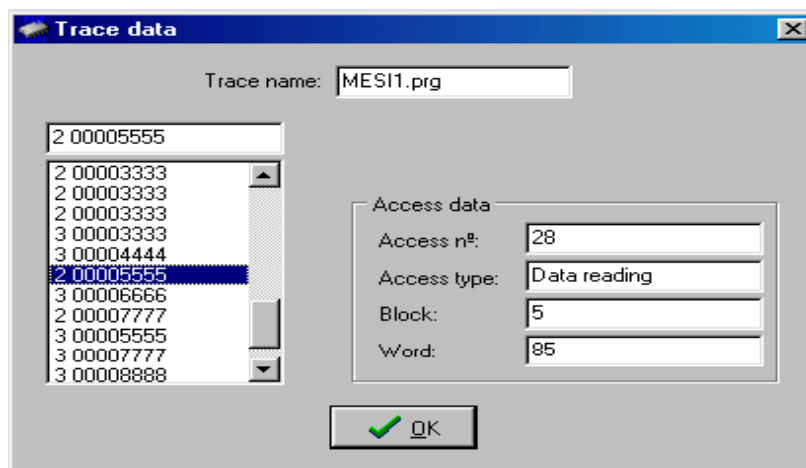
Вторият вариант на визуализация на изпълнението (Cache Evolution) позволява да се наблюдава развитието на конкретен кеш според направената конфигурация и обръщенията към паметта, които се генерират от програмите (трейс файловете).

Избира се кешът, който трябва да се изследва и начина, по който да бъдат показани резултатите (текстово или графично). Ако се избере текстов формат на представяне на данните се отваря следния прозорец, фиг. 1.14.



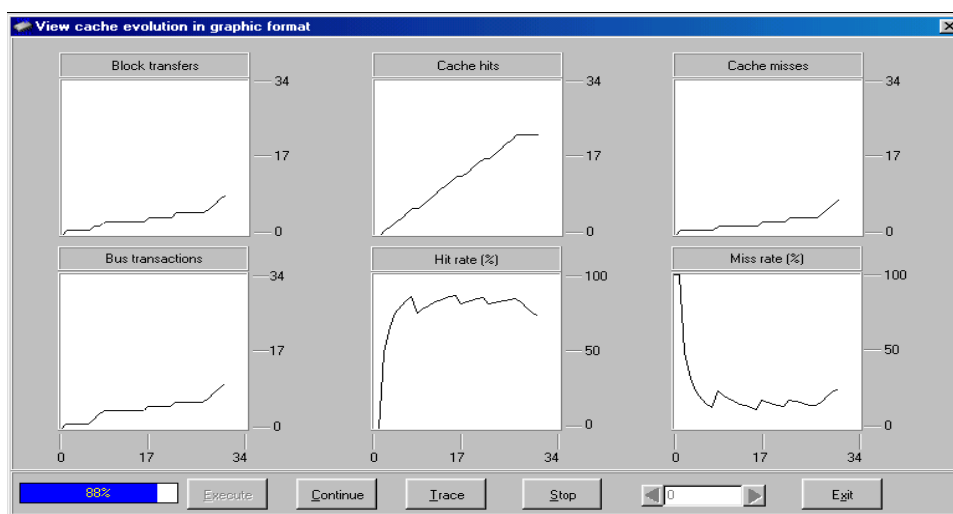
Фигура 1.14: Визуализация на изпълнението „Cache Evolution“

Информацията, която може да бъде получена е номера на кеша (процесора), броя на блоковете в кеш, състоянието на всеки блок от кеш и др. От тази диалогова кутия може да се получи подробна информация за трейс файла, асоцииран със съответния кеш (процесор): име на трейс файла, обръщенията към паметта по време на изпълнение на този трейс.



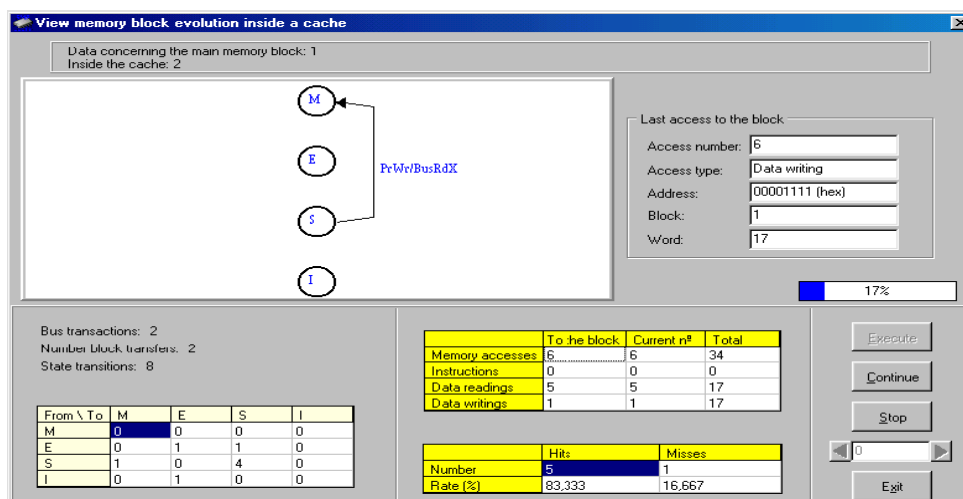
Фигура 1.15: Данни за трайс файла

Ако се избере графичен формат за представяне на данните резултатите показват броя на трансферираните блокове по шината, броя на транзакциите по шината генерирани от този кеш, броя на попаденията и липсите, както и степента на попадение и степента на липсите, фиг. 1.16.



Фигура 1.16: Изследването на кеша в графичен формат

Последният вариант на визуализация на изпълнението (Memory block evolution) позволява да се наблюдава конкретен блок от конкретен кеш. Може да се изберете блока от оперативната памет, който трябва да се изследва.



Фигура 1.17: Изглед на еволюцията на блоковете памет в кеша

5. Експериментални резултати и оформяне на протокол

На база на проведените експериментални изследвания по указаните входни данни и архитектурни конфигурации описани в табл. 1.1:

- Да се изследват и да се анализират стратегиите и протоколите за кохерентност на кешовете памети в симетрични мултипроцесори.
- Да се оцени влиянието на размера на кеш паметите и броя на процесори върху трафика по шината и липсите (miss rate) в кеша на симетричните мултипроцесори.
- Да се представи в графичен формат еволюцията на кешовете за всяка конфигурация и да се направи анализ на получените резултати.
- Получените от експериментите резултати да бъдат представени в протокол и дискутирани в рамките на лабораторното упражнение.