УДК 04.07:04.08:04.25:04.33 DOI: 10.31673/2786-8362.2024.023081

Целованський Т.Р., Шикула О.М., д.ф.-м.н.

ДОСЛІДЖЕННЯ МЕТОДІВ ЗБІЛЬШЕННЯ ШВИДКОДІЇ ПІДСИСТЕМИ ПАМ'ЯТІ У ЕЛЕКТРОННО ОБЧИСЛЮВАЛЬНИХ МАШИНАХ

Tselovanskyi T.R., Shykula O.M. Research of methods of increasing the performance of the memory subsystem in electronic computing machines. This article explores methods of optimizing the interaction between the processor and the random access memory (RAM) to improve the performance of computer systems. The main parameters of DDR4 memory timings according to the JEDEC specification are studied, as well as the influence of these parameters on system speed. The interdependence of internal parameters of memory delays and their influence on each other are investigated. Recommendations for setting primary and secondary RAM timings are presented, as well as features of multi-channel configurations that allow achieving optimal memory performance for scientific computing, machine learning, and other resource-intensive tasks. The main optimization methods, key features and impact on performance are considered. The main programs for testing the stability of the system when performing performance optimization actions are specified, a theoretical presentation of key information for further in-depth research of the topic is made, and the implementation of optimization methods for improving performance when working on a computer and performing tasks that require high performance of the system. Systematization of information about the latency at the hardware level, in the central processor and the interaction of the built-in memory controller with the RAM.

Keywords: RAM, JEDEC, DRAM, DDR, memory latency, CPU cache, timing optimization, methods of memory optimization, memory management

Целованський Т.Р., Шикула О.М. Дослідження методів збільшення швидкодії підсистеми пам'яті у електронно обчислювальних машинах. У даній статті розглядаються методи оптимізації взаємодії між процесором та оперативною пам'яттю (ОЗП) для підвищення продуктивності комп'ютерних систем. Досліджуються основні параметри таймінгів пам'яті DDR4 згідно зі специфікацією JEDEC, а також вплив цих параметрів на швидкодію системи. Досліджується взаємозалежність внутрішніх параметрів затримок пам'яті та їх вплив між собою. Представлено рекомендації щодо налаштування первинних та вторинних таймінгів ОЗП, а також особливості багатоканальних конфігурацій, які дозволяють досягти оптимальної швидкодії пам'яті для наукових обчислень, машинного навчання та інших ресурсомістких завдань. Розглянуті основні методи оптимізації, ключові особливості та вплив на швидкодію. Зазначено основні програми для тестування стабільності системи при виконанні дій з оптимізації швидкодії, зроблена теоретична викладка ключової інформації для подальших поглиблених досліджень теми, та впровадження методів оптимізації для покращення швидкодії при роботі на ЕОМ та виконанні задач, що потребують високої швидкодії системи. Проведено систематизацію інформації про затримки на хардверному рівні, всередині центрального процесору та взаємодії внутрішнього контролеру пам'яті з оперативними запам'ятовуючими пристроями.

Ключові слова: RAM, JEDEC, DRAM, DDR, латентність пам'яті, кеш-пам'ять процесора, оптимізація затримок, таймінги ОЗП, методи оптимізації пам'яті, memory management

Вступ

У сучасних обчислювальних системах ефективна взаємодія процесора, оперативної пам'яті (ОЗП) та підсистеми постійної пам'яті (SSD, HDD) є критично важливою для досягнення високої продуктивності. Пам'ять впливає на швидкодію виконання багатьох інтенсивних завдань, включаючи наукові розрахунки, аналіз Big Data, віртуалізацію, обробку графіки та машинне навчання. Час доступу до даних і швидкість їх передачі можуть суттєво обмежувати швидкодію системи, що є серйозною проблемою для сучасних задач, що потребують складних розрахунків з великими обсягами даних.

Існує низка методів щодо покращення взаємодії між ОЗП, кешем процесора та постійною пам'яттю. Зокрема предметом розгляду є підвищення швидкодії через оптимізацію таймінгів ОЗП, використання багатоканальних систем пам'яті, а також застосування адаптивних методів управління таймінгами на рівні програмного забезпечення. Вивчення особливості взаємодії компонентів підсистеми пам'яті, такі як багатошаровий кеш процесора (L1, L2, L3), а також технологій на зразок DDR3, DDR4 та DDR5 із зниженими затримками, мають фундаментальну

значимість при дослідженні швидкодії пам’яті [1, 2].

Велика кількість аспектів оптимізації залишаються частково недослідженими, особливо у випадках, коли дослідження шляхом реверс-інженіринга архітектури є неможливою. Залишаються актуальними питання, пов'язані з програмними методами оптимізації параметрів пам'яті, покращенням взаємодії кешу з ОЗП і підсистемою постійної пам'яті, а також дослідженням впливу таймінгів пам'яті на продуктивність системи.

Аналіз останніх досліджень. Питання підвищення продуктивності ОЗП через оптимізацію її параметрів активно досліджується у сфері комп'ютерних наук і обчислювальної техніки[5]. Попередні дослідження свідчать про те, що правильне налаштування таймінгів пам'яті та підвищення її частоти може суттєво знизити затримки та збільшити пропускну здатність. Багатоканальна пам’ять та багаторангові модулі, також демонструють підвищення продуктивності у завданнях, що потребують великих обсягів даних, оскільки такі конфігурації забезпечують більш ефективне використання пропускної здатності контролера пам'яті [9]. У той же час, питання стабільності системи при агресивних налаштуваннях ОЗП залишаються недостатньо дослідженими, особливо для сучасних стандартів DDR4 і новітнього DDR5 [1, 3].

Постановка завдання. Основною задачею цього дослідження є виявлення і оптимізація ключових параметрів оперативної пам’яті для зменшення затримок і збільшення пропускної здатності. Це включає аналіз та налаштування первинних і вторинних таймінгів пам'яті DDR4, оцінку впливу частоти та таймінгів на швидкість передачі даних, а також розгляд особливостей багатоканальних та рангових конфігурацій пам’яті. Дослідження передбачає теоретичне та експериментальне вивчення цих параметрів з метою отримання рекомендацій щодо їх оптимального налаштування для забезпечення стабільної та високопродуктивної роботи системи.

Метою роботи є визначення оптимальних способів та інструментів для налаштування частоти і таймінгів оперативної пам'яті DDR4, що дозволять досягти максимальної пропускної здатності при збереженні стабільності системи. Робота має на меті аналіз первинних та вторинних таймінгів ОЗП згідно зі специфікацією JEDEC для DDR4, виявлення та систематизацію залежностей між частотою пам'яті та затримками при різних конфігураціях таймінгів, оцінку впливу багатоканальних конфігурацій та багаторангових модулів на продуктивність

Виклад основного матеріалу дослідження.

Методи оцінки швидкодії пам’яті. Для оцінки впливу параметрів пам’яті на продуктивність а також при налаштуванні пам’яті існує ряд бенчмарків, які дозволяють протестувати швидкість ОЗП та стабільність використаних параметрів при налаштуванні (особливо в випадку втручання у функціонування ОЗП на хардверному рівні) у різних аспектах.

AIDA64 Memory Benchmark є одним з найбільш репрезентативних та найпопулярніших тестів для ОЗП, який вимірює пропускну здатність пам'яті (швидкість читання, запису та копіювання) і латентність, а отриманий результат представлено в вигляді точних цифр, а не системи балів. AIDA64 надає детальну інформацію про продуктивність пам’яті в різних режимах, що дозволяє зрозуміти, як параметри, такі як таймінги, впливають на загальну швидкодію. Geekbench Memory Test виділяється тим, що проводить серію тестів для оцінки загальної продуктивності системи, в тому числі тестує швидкість ОЗП. Він вимірює, як пам'ять працює у різних типах навантажень, наприклад, при обробці великих блоків даних, що допомагає оцінити продуктивність пам’яті для реальних завдань, до недоліків можна віднести тільки використання бальної системи оцінки продуктивності. PassMark PerformanceTest – Memory Mark тестує продуктивність пам'яті, зокрема швидкість операцій читання, запису і доступу до даних. Цей бенчмарк також дозволяє порівняти продуктивність пам’яті різних систем, що може бути корисним при зміні таймінгів або частоти роботи ОЗП. SiSoftware Sandra Memory Benchmark проводить кілька тестів для оцінки продуктивності пам'яті, таких як пропускна здатність для читання, запису і копіювання, а також латентність. Sandra також

підтримує тестування багатоканальної пам'яті, що може бути корисним для систем з кількома модулями пам'яті. LinX і Intel Memory Latency Checker (MLC) дозволяють вимірювати латентність пам’яті під високим навантаженням, що особливо важливо для наукових розрахунків і великих обчислювальних завдань. Intel MLC спеціально розроблений для тестування латентності та пропускної здатності у системах з великою кількістю процесорних ядер, що корисно для серверних систем. MemTest86є однією з найпопулярніших програм для діагностики та тестування стабільності ОЗП, що працює автономно, завантажуючись з флеш-накопичувача (з-під BIOS), що дозволяє перевіряти пам'ять без впливу ОС. Виконує низку алгоритмів тестування для виявлення помилок, зокрема помилок внаслідок нестабільного розгону або занижених таймінгів та ідеально підходить для глибокого тестування з повним охопленням пам'яті. TestMem5 це Інструмент для тестування стабільності ОЗП у Windows, що пропонує різні попередньо налаштовані профілі тестування, які адаптовані для перевірки розгону і стабільності пам'яті. Може проводити тривалі стрес-тести з високою інтенсивністю для виявлення навіть дрібних помилок Майже повністю дублює функціонал MemTest86 та виступає в ролі взаємозамінного аналогу.

Основні аспекти взаємодії кешу процесора, оперативної пам'яті та підсистеми постійної пам'яті. Сучасні системи пам'яті організовані як багаторівнева ієрархія, що складається з кешу процесора, оперативної пам'яті та підсистеми постійної пам'яті. Кожен з цих компонентів має різні характеристики швидкості, обсягу та енергоспоживання, ієрархія системи пам’яті усередненого вигляду, в залежності від швидкості доступу та часових затримок.

Кеш процесора (L1, L2, L3) має мінімальну латентність і найвищу швидкодію, але обмежений обсягом. Він зберігає найбільш часто використовувані дані. Кеш також поділяється на три основні кластери що відрізняються за розташуванням на кристалі, обсягом та швидкістю доступу. Усереднено кожен з рівнів має наступні параметри об’єму та швидкості доступу: L1 кеш має латентність близько 1-2 наносекунд (ns) є найшвидшим, але має обмежений обсяг (зазвичай 32–64 КБ на ядро), таким чином він забезпечує миттєвий доступ до найчастіше використовуваних та зачасту найбільш низькорівневих команд та даних. L2 кеш має латентність близько 3-10 наносекунд (ns) L2 кеш трохи повільніший за L1, але має більший обсяг (256 КБ – 1 МБ на ядро), та підходить для зберігання більших за обсягом команд, яким не так критична швидкість відповіді. L3 кеш має латентність в межах 10-20 наносекунд (ns), є спільним для всіх ядер у багатоядерних процесорах і має більший обсяг (від 4 МБ до 64 МБ для сучасних моделей). Також Intel тестували використання L4 кешу, в поколінні Broadwell, що на папері давало великий обсяг (128 мегабайт кешу), з швидкою пам’яттю та не отримали розповсюдження в зв’язку собівартістю при виробництві, та недостатньо низькими затримками доступу, що сягали близько 40 наносекунд [5, 8].

В завершенні теми кешу, варто зазначити, що швидкість кешу та затримки звернення напряму залежать від того як близько до ядер знаходиться модуль кешу на літографії. В рамках однієї архітектури та техпроцесу, швидкість взаємодії кеша та процесору залежить переважно від одного єдиного параметру – частоти CPU-NB (північного мосту) який перемістився з материнської плати всередину процесора уже майже 20 років тому, та є зв’язуючим елементом взаємодії материнської плати з процесором, та підсистемою пам’яті. Тобто маніпуляції пов’язані з частотою CPU-NB та його напругою, напряму впливають на швидкість обміну даними по між’ядерними зв’язками, та пропускну здатність кільцевої шини або мешу (типів зв’язку між ядрами) [1, 8].

Оперативна пам'ять (DRAM) є повільнішою за кеш, але має більший обсяг. Вона зберігає дані та команди для активних процесів. Латентність доступу до оперативної пам'яті залежить від типу пам’яті (DDR3 або DDR4), її частоти, а також налаштувань таймінгів. Зазвичай виробники встановлюють стандартні таймінги згідно специфікацій JEDEC для різних частот, що визначає середній час доступу. Середня латентність доступу всередині модулів пам’яті для DDR3приблизно 10–15 наносекунд (ns), для DDR4 приблизно 12–18 ns, ця різниця пов’язана з різними налаштуваннями CAS Latency (tCL) і частотами, характерними для кожного типу

пам’яті. Пропускна здатність оперативної пам’яті обчислюється на основі її тактової частоти та ширини шини пам'яті. DDR-пам'ять здійснює дві передачі даних за один такт, що подвоює пропускну здатність [5, 9]. Формула для обчислення пропускної здатності (швидкості) оперативної пам'яті виглядає наступним чином:

Пропускна здатність (МБ/с)=Частота (МГц)×8×2; де множник 8 — це ширина шини (64 біти) у байтах, а 2 враховує дві передачі за такт (формула має спрощений вигляд, без урахування особливостей внутрішніх налаштувань затримок, враховуючи характеристики згідно стандартних специфікацій).

Таким чином середня швидкість планки пам’яті без урахування особливостей архітектури для DDR3 та DDR4 в залежності від частоти виглядає наступним чином:

DDR3 (ширина шини – 64 біти, множник 2 для DDR):

- DDR3-1600 (1600 МГц): ~12,800 МБ/с
- DDR3-1866 (1866 МГц): ~14,928 МБ/с
- DDR3-2133 (2133 МГц): ~17,064 МБ/с

DDR4 (ширина шини — 64 біти, множник 2 для DDR):

- DDR4-2133 (2133 МГц): ~17,064 МБ/с
- DDR4-2400 (2400 МГц): ~19,200 МБ/с
- DDR4-2666 (2666 МГц): ~21,328 МБ/с
- DDR4-3200 (3200 МГц): ~25,600 МБ/с

Що корелює з даними тестування ОЗП на реальних комп'ютерах.

Також при прогнозуванні планованої швидкодії комп'ютера слід враховувати тип модулів ОЗП що буде встановлено в системі (чотирьохрангову дворангову чи однорангову), існує багато фундаментальних відмінностей між ними, що напряму впливають на роботу системи, в рамках даної статті нас цікавлять лише основні відмінності, що впливають на кінцевий результат.

Переважна кількість модулів пам’яті є одноранговими (single-rank) та двуранговими (dual-rank), де ранг представляє собою окремий блок комірок пам'яті, до якого можна звертатися незалежно від іншого, кожен з них має свої особливості при використанні. Однорангові модулі мають меншу кількість чипів і зазвичай мають нижчу затримку, але обмежений обсяг. Вони працюють із меншою затримкою, оскільки немає перемикання між рангами. Дворангові модулі мають більшу кількість чипів та обсяг, і дозволяють контролеру пам’яті працювати ефективніше, оскільки в момент затримки на одному ранзі можна ініціювати доступ до іншого. Це може призвести до підвищення пропускної здатності системи на 5-10% у порівнянні з одноранговими модулями за умови роботи в багатозадачному режимі. Таким чином, вибір між одноранговими і дворанговими модулями пам’яті залежить від специфіки завдань. У додатках, що потребують низької затримки, однорангові модулі можуть бути кращими, тоді як для інтенсивних навантажень дворангові модулі забезпечать більшу ефективність завдяки підвищеній пропускній здатності [5, 9].

В доповнення варто відмітити, що при розгляді затримки звернення до ОЗП, було вказано внутрішні затримки модулів пам'яті, в той час як реальні затримки обміну інформації значно відрізняються. Під час тестування, наприклад, через AIDA64, ми бачимо значно більшу латентність доступу до оперативної пам’яті, яка складає 40–80 наносекунд. Це розходження з теоретичними 10–18 наносекундами для DDR3/DDR4 виникає з кількох причин, пов'язаних з особливостями архітектури та принципами вимірювання латентності.

Теоретичні значення латентності (10–18 нс) часто враховують лише час виконання однієї операції (наприклад, затримку при виконанні CAS Latency (tCL) або Row-to-Column Delay (tRCD)). Однак фактична латентність доступу до пам’яті включає кілька додаткових затримок, таких як Затримка вибору рядка (tRCD), Затримка передзарядки (tRP), Затримка оновлення (tRFC). Повний цикл доступу до нових комірок або нового рядка пам’яті може включати всі ці етапи, значно збільшуючи сумарну латентність. Таким чином, тест в AIDA64 відображає сукупність усіх затримок, пов’язаних з доступом до ОЗП, а не лише окремий параметр, який вказано в специфікаціях, та випливає з теоретичних відомостей.

Також значний вплив вказує контролер пам’яті (CPU-NB), який інтегрований у процесор, обробляє запити доступу до пам'яті та має свої додаткові затримки. Контролер пам'яті в процесорі організовує звернення до банків та ранжування пам'яті, що додає певну латентність. Це особливо важливо в багатоканальних системах, де контролер координує запити між кількома модулями пам'яті а не тільки всередині модуля між банками. Кожен запит до пам'яті може складатися з різних етапів доступу. Наприклад, якщо рядок вже активний, доступ може бути швидшим, але якщо рядок потрібно передзарядити, час доступу збільшується. В AIDA64 вимірюється загальний час, включаючи всі можливі затримки, необхідні для вибірки даних.

Налаштовані таймінги та відсутність помилок при передачі даних між компонентами забезпечують високу продуктивність. Якщо дані не знайдені в кеші процесора (cache miss), система звертається до ОЗП. Якщо ж і в ОЗП даних немає, звернення здійснюється до підсистеми постійної пам'яті, що значно збільшує час доступу до даних. Навіть якщо оперативна пам'ять сама по собі здатна працювати на низьких затримках, існують також додаткові затримки між пам'яттю та процесором, які виникають при передачі даних через шину. Інтерфейси DDR4, особливо на високих частотах, можуть вносити додаткові тактові цикли на кожен доступ до пам'яті, щоб забезпечити синхронізацію і стабільність передачі даних.

Сучасні процесори застосовують оптимізацію доступу до пам’яті, такі як prefetching (попереднє завантаження) і out-of-order execution (виконання поза порядком). Це дозволяє «приховувати» деякі затримки за рахунок паралельного виконання команд і вільного вибору команд для виконання в процесі очікування доступу до пам'яті. Проте при тестуванні в AIDA64 вимірюється «сирий» час доступу, що не включає ці оптимізації, а показує фактичну тривалість очікування у середньому.

Через важливість швидкодії пам'яті та бажання зменшити затримки при зверненні до ОЗП, існують методи покращення взаємодії процесора та оперативної пам'яті без втручання в архітектуру, до найбільш поширених можна віднести:

– Оптимізацію первинних та вторинних таймінгів пам'яті: Таймінги ОЗП, такі як tCL, tRCD, tRP, та tRAS, що відносяться до первинних, мають великий вплив на продуктивність системи. Зменшення цих таймінгів гарантовано зменшує латентність пам'яті, та є найбілш дослідженим та простим, але водночас дієвим. В той час як налаштування вторинних таймінгів можуть нести за собою більші ризики, та вище шанси нестабільності системи.

– Використання профілів XMP (Extreme Memory Profile): Більшість модулів DDR4 з заводу підтримують XMP, що дозволяє автоматично налаштовувати оптимальні параметри пам'яті для підвищення швидкодії. Даний метод є найбільш простим, в зв’язку з тим що налаштування відбувається в декілька кліків в біосі. До недоліків слід віднести поганий контроль якості на заводах, коли не всі модулі пам’яті що мають прошитий профіль XMP стабільно себе поводять на тих налаштуваннях. Також зміна профілю супроводжується суттєвим підвищенням напруги на модулях пам’яті (1.2 згідно специфікацій JEDEC, можуть бути підвищеними в профілі XMP до 1.5 вольт), що може призвести до завчасної деградації комірок. Також цей метод не завжди може бути використаний, через несумісність або обмеження на великій кількості материнських плат, та обмеженні процесорів на максимальну швидкість сумісної пам’яті.

– Програмна оптимізація обробки пам'яті: Ефективне використання кешу та попереднє завантаження (prefetching) даних зменшує кількість звернень до ОЗП, що покращує продуктивність. Недоліком цього методу є високорівневість, виконання на софтверному рівні, та відсутність впливу на залізо (хардверному рівні)

– Використання багатоканальної пам'яті: Багатоканальна конфігурація пам'яті збільшує пропускну здатність системи, що може бути реалізовано без модифікації процесора або ОЗП. Це не є методом збільшення швидкодії в класичному розумінні, так як при використані багатоканальних систем, ми просто збільшуємо кількість модулів пам’яті, за рахунок чого збільшується кількість даних що може бути оброблена за одиницю часу. В двоканальному

режимі – двукратно, в чотирьохканальному – чотирьохкратно. Більшість десктопних рішень підтримують лише двоканальний режим роботи, в той час як серверні материнські плати та процесори мають 4 канали, а в деяких випадках навіть 8 (найчастіше на двопроцесорних платах).

Згідно з написаним вище, налаштування таймінгів є найбільш сумісним та безпечним але водночас й найбільш впливовими на швидкодію пам'яті DDR4. Серед основних для налаштування можна відмітити наступні первинні та вторинні таймінги згідно специфікації JEDEC. Також слід відмітити що всі таймінги вимірюються в кількості тактів між операціями, таким чином мають пряму залежність від частоти пам’яті й відповідно до частоти можливо також розрахувати чисту затримку в наносекундах між виконанням операцій.

Найбільш розповсюдженими первинними таймінгами, що більше всього впливають на швидкодію є:

– CAS Latency (tCL) — це час, що проходить між подачею команди читання і доступом до даних у певній колонці. Параметр tCL є одним з найважливіших, оскільки він безпосередньо впливає на затримку доступу до даних у пам’яті.

– Row Address to Column Address Delay (tRCD) — це затримка між активацією рядка (Row) і подачею команди доступу до колонки (Column). Параметр tRCD впливає на швидкість виконання команд, оскільки контролює час очікування перед тим, як можна почати читати дані з комірки.

– Row Precharge Time (tRP) — це час, необхідний для завершення доступу до активного рядка і підготовки банку для доступу до нового рядка. Параметр tRP визначає затримку перед деактивацією одного рядка і активацією іншого, що є критичним для ефективної роботи пам'яті.

– Row Active Time (tRAS) — це загальний час, протягом якого рядок залишається активним для доступу до даних. Після активації рядка цей параметр визначає мінімальний час, протягом якого рядок має залишатися відкритим для виконання операцій читання або запису.

– Формула для tRAS іноді записується у вигляді: tRAS= tRCD+ tCL, де tRCD і tCL — відповідні первинні таймінги.

Вторинні таймінги

Refresh Cycle Time (tRFC) — це час, необхідний для оновлення всіх комірок пам'яті в банку. Параметр tRFC дуже важливий для стабільності даних, оскільки кожна комірка пам'яті повинна регулярно оновлюватися для збереження своїх значень. Це пов’язано з тим, що у DRAM комірки зберігають дані у вигляді зарядів, які можуть розряджатися.

Command Rate (CR) — це час між подачею команд доступу до пам'яті. Цей параметр, позначається як 1T або 2T, визначає, скільки тактів потрібно почекати перед подачею нової команди після попередньої.

- 1T означає, що нова команда може бути подана на наступному такті.
- 2T означає, що перед подачею нової команди потрібно зачекати один такт.

Four Activate Window (tFAW) — це мінімальний час, який повинен пройти між активацією чотирьох різних рядків у межах одного банку. Параметр tFAW обмежує кількість одночасних активацій для запобігання перевантаження пам'яті та захисту від перегріву.

Write Recovery Time (tWR) — це час, необхідний для завершення операції запису перед тим, як наступна команда може бути виконана. Цей параметр є важливим для коректного завершення запису даних, і його значення визначає, коли можна починати інші операції після запису.

Таймінги оперативної пам'яті DDR4 визначаються численними параметрами, які взаємопов’язані між собою. Налаштування одного параметра часто впливає на значення інших, і оптимізація параметрів вимагає збалансованого підходу. Основними з цих параметрів є первинні таймінги (tCL, tRCD, tRP, tRAS), а також деякі вторинні таймінги (tRFC, tFAW, tWR) [9].

Приклади налаштувань основних залежностей та взаємодії таймінгів. Основні залежності між таймінгами та їх взаємозвязок при налаштуванні представлено у таблиці 1.

Таблиця 1

Основні залежності таймінгів

Залежність між tRAS, tRCD і tCL	**tRAS —** це загальний час, протягом якого рядок активний для виконання операцій. Його значення обчислюється на основі tRCD та tCL, оскільки він об'єднує час активації рядка і доступу до колонки. Базова формула, що встановлює зв'язок між цими таймінгами: tRAS=tRCD+tCLtRAS = tRCD + tCLtRAS=tRCD+tCLЗменшення tRCD або tCL дозволяє знижувати tRAS, що прискорює доступ до нового рядка. Однак надто низьке значення tRAS може викликати помилки, оскільки рядок може не встигнути завершити необхідні операції.
Співвідношення між tRP і tRCD	**tRP** визначає час підготовки банку для активації нового рядка після завершення операції з попереднім рядком. Це значення повинно бути принаймні рівне або більшим за tRCD, щоб забезпечити достатню підготовку до нової операції. При налаштуванні tRP зазвичай враховують значення tRCD. В ідеалі значення tRP ≈ tRCD, оскільки обидва параметри впливають на час доступу до нового рядка. Високе значення tRP підвищує затримку при переході між рядками, знижуючи загальну продуктивність.
Залежність між tRFC і tRAS	**tRFC** — це час, необхідний для оновлення всіх комірок у банку, і він має тісний зв’язок з **tRAS**, оскільки обидва параметри впливають на періодичність оновлення рядків. При зменшенні tRAS важливо забезпечити достатньо часу для оновлення пам'яті, тобто tRFC має бути достатньо високим, щоб завершити процес оновлення даних. Якщо tRFC занадто малий, дані можуть бути нестабільними, особливо під час високих температур або при великих навантаженнях на пам'ять.
Зв'язок між tWR і tRP	**tWR** визначає час, необхідний для завершення операції запису, і повинен бути налаштований таким чином, щоб не конфліктувати з **tRP**. Якщо значення tRP менше tWR, операція переходу до нового рядка може розпочатися до завершення попереднього запису, що призведе до втрати даних. Зазвичай tWR ≥ tRP, щоб забезпечити завершення запису перед початком підготовки нового рядка до активації.
Час активації банків tFAW і tRRD	**tFAW** — це мінімальний час, що повинен пройти між активацією чотирьох різних рядків у банку. Він обмежує кількість активацій одночасно для запобігання перегріву і захисту від перевантаження. **tRRD** визначає мінімальний час між активацією двох рядків у банку. Його значення повинно бути меншим за tFAW, щоб забезпечити можливість послідовного активаційного доступу до кількох рядків, але з обмеженням на кожні чотири активації. Зазвичай, якщо tRRD занадто малий, це створює надмірне навантаження на банки, що знижує стабільність роботи пам'яті. Тому tFAW ≥ 4 * tRRD для забезпечення стабільності. При налаштуванні таймінгів важливо дотримуватись балансу між параметрами для забезпечення стабільності пам'яті та уникнення конфліктів.
Співвідношення CAS Latency (tCL) і частоти пам’яті	Зі збільшенням частоти пам'яті значення tCL зазвичай підвищується для забезпечення стабільності системи, оскільки доступ до даних у кожному такті стає швидшим.При збільшенні частоти пам'яті та одночасному зниженні tCL можна досягти високої продуктивності, однак це збільшує ризик помилок.
Баланс між tRCD, tRP і tRAS	Оптимізація tRCD і tRP з одночасним коригуванням tRAS дозволяє знизити загальну затримку при доступі до нового рядка. Це особливо важливо для завдань, що вимагають високої частоти доступу до різних рядків, таких як обробка великих обсягів даних. Наприклад, зменшення tRCD і tRP на одиницю такту може призвести до значного приросту швидкодії в системах з інтенсивним випадковим доступом до пам’яті.
Вплив обмежень на стабільність пам'яті	Значення tRFC не повинно бути надто низьким, оскільки це може викликати втрату даних. У випадках, коли інтенсивність використання пам'яті висока, зниження tRFC може погіршити стабільність системи. При агресивній оптимізації параметрів важливо враховувати стабільність, оскільки зміна tRFC може вимагати також змін інших параметрів, таких як tRAS і tWR, для забезпечення завершення всіх процесів.

В доповнення до таблиці слід зазначити, що якщо значення tCL і tRCD знижені, загальний час доступу до пам'яті стає меншим, що дає приріст у швидкодії. Однак потрібно, щоб значення tRP забезпечувало підготовку рядка до деактивації перед новим доступом.

Налаштування tRAS на основі суми tRCD і tCL дозволяє забезпечити збалансовану роботу, при якій рядок залишається активним на достатньо довгий час для завершення всіх операцій. Збалансування tFAW і tRRD особливо важливо в багатоканальних системах, де одночасно активуються численні рядки. Значення tFAW ≥ 4 * tRRD допомагає уникнути перегріву і перевантаження, що особливо важливо для стабільності при інтенсивному навантаженні на пам'ять.

Оптимізація первинних і вторинних таймінгів може суттєво знизити латентність пам'яті та збільшити швидкодію системи. Зниження tCL та tRCD прискорює доступ до нових комірок, що особливо помітно у завданнях із випадковим доступом. Зменшення tRP та tRAS дозволяє швидше переходити між рядками, що знижує затримки для великих обсягів даних. Параметр tFAW дозволяє уникнути перегріву при інтенсивному використанні пам'яті, що особливо важливо для високопродуктивних обчислень [5, 9].

Задачі залежні від швидкодії пам’яті. Існує цілий перелік задач, найбільш чутливих до швидкості доступу до оперативної пам'яті що можуть отримати найбільш помітний приріст в швидкодії, оскільки продуктивність напряму залежить від обсягу і швидкості обміну даними між процесором і пам’яттю [7]. Насамперед це:

– Обробка великих обсягів даних (Big Data): Аналітичні завдання та обробка великих масивів даних, наприклад, для дослідження ринкових трендів або аналізу наукових даних, значно залежать від швидкості пам'яті. Системи пам'яті впливають на час доступу до даних, що критично при обробці мільярдів записів.

– Наукові розрахунки та моделювання: Задачі наукового моделювання, такі як моделювання клімату, симуляції фізичних процесів і прогнозування погоди, вимагають обробки великих обсягів даних у реальному часі. Тут швидкість ОЗП має критичне значення, оскільки багато чисельних алгоритмів працюють з великими масивами даних і виконують численні звернення до пам’яті.

– Машинне навчання та глибоке навчання (ML/DL): Навчання та інференція великих нейронних мереж залежать від швидкості ОЗП, особливо якщо моделі не можуть повністю вміститись у кеш. При великих розмірах даних (наприклад, у NLP або обробці зображень) швидкість передачі між ОЗП і процесором суттєво впливає на загальний час навчання.

– Віртуалізація та багатозадачність: У віртуалізованих середовищах, де декілька віртуальних машин (ВМ) одночасно використовують ОЗП, ефективний доступ до пам'яті відіграє важливу роль. Висока швидкість ОЗП допомагає зменшити затримки та забезпечити стабільну роботу для кожної ВМ.

– Редагування відео і графіки: Обробка відео в реальному часі або робота з графікою великого розміру потребують високої швидкості передачі даних з ОЗП для ефективної роботи. При обробці високоякісного відео (4K і вище) або створенні складних 3D-моделей часті звернення до пам'яті можуть значно вплинути на продуктивність.

– Ігрові та графічні додатки: Сучасні ігри та графічні програми часто потребують великого обсягу пам'яті для текстур і даних, які повинні швидко завантажуватися в графічний процесор (GPU) через ОЗП. Висока швидкість ОЗП мінімізує затримки в обробці графіки, що позитивно позначається на FPS (кадрах за секунду) та загальній плавності гри.

Висновки

В статті було розглянуто особливості оптимізаціх взаємодії між процесором та оперативною пам’яттю, зокрема через налаштування таймінгів та частоти ОЗП. Проведено аналіз основних особливостей архітектур та взаємодій всередині пам’яті на рівні контролерів та материнських плат. Було виявлено що налаштування таймінгів, а також застосування багатоканальної пам'яті дозволяють значно підвищити швидкодію без втручання в апаратну архітектуру. Особливо це актуально для задач, що вимагають інтенсивного використання пам’яті, таких як наукові обчислення, машинне навчання та віртуалізація. Налаштування первинних і вторинних таймінгів, таких як CAS Latency (tCL), Row-to-Column Delay (tRCD),

Row Precharge Time (tRP) та Row Active Time (tRAS), дозволяє знизити затримку доступу до даних. Підвищення частоти пам'яті дозволяє значно збільшити пропускну здатність пам'яті, що сприяє підвищенню продуктивності у багатозадачних середовищах і під час роботи з великими обсягами даних.

Розглянуті програмні методи оптимізації підсистеми пам’яті та налаштування таймінгів дозволять ефективно використовувати наявні ресурси, навіть без зміни апаратної частини. Зокрема, для багатопотокових завдань з високим навантаженням на пам’ять такі методи можуть забезпечити стабільну роботу і високу продуктивність без додаткових витрат на обладнання.

У майбутніх дослідженнях доцільно розширити аналіз на різні моделі і покоління процесорів та виробників чипів пам’яті. Також більш детально зосередитись на дослідженні особливостей архітектури ОЗП, з тестуванням реальних відмінностей та параметрів різних модулів пам’яті в залежності від архітектури виробника, та ранговості. А також більш детально розглянути особливості контролерів пам’яті процесорів, що можуть фундаментально відрізнятись та мати особливості що критично впливають на кінцеву продуктивність систем.

Список використаної літератури:

1. Tanenbaum, A. S., Austin, T. Structured Computer Organization: International Edition. New York: Pearson Education, 2013. 656 с.
2. Hennessy, J. L., Patterson, D. A. Computer Architecture. Boston: Morgan Kaufmann, 2017. 856 с.
3. Patterson, D. A., Hennessy, J. L. Computer Organization and Design: The Hardware/Software Interface. Morgan Kaufmann, 2021. 812 с.
4. Gilreath, W. F. Computer Architecture: A Minimalist Perspective. Boca Raton: CRC Press, 2019. 352 с.
5. Intel Corporation. Intel 64 and IA-32 Architectures Software Developer's Manual: Combined Volumes: 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D, and 4. Santa Clara: Intel Press, 2019. 1247 с.
6. Lin, Y., Snyder, L. Understanding Modern x86 Assembly Language: 32-bit, 64-bit, SSE, and AVX. San Francisco: No Starch Press, 2018. 464 с.
7. Hwang, K., Briggs, F. A. Computer Architecture and Parallel Processing. New York: McGraw-Hill Education, 2010. 752 с.
8. Kain, M. M., Mocaby, W. J. Intel Microprocessors: Hardware, Software, and Applications. Upper Saddle River: Prentice Hall, 2014. 824 с.
9. JEDEC Solid State Technology Association. DDR4 SDRAM JESD79-4. 2012. 312 с.

Автори статті

Целованський Тихон – аспірант, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.
ORCID: 0009-0006-7574-126X
Шикула Олена – доктор фізико–математичних наук, професор, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.
ORCID: 0000-0002-7385-2816

Authors of the article

Tselovanskyi Tykhon – postgraduate, State University of Information and Communication Technologies, Kyiv, Ukraine.
ORCID: 0009-0006-7574-126X
Shykula Olena – Doctor of Science (physics and mathematics), Professor, State University of Information and Communication Technologies, Kyiv, Ukraine.
ORCID: 0000-0002-7385-2816