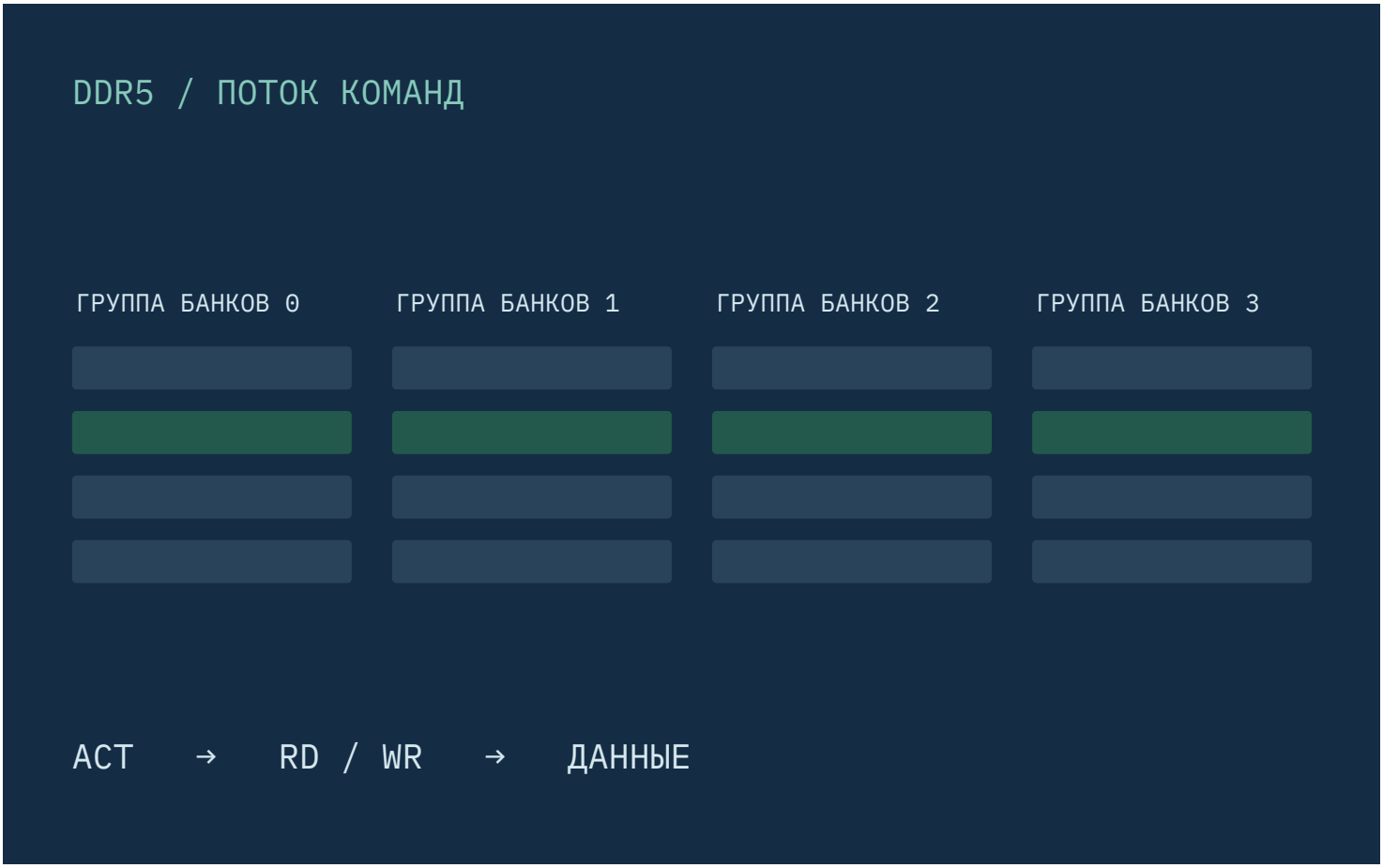
2026



DDR5: ПРИНЦИПЫ РАБОТЫ ПАМЯТИ

Архитектура, команды и тайминги

ЯЧЕЙКА / СТРОКА / КОМАНДЫ / РЕГЕНЕРАЦИЯ ДАННЫХ

Самостоятельная теоретическая часть серии руководств по DDR5

ВЕРСИЯ 1.01 (05.11.2026)

Русское издание · От устройства памяти к командам и таймингам

ПАХТУНОВ М. Д. / T.ME/PXTNV | T.ME/PXTECH

© Максим Пахтунов, 2026 · Все права защищены

Оглавление

01. Введение в теорию DDR5 . . . 5

1.1. Размещение данных работающей программы . . . 5

1.2. Биты, байты и основные характеристики памяти . . . 5

1.3. Основные темы руководства . . . 6

1.4. Порядок работы с руководством . . . 7

02. Ячейка DRAM, считывание и восстановление данных . . . 8

2.1. Заряд, ключ и линии массива . . . 8

2.2. Активация строки . . . 8

2.3. Открытая строка и буфер строки . . . 9

2.4. Назначение команд PRE и REF . . . 9

03. Организация памяти: от подканала до ячейки . . . 10

3.1. Уровни организации памяти . . . 10

3.2. Ранг и разрядность микросхемы DRAM . . . 10

3.3. Банки и группы банков в микросхеме DRAM . . . 11

3.4. Размер строки и пределы независимости . . . 12

3.5. Вспомогательные компоненты модуля . . . 13

3.6. Коррекция ошибок внутри DRAM и защита канала . . . 13

3.7. Ключевые отличия DDR5 . . . 14

04. Тактовый сигнал CK, MT/s и наносекунды . . . 15

4.1. Скорость передачи и обозначение DDR5-6000 . . . 15

4.2. Сравнение задержек CL30 и CL36 . . . 16

4.3. Минимальный интервал в тактах и наносекундах . . . 16

4.4. Такт памяти и такт контроллера . . . 16

05. Обработка запроса: CPU, IMC и планировщик . . . 17

5.1. Условия обращения к DRAM . . . 17

5.2. Адресация и выбор ресурса памяти . . . 17

5.3. Выбор команд планировщиком . . . 18

06. Команды и состояния банка . . . 19

6.1. Основные операции DRAM . . . 19

6.2. Сценарии доступа к строке A . . . 20

6.3. Временные ограничения на выдачу PRE . . . 21

07. Тайминги: система временных ограничений . . . 22

7.1. Типы временных ограничений . . . 22

7.2. Совместное действие временных ограничений . . . 22
7.3. Настройки BIOS и производительность программ . . . 23

08. Активация, чтение и закрытие строки . . . 24

8.1. tCL: интервал от READ до начала выдачи данных . . . 24
8.2. tRCD: интервал от ACT до READ или WRITE . . . 24
8.3. tRP: подготовка банка к следующей активации . . . 25
8.4. tRAS: минимальное время активности строки . . . 25
8.5. tRC: полный цикл одного банка . . . 26

09. Активации: RRD и скользящее окно FAW . . . 27

9.1. Ограничения на выдачу ACT . . . 27
9.2. Скользящее окно четырёх активаций . . . 27
9.3. Соотношение FAW и 4 × RRD . . . 28
9.4. Область действия окна FAW и дополнительные события . . . 29

10. Доступ к столбцам и интервалы CCD . . . 30

10.1. Объём данных на одну команду доступа . . . 30
10.2. Задержка первого ответа и непрерывная передача данных . . . 31
10.3. tCCD_L_WR и tCCD_L_WR2: условия применения при записи . . . 31
10.4. Чередование обращений к группам банков . . . 32
10.5. Коррекция ошибок, ECS и CRC (контроль передачи) . . . 32

11. Запись, смена направления передачи и условия выдачи PRE . . . 34

11.1. Общая шина данных и два источника сигнала . . . 34
11.2. CWL: интервал от WRITE до начала передачи данных . . . 34
11.3. tWTR: интервал от завершения передачи данных записи до READ . . . 35
11.4. READ → WRITE: переключение с чтения на запись . . . 35
11.5. tWR и tRTP: интервалы до выдачи PRE . . . 36
11.6. Ограничение tCCD_WTRA . . . 37

12. Регенерация и сохранность данных . . . 39

12.1. Периодическая регенерация ячеек DRAM . . . 39
12.2. tREFI и tRFC: максимальный средний интервал и минимальная пауза . . . 39
12.3. Обычный режим регенерации и FGR . . . 40
12.4. REFsb: регенерация одноимённых банков . . . 41
12.5. Планирование регенерации и ограничения отсрочки . . . 42
12.6. Температура, RFM и PRAC . . . 43

13. Служебные режимы DRAM . . . 44

13.1. Power-down: условия входа, пребывания и выхода . . . 44
13.2. Self-refresh: автономная регенерация данных . . . 44

13.3. Регистры режима: MRR и MRW . . . 46
13.4. ZQ-калибровка и осциллятор DQS . . . 46
14. Параллелизм и расписание команд . . . 47
14.1. Работа двух банков на общей временной шкале . . . 47
14.2. Чередование обращений, локальность и очередь запросов . . . 47
14.3. Параллелизм рангов и подканалов . . . 48
15. Задержка, пропускная способность и нагрузка . . . 49
15.1. Задержка CAS и полное время выполнения запроса . . . 49
15.2. Пропускная способность интерфейса . . . 49
15.3. Параметры памяти на критическом пути запроса . . . 50
15.4. Интерпретация результатов тестирования . . . 51
16. Контроллер Intel: третичные тайминги, RTL и параметры регенерации . . . 52
16.1. Обозначения RDRD, WRWR, RDWR и WRRD . . . 52
16.2. RTL, QCLK и тренировка памяти . . . 52
16.3. Параметры регенерации в контроллере . . . 53
17. Сводная модель работы памяти и практическое применение . . . 54
17.1. Общая модель последовательности операций . . . 54
17.2. Пять критериев анализа таймингов . . . 55
17.3. Материалы для дальнейшего изучения . . . 55
Приложение A. DDR3 → DDR4 → DDR5: развитие архитектуры памяти . . . 56
A.1. Общие принципы и различия поколений DDR . . . 56
A.2. DDR3 → DDR4: изменения архитектуры при сохранении предвыборки 8n . . . 57
A.3. DDR4 → DDR5: переход к двум подканалам . . . 58
A.4. Передача 64 байт по 32 линиям данных при BL16 . . . 58
A.5. Обозначения таймингов и различия условий их применения . . . 60
A.6. Изменения межбанковых ограничений и интервалов после записи . . . 60
A.7. CL8, CL16 и CL30: примеры задержки CAS 10 нс . . . 61
A.8. Четыре показателя времени и скорости . . . 62
A.9. Способы организации регенерации . . . 62
A.10. Питание, защита данных и настройка интерфейса . . . 63
A.11. Влияние архитектуры на производительность памяти . . . 64
Приложение B. Сокращения и обозначения . . . 65
B.1. Сокращения, команды и регистры . . . 65
B.2. Параметры таймингов . . . 68
B.3. Единицы измерения и условные обозначения . . . 70
Источники и документация . . . 71

01. Введение в теорию DDR5

Процессор выполняет программу, а оперативная память хранит нужные ей данные. Разберём, как память получает запрос, находит данные и передаёт их обратно — и почему это занимает время.

1.1. Размещение данных работающей программы

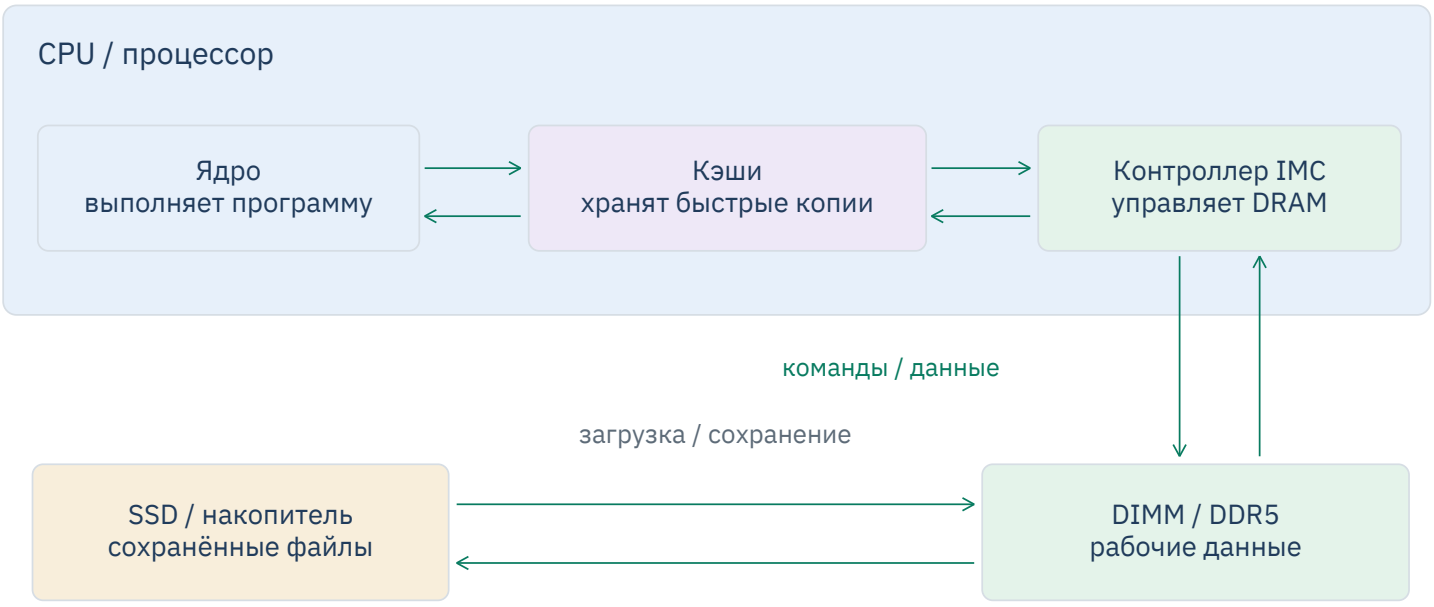
Когда вы открываете фотографию, запускаете игру или проводите расчёты в таблице, процессор (CPU) выполняет команды программы и работает с её данными. Оперативная память (RAM) хранит используемые сейчас команды и данные. DDR5 — одно из поколений такой памяти.

Файлы хранятся на накопителе, например SSD. При работе нужные части файлов загружаются в RAM. Чтобы удерживать данные, обычной DRAM нужны питание и регулярное обслуживание. После выключения компьютера её содержимое не сохраняется, поэтому результат работы нужно записать в файл.

Внутри процессора есть кэши — небольшие быстрые хранилища копий данных. Если нужная копия уже там, обращаться к DRAM не требуется.

Обменом управляет контроллер памяти: он принимает запросы, выбирает порядок обслуживания и посылает DRAM команды. В рассматриваемых современных платформах он встроен в процессор и называется IMC. Модуль DIMM — знакомая плата памяти, которую вставляют в разъём компьютера.

Рисунок 1. Где находятся и как перемещаются данные



Обмен с DDR5 проходит через IMC. Загрузка и сохранение файлов показаны как логический обмен; интерфейсы ввода-вывода не изображены.

1.2. Биты, байты и основные характеристики памяти

Бит — единица информации с двумя значениями: 0 или 1. Восемь бит образуют байт. Программы кодируют числа, буквы и изображения последовательностями битов. Микросхема хранит эти последовательности в ячейках; ячейки DRAM рассмотрим в главе 2.

Ёмкость показывает, сколько данных помещается в памяти. Задержка — сколько приходится ждать ответ на запрос. Пропускная способность — сколько данных удаётся передать за секунду.

Для больших объёмов используют приставки. Каждая следующая десятичная единица в тысячу раз больше предыдущей: 1 кБ = 1000 байт, 1 МБ = 1000 кБ, 1 ГБ = 1000 МБ = 1 000 000 000 байт. Для двоичных единиц этот множитель равен 1024, то есть 2^{10}. Они удобны при описании памяти с двоичной адресацией и имеют собственные названия.

КиБ — кибибайт: 1024 байта. МиБ — мебибайт: 1024 КиБ, или 1 048 576 байт. ГиБ — гибибайт: 1024 МиБ, или 1 073 741 824 байта. Международные обозначения — KiB, MiB и GiB. Вставка «би» указывает на двоичную приставку. Например, 4 КиБ содержат 4096 байт, а 4 кБ — 4000 байт; 1 ГиБ примерно равен 1,074 ГБ.

В английских обозначениях bit или b означает бит, B — байт. Плотность кристалла в этом мануале записана как «Гбит»; английский вариант — Gbit. В названиях документов сохраняется запись производителя, в том числе Gb. Для точных двоичных размеров используем КиБ и ГиБ, а скорость в ГБ/с выражаем в миллиардах байт в секунду.

В документации DRAM обозначение плотности исторически связано с двоичной организацией массива. Например, рассматриваемый кристалл SK hynix с паспортной плотностью 32 Гбит содержит 32×2^{30} бит пользовательских данных, то есть 4 ГиБ. Плотность описывает один кристалл; ёмкость комплекта складывается из всех его модулей. Устройство модуля разберём в главе 3.

Наносекунда, нс, равна одной миллиардной доле секунды; микросекунда, мкс, — одной миллионной. Работа интерфейса также измеряется тактами. Такт — один период повторяющегося синхронизирующего сигнала. В главе 4 переведём такты в наносекунды на численном примере.

Характеристика	Вопрос	Пример единицы
Ёмкость	Сколько данных поместится?	ГиБ
Задержка	Когда придёт нужный ответ?	нс
Пропускная способность	Сколько данных придёт за секунду?	ГБ/с

1.3. Основные темы руководства

Массив памяти организован в строки и столбцы. Он разделён на банки — части с собственным состоянием выбранной строки. Контроллер сначала подготавливает нужную строку, затем выбирает порцию данных для чтения или записи. Эти действия занимают разное время в зависимости от исходного состояния банка.

Пока один банк готовит строку, другой способен выполнять свою работу. Но несколько банков могут использовать одни и те же линии передачи данных. Такие общие линии называют шиной. Поэтому время ожидания отдельного ответа и скорость большого потока запросов зависят от разных ограничений.

Тайминг задаёт временное условие, которое должно быть выполнено до следующего действия. При меньшем значении контроллер может сократить интервал, если такой режим допустим для устройства. Программа от этого не обязательно ускорится: очередь, занятая шина или другое условие могут по-прежнему задерживать ответ.

> Критический путь — цепочка ожиданий, которая определяет момент получения конкретного результата. Изменение параметра ускоряет выполнение этого запроса, только если сокращает ожидание на таком пути.

1.4. Порядок работы с руководством

Основная командная модель относится к DDR5 SDRAM — синхронной памяти с динамическим хранением данных. Базовые учебные примеры используют BL16, пакет из 16 передач на каждой линии данных. Другую организацию x4/x8/x16, плотность или специальный режим называем рядом с выводом, который от них зависит.

Подробные численные требования выбранного устройства взяты из SK hynix 32Gb DDR5 SDRAM, Rev. 1.2, July 2025. В главе 16 отдельно разобраны поля контроллера из названных документов Intel. Эти уровни описания нужно различать при сопоставлении чисел.

Главы 2–6 дают основу: ячейка, модуль, такт, запрос и команды. Главы 7–13 объясняют ограничения, регенерацию и служебные режимы; главы 14–15 — параллелизм и производительность. Основной маршрут завершается итоговой картой в главе 17. Главу 16 и приложения A и B можно оставить для второго чтения: в них собраны платформенные поля, подробное сравнение поколений и справочник обозначений.

Даташит — техническое описание устройства от производителя. BIOS/UEFI — встроенное программное обеспечение платы, через меню которого доступны настройки. Название поля BIOS может отличаться от обозначения ограничения DRAM. Английские термины даны рядом с объяснениями.

02. Ячейка DRAM, считывание и восстановление данных

В DRAM бит хранится как электрическое состояние. Чтобы прочитать его надёжно, память усиливает слабый сигнал и восстанавливает содержимое ячейки.

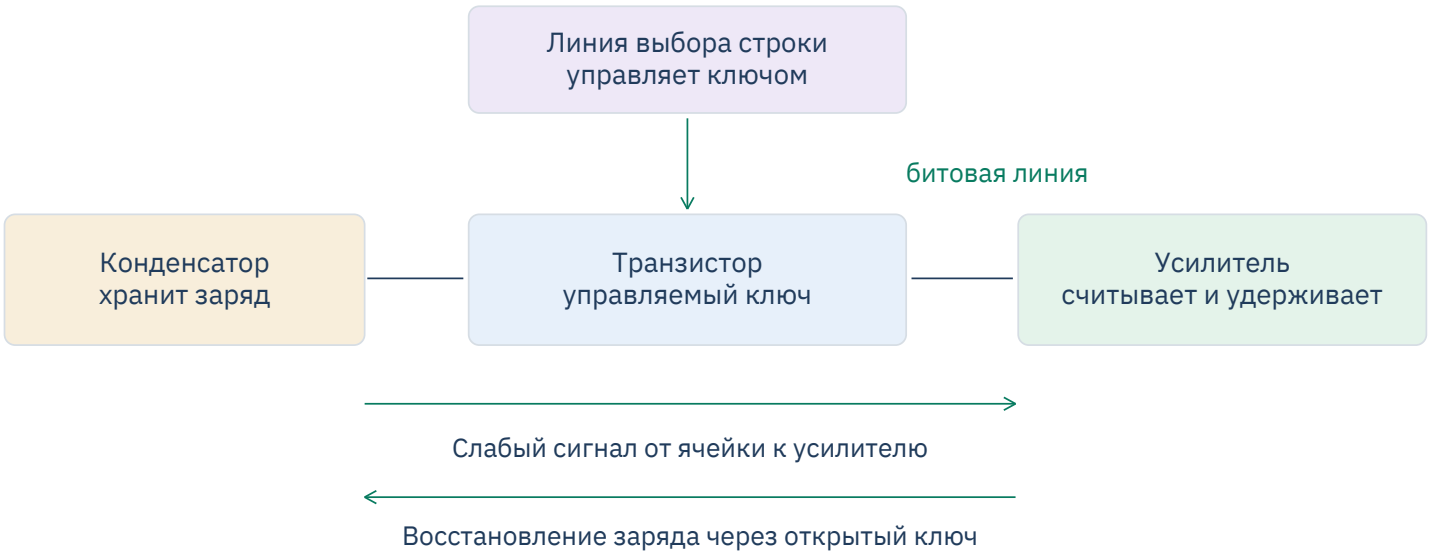
2.1. Заряд, ключ и линии массива

Упрощённая ячейка DRAM состоит из конденсатора и транзистора доступа. Конденсатор — элемент, способный удерживать электрический заряд. Транзистор работает как управляемый ключ: соединяет ячейку с битовой линией или разъединяет их. Два различимых состояния заряда кодируют один бит. Модель называют 1T1C: один транзистор и один конденсатор.

Wordline, или линия выбора строки, управляет транзисторами ячеек строки. Bitline — битовая линия, через которую состояние выбранной ячейки влияет на вход усилителя считывания. Слова row (строка) и column (столбец) обозначают объекты, выбираемые на разных этапах обращения: сначала выбирается строка массива, затем нужная часть активной строки для передачи.

Заряд со временем утекает. Даже если программа ничего не меняет, состояния становится труднее различить. Поэтому DRAM нуждается в периодическом восстановлении данных. Отсюда буква D: dynamic, память с динамическим хранением.

Рисунок 2. Ячейка и линии массива



Модель 1T1C: линия выбора строки управляет ключом, битовая линия связывает ячейку с усилителем.

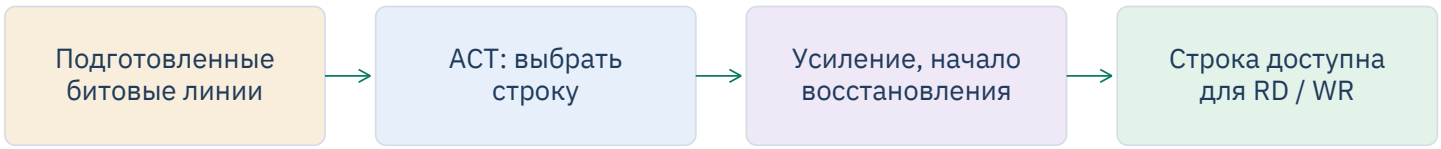
2.2. Активация строки

Перед активацией битовые линии подготовлены к считыванию. Когда открывается транзистор выбранной ячейки, заряд перераспределяется между небольшой ёмкостью ячейки и битовой линией. Возникает слабое отклонение напряжения. Усилитель считывания — sense amplifier — различает это отклонение, усиливает его до устойчивого логического состояния и удерживает результат.

Такое внутреннее считывание меняет исходное состояние заряда ячейки, поэтому его называют разрушающим, destructive sensing. При этом после нормально завершённой операции данные должны сохраниться: пока ячейка соединена с усилителем, её заряд восстанавливается до надёжного уровня. Усиление и восстановление происходят при активации.

Сигнал должен успеть стать достаточно надёжным для следующего действия. Выдача выбранных данных и безопасное завершение цикла строки требуют разного времени подготовки. Эта разница позже объяснит, почему существуют и tRCD, и tRAS.

Рисунок 3. Активация и внутреннее считывание



Слабое изменение заряда становится устойчивым состоянием усилителя.

Позже RD выбирает порцию активной строки для внешней передачи.

Последовательность состояний, без масштаба времени. Восстановление строки может продолжаться после достижения готовности к RD; tRAS проверяется отдельно.

2.3. Открытая строка и буфер строки

Когда говорят, что строка открыта, имеют в виду активное состояние банка: данные выбранной строки удерживаются его усилителями считывания. Этот набор усилителей обычно и называют row buffer — буфером строки.

Пока строка открыта, её ячейки остаются соединены с усилителями. Усилители удерживают считанные значения и поддерживают восстановление заряда ячеек. При закрытии связь размыкается, а битовые линии подготавливаются к новому считыванию. В стандартной командной модели одного банка активна одна строка.

Команда ACT активирует строку внутри DRAM: данные строки остаются в усилителях считывания. Команда RD выдаётся позже и выбирает часть уже активной строки для передачи через внешние выводы данных. Заряд ячеек считывается при активации, а последующие команды RD обращаются к данным в буфере строки.

2.4. Назначение команд PRE и REF

Перед открытием другой строки тот же банк нужно вернуть в подготовленное состояние. Команда PRECHARGE, сокращённо PRE, завершает работу с активной строкой и подготавливает битовые линии к следующей активации. Перед закрытием должны быть соблюдены условия восстановления и записи данных. После PRE нужно выдержать интервал tRP; только затем банк готов к следующей ACT.

Обычная активация восстанавливает ту строку, к которой обратились. Но программа не обязана регулярно обращаться ко всем строкам. Поэтому нужна регенерация, refresh: память периодически считывает и восстанавливает данные строк независимо от обращений программы. Это сохраняет различимые состояния ячеек по мере утечки заряда. Расписание регенерации и занятые ею ресурсы разберём в главе 12.

03. Организация памяти: от подканала до ячейки

Банки, ранги и подканалы позволяют выполнять часть операций параллельно. Но у каждого уровня свои общие ресурсы и ограничения.

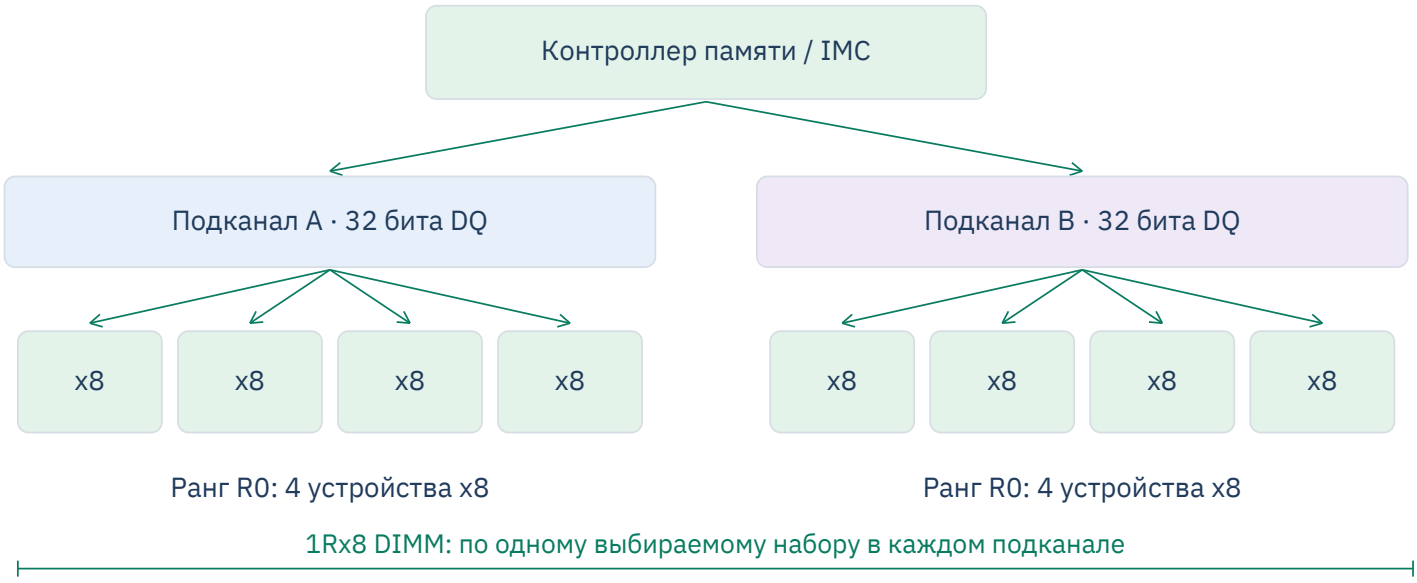
3.1. Уровни организации памяти

DIMM — модуль памяти с микросхемами и служебными компонентами. Устройство DRAM — отдельный чип памяти со своим интерфейсом и внутренней организацией. Канал объединяет несколько таких устройств. По корпусам на плате не всегда можно судить о числе кристаллов: один корпус может содержать несколько кристаллов.

В типичном DDR5 DIMM полезные 64 бита интерфейса разделены на два подканала по 32 бита. Каждый подканал может принимать собственную последовательность команд и передавать собственные данные. У модулей с дополнительными битами коррекции ширина физического интерфейса больше; 32 бита здесь означают именно данные без этих дополнительных линий.

Слово channel (канал) употребляется производителями и программами на разных уровнях: как 64-битный интерфейс платформы, его 32-битный подканал DDR5 либо отдельная сущность контроллера. Поэтому полезнее сначала установить ширину и независимость команд, а уже затем сравнивать надписи «dual channel» (двухканальный режим) в двух программах. Два подканала относятся к организации одного модуля DDR5; два независимых канала памяти — к организации контроллера процессора.

Рисунок 4. Подканалы и совместно работающие устройства



У 2Rx8 в каждом подканале два набора: R0 и R1, с общими для них DQ.

Пример полного 1R DDR5 DIMM без внешних ECC-битов, устройства x8. Показаны логические соединения.

3.2. Ранг и разрядность микросхемы DRAM

Ранг, rank, — совместно выбираемый набор устройств. Они получают согласованные команды и передают разные разряды данных. UDIMM означает модуль без регистра, буферизующего команды и адреса.

В обычном 1Rx8 UDIMM без внешнего ECC на каждом из двух подканалов работает по одному выбираемому набору из четырёх x8-устройств. У 2Rx8 таких наборов по два на каждом подканале. Наборы одного подканала совместно используют его линии данных. Второй ранг добавляет массивы памяти; число внешних подканалов остаётся равным двум.

Обозначение x8 означает восемь линий данных одного устройства. Чтобы составить полезные 32 бита подканала, в простой организации без внешнего ECC совместно работают четыре устройства x8. Для x4 понадобятся восемь, для x16 — два.

Ранг и сторона печатной платы — разные понятия. Микросхемы одного ранга могут располагаться на двух сторонах; корпуса могут содержать больше одного кристалла. Нельзя надёжно заменить 1R/2R словами «односторонний/двусторонний» или определить организацию исключительно по числу видимых чёрных корпусов.

Объект	Что обозначает	Что нельзя определить по этому объекту
DIMM	Модуль и его соединения	Число рангов по расположению микросхем на плате
Ранг	Совместно выбираемый набор устройств	Независимую шину данных для каждого ранга
x4 / x8 / x16	Ширину интерфейса DQ одного устройства	Ёмкость и число банков без даташита
Подканал	Область команд и передачи данных	Полную независимость от IMC и питания

Ёмкость удобно посчитать по всему модулю. В полном DIMM на монолитных x8-устройствах без внешнего ECC один ранг объединяет восемь устройств: по четыре на каждом подканале. В принятом двоичном обозначении плотности 16 Гбит дают 2 ГиБ на устройство, а 32 Гбит — 4 ГиБ.

Организация	Устройств во всём DIMM	Плотность устройства	Ёмкость DIMM
1Rx8	8	16 Гбит	16 ГиБ
1Rx8	8	32 Гбит	32 ГиБ
2Rx8	16	16 Гбит	32 ГиБ

Это арифметические примеры организации, без привязки к конкретным комплектам. Одинаковые 32 ГиБ получаются из восьми устройств по 32 Гбит или шестнадцати по 16 Гбит. Для регенерации эта разница существенна: параметры устройства зависят от плотности кристалла, что разберём в главе 12.

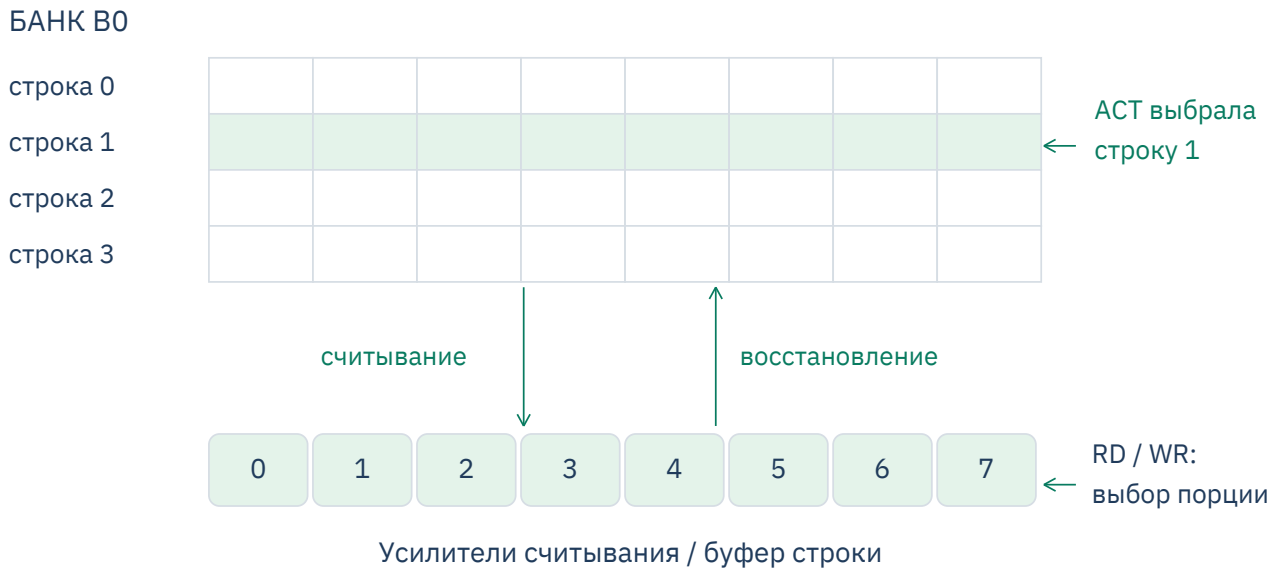
3.3. Банки и группы банков в микросхеме DRAM

Внутри DRAM массив разделён на банки, banks. В каждом банке выбирается своя активная строка, поэтому часть действий в разных банках может идти одновременно. Группа банков, bank group или BG, объединяет банки с дополнительными общими ограничениями. Между некоторыми командами к банкам одной группы требуется больший интервал, чем между командами к разным группам.

Для плотностей 16–64 Гбит в рассматриваемой организации DDR5 x4/x8 используются восемь групп по четыре банка, то есть 32 банка на устройство. В рассматриваемой организации x16 — четыре группы по четыре, то есть 16 банков. Для 8 Гбит в таблице 1 того же даташита SK hynix по два банка на BG: 16 банков у x4/x8 и 8 у x16. Геометрию и плотность конкретного устройства всегда проверяют в его даташите.

Строки и столбцы находятся внутри банка. Адрес строки выбирает строку при ACT; адрес столбца участвует в выборе порции данных при RD или WR. Физически массив устроен сложнее и содержит подмассивы. Здесь достаточно модели банка, которой пользуется внешний контроллер при выдаче команд.

Рисунок 5. Строка, столбцы и буфер строки



Связь с выбранной строкой сохраняется до закрытия.

Один логический банк устройства. Выбранная строка соединена с усилителями, которые считывают данные и восстанавливают заряд. Команды RD/WR выбирают порцию данных.

3.4. Размер строки и пределы независимости

Размер строки одного DRAM-устройства — объём данных, доступных после её активации; в таблицах геометрии его также называют размером страницы. В рассматриваемом даташите у x4 это 2048 столбцов × 4 бита = 1 КиБ, у x8 — 1024 × 8 бит = 1 КиБ, у x16 — 1024 × 16 бит = 2 КиБ. Четыре параллельно работающих x8-устройства дают 4 КиБ активной строки подканала. Команда чтения выдаёт выбранную порцию этой строки одним пакетом — burst.

Если четыре x8-устройства параллельно обслуживают подканал, их одноимённые банки активируются согласованно, а данные объединяются по разрядам. Команда ранга обращается сразу ко всем участвующим устройствам. Поэтому умножение числа устройств на число банков не даёт числа банков, которыми контроллер может управлять независимо.

Другой банк может готовить строку, пока первый передаёт данные. Другой ранг имеет собственные массивы и состояния, но обычно использует те же линии данных своего подканала. Другой подканал снимает часть этой конкуренции, однако контроллер, очереди процессора, питание и ограничения платформы остаются общими. Степень параллелизма зависит от ресурса, о котором идёт речь.

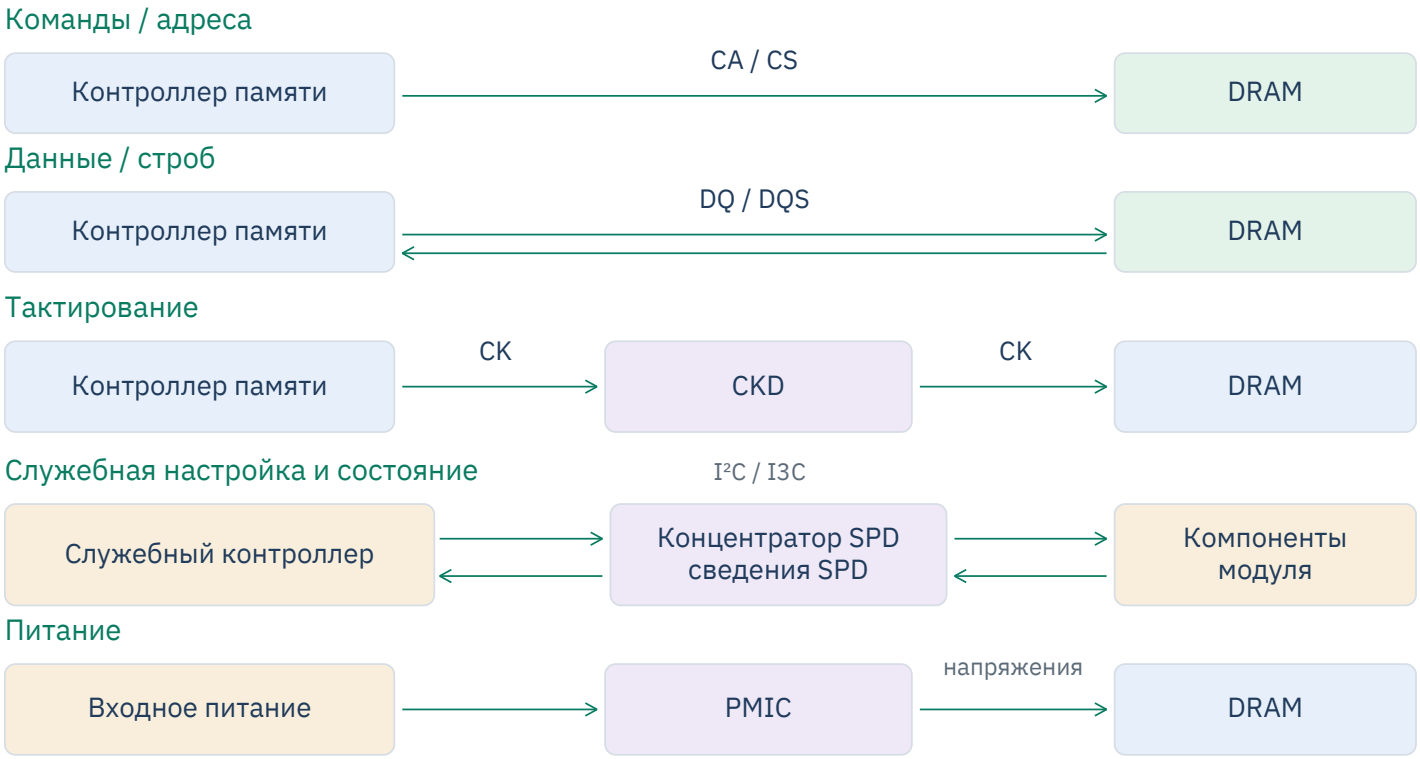
3.5. Вспомогательные компоненты модуля

SPD, Serial Presence Detect, хранит сведения о модуле в энергонезависимой памяти. В DDR5 концентратор SPD связывает внешнюю служебную шину I²C/I3C Basic с локальными компонентами, позволяя системе получать их параметры и состояние. I²C — последовательная шина с двумя сигнальными линиями: данных и тактирования. I3C Basic тоже использует две линии, но поддерживает более быстрый служебный обмен и уведомления от устройства по самой шине. В рассматриваемом исполнении концентратор также содержит датчик температуры.

PMIC, Power Management Integrated Circuit, формирует из входного питания напряжения для микросхем модуля и управляет их подачей.

CUDIMM и CSODIMM — варианты обычного и компактного модуля с буфером тактового сигнала CKD, Client Clock Driver. Он буферизует и распределяет CK от контроллера к DRAM. Его режимы и совместимость зависят от платформы.

Рисунок 6. У модуля несколько разных путей связи



Каждый ряд показывает отдельную функцию клиентского DIMM, поэтому блок DRAM повторяется. CKD буферизует тактовый сигнал CK в CUDIMM/CSODIMM; у обычного UDIMM этот сигнал идёт к DRAM без CKD.

3.6. Коррекция ошибок внутри DRAM и защита канала

On-die ECC (встроенная коррекция ошибок), ODECC, исправляет одиночные ошибки внутри кристалла DRAM. Она не исправляет ошибки, возникшие на внешнем пути передачи данных к контроллеру. Для системной ECC-защиты нужны поддержка контроллера и соответствующая организация памяти с дополнительными проверочными данными. В разделе 10.5 разберём внутреннюю коррекцию, обслуживание ECS (проверку с исправлением и сохранением результата) и контроль передачи с CRC (циклическим избыточным кодом).

3.7. Ключевые отличия DDR5

Главные отличия видны в размере внутренней выборки и организации обмена. Prefetch — порция данных, которую DRAM подготавливает параллельно для последующей выдачи: 8n и 16n означают 8 и 16 бит на каждую линию DQ. BL задаёт число передач во внешнем пакете. Ниже сравниваются конкретные x8-устройства и обычные UDIMM без системной ECC; подробности поколений сохранены в приложении A.

Признак	DDR3, 2 Гбит x8	DDR4, 8 Гбит x8	DDR5, 32 Гбит x8
Внутренняя выборка	8n	8n	16n
Банки устройства	8, без групп	4 BG × 4 = 16	8 BG × 4 = 32
Полезный интерфейс DIMM	64 DQ	64 DQ	2 подканала × 32 DQ
Базовый полный пакет	BL8	BL8	BL16

Два подканала позволяют обслуживать разные обращения, а BL16 сохраняет порцию 64 байта при ширине 32 DQ. Число банков относится к указанным плотностям и организации; варианты x4/x16 и другие режимы пакета рассмотрены в приложении A.

04. Тактовый сигнал CK, MT/s и наносекунды

Одно и то же число тактов при разной скорости памяти даёт разную задержку в наносекундах. Для сравнения таймингов нужна длительность одного такта CK.

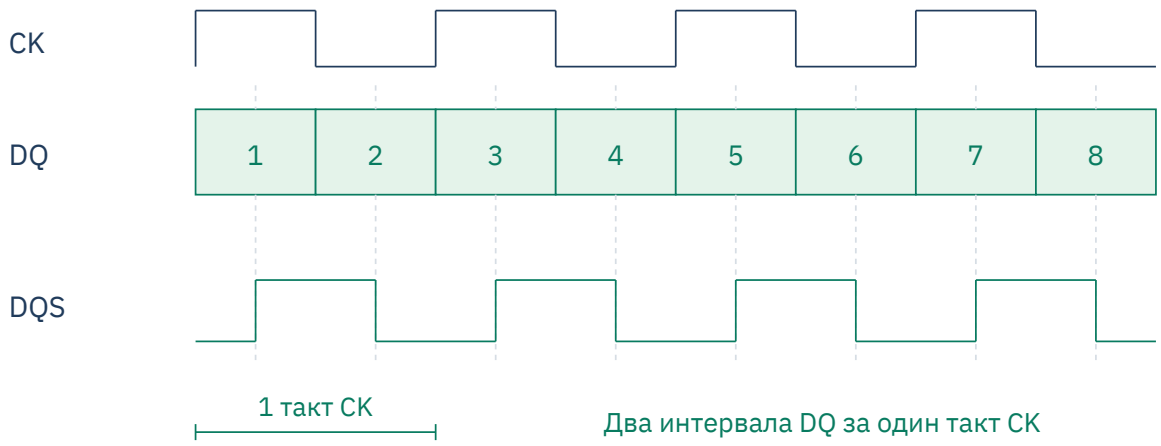
4.1. Скорость передачи и обозначение DDR5-6000

DDR означает double data rate (удвоенная скорость передачи данных): данные передаются дважды за период тактового сигнала. Для интерфейса с номинальной скоростью 6000 MT/s частота тактового сигнала CK составляет 3000 МГц. MT/s — миллионы передач в секунду на одной линии данных; MHz — миллионы периодов в секунду.

CK — тактовый сигнал интерфейса, tCK — длительность его периода. Обозначение nCK означает число таких периодов. Частота обработки внутри массива может отличаться от частоты тактового сигнала CK: DDR5 использует внутреннюю параллельную выборку и сериализацию данных на внешнем интерфейсе. Частота контроллера тоже может отличаться от частоты сигнала CK.

DQ — линии, по которым передаются данные. DQS — сопровождающий их строб: по его фронтам приёмник определяет момент приёма данных. При записи DQ и DQS посылает контроллер, при чтении — DRAM.

Рисунок 7. Тактовый сигнал CK, данные DQ и строб DQS при записи



1–8 — номера передач; рисунок показывает часть BL16.

Условный фрагмент записи: DQ и DQS выдаёт контроллер, DQS расположен в центрах окон данных на входе DRAM. Реальная фаза относительно тактового сигнала CK, преамбула, постамбула и допуски не показаны. При чтении источник — DRAM, а DQS и DQ на выходе выровнены по границам; приёмник подстраивает момент захвата данных.

На входе DRAM фронты DQS при записи располагаются внутри допустимых окон данных; при чтении исходное выравнивание DQS и DQ другое, а приёмник настраивает момент захвата.

У DQS есть специальные участки непосредственно перед пакетом данных и после него — преамбула и постамбула.

tCK, нс = 2000 / скорость, MT/s
интервал, нс = nCK × tCK

4.2. Сравнение задержек CL30 и CL36

Число CL само по себе не определяет задержку. В выбранных учебных режимах CL30 при 6000 MT/s соответствует 10 нс, а CL36 при 8000 MT/s — 9 нс; расчёты сведены в таблицу.

Здесь учтён только интервал READ→начало данных. Очередь, открытие новой строки, путь запроса от ядра и возврат данных через процессор в него не входят.

BL16 — 16 передач на линию DQ, по две за такт CK: пакет занимает 8 CK. При 6000 MT/s это 2,667 нс, при 8000 MT/s — 2 нс. CL задаёт время до начала данных, а BL — число передач.

Учебная конфигурация	Частота CK (тактового сигнала)	tCK (период)	CL × tCK	BL16 / 2
6000 MT/s, CL30	3000 МГц	0,333… нс	10 нс	8 тактов CK = 2,667 нс
8000 MT/s, CL36	4000 МГц	0,25 нс	9 нс	8 тактов CK = 2 нс

4.3. Минимальный интервал в тактах и наносекундах

Минимальные интервалы в тактах и наносекундах соблюдают одновременно. Например, требуется не менее 8 CK и 5 нс. При 6000 MT/s один такт занимает 1/3 нс, поэтому 5 нс требуют 15 CK. Это больше 8 CK: действующий минимум равен 15 CK.

Минимальный интервал округляют вверх до разрешённого шага. Полученное число нужно также проверить по кодировке регистра или доступным значениям BIOS: некоторые поля допускают не каждый целый такт. Для максимального допустимого интервала, например заданного требованием к частоте регенерации, смысл условия и направление безопасного округления будут другими.

Минимум в тактах CK = max(предел в nCK, ceil(предел в нс / tCK))

> nCK — число тактов; tCK — длительность такта. max выбирает наибольшее число, ceil округляет вверх до целого. При 6600 MT/s: 5 нс / tCK = 16,5; минимум при шаге 1 CK равен 17 CK. Кодировку регистра и условия режима проверяют отдельно.

4.4. Такт памяти и такт контроллера

QCLK — такт контроллера в документации Intel. Gear задаёт отношение частот физического интерфейса памяти (DDR PHY) и контроллера. В Gear 2 частота интерфейса вдвое выше, поэтому tQCLK = 2 × tCK.

При 6000 MT/s получаются tCK = 1/3 нс и tQCLK = 2 × tCK = 2/3 нс. Поэтому одинаковое число тактов CK и QCLK означает разное время. В регистре Intel MC_BIOS_REQ, 5E00, поле GEAR задаёт режим; здесь используется документ ID 767624, Rev. 1.3.

Command Rate (командный режим) описывает выдачу и приём команд. Его и Gear нельзя прибавлять к CL или подставлять вместо tCK. Единицу каждого поля нужно установить по документации: например, RTL в QCLK подробно разобран в разделе 16.2.

05. Обработка запроса: CPU, IMC и планировщик

CPU запрашивает данные по адресу. Контроллер превращает этот запрос в допустимую последовательность команд памяти.

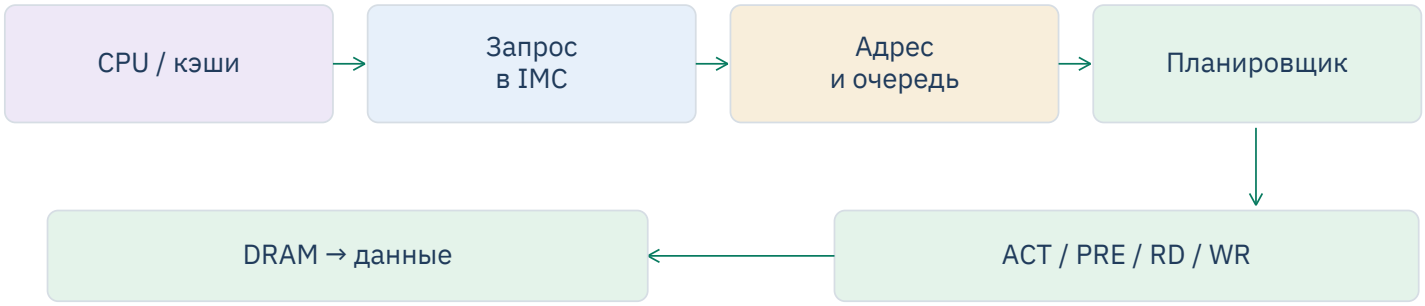
5.1. Условия обращения к DRAM

Обращение к данным начинается с кэшей процессора. После промаха на одном уровне кэша поиск продолжается: данные могут найтись дальше в иерархии или поступить из другого кэша по протоколу согласованности.

Возьмём запрос, которому нужна DRAM. Он проходит через внутренние соединения процессора и очереди подсистемы памяти к контроллеру. Часть процессора за пределами вычислительных ядер называют uncore. Устройство этого пути зависит от модели CPU; кольцевое межсоединение, ring, — один из его вариантов.

Если инструкция использует прочитанное значение, ей приходится дождаться результата. Записи нередко накапливаются в буферах и отправляются позже. Поэтому завершение инструкции записи, команда WR на интерфейсе и физическое восстановление записанных ячеек — три разных события.

Рисунок 8. От запроса к передаче данных



Логическая последовательность запроса, которому требуется DRAM.

5.2. Адресация и выбор ресурса памяти

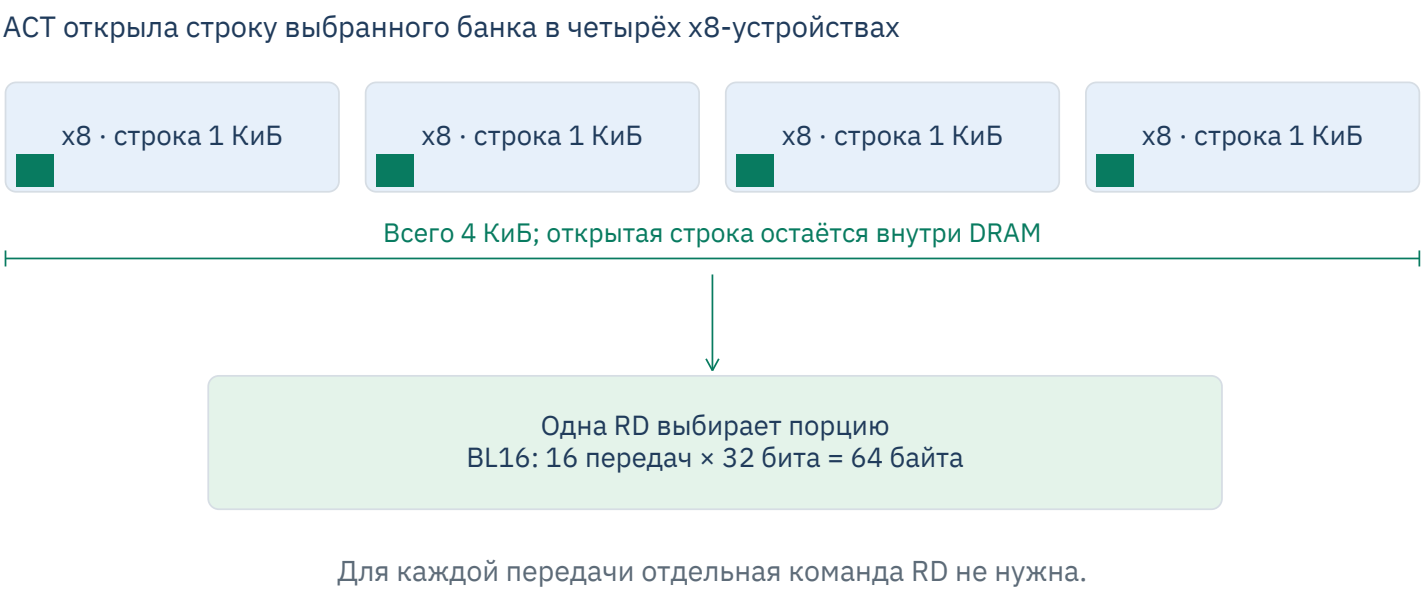
Виртуальный адрес программы предварительно проходит трансляцию в физический. По физическому адресу запроса IMC определяет, где находятся данные: канал или подканал, ранг, группа банков, банк, строка и столбец. Такое соответствие называют address mapping.

Контроллер может перемешивать адресные биты и распределять обращения ради параллелизма. Два последовательных физических адреса поэтому не обязательно соответствуют двум последовательным столбцам одного банка во всех системах.

Точная карта зависит от платформы, конфигурации и настроек; чтобы её установить, нужны данные о конкретной реализации. Для разбора команд достаточно её результата: например, ранг 0, BG1, банк 2, строка A.

После выбора банка и строки контроллер указывает в RD или WR часть адреса, определяющую столбец. Одна такая команда запускает целый burst (пакет данных): отдельная команда для каждой из 16 передач BL16 не нужна. По 32 линиям полезных данных в BL16 передаются 16 × 32 бита, то есть 64 байта; это выбранная для внешнего обмена порция активной строки.

Рисунок 9. Активная строка больше одного пакета данных



Учебный пример: четыре x8-устройства одного подканала; BL16 по 32 полезным линиям DQ. Активная строка: 4 × 1 КиБ = 4 КиБ; пакет: 16 × 32 бита = 64 байта. Закрашенная порция условна; порядок смещений и карта адресов разобраны в тексте.

Углубление: порядок данных внутри пакета

Порядок данных внутри пакета тоже задаётся правилами DRAM. Например, в таблице 29 SK hynix для READ BL16 при C3 = C2 = 0 идут смещения 0–15, а при C3 = 0 и C2 = 1 порядок начинается с 4–7, затем идёт 0–3, 12–15 и 8–11. Это локальные смещения внутри выбранной порции. Их нельзя напрямую считать младшими битами произвольного программного адреса: сначала действует адресная карта контроллера. В главе 10 отделим эту порцию от внутренней параллельной выборки.

5.3. Выбор команд планировщиком

Планировщик, scheduler, выбирает допустимую команду с учётом зависимостей запросов, состояний банков, занятости шин, очередей чтения/записи и сроков регенерации. Команды могут идти не в порядке запросов, но видимое программе поведение сохраняет установленные правила.

Приоритет открытой строки экономит закрытие и новую активацию, но может задержать запросы к другим строкам. Планировщик ищет компромисс между пропускной способностью, задержкой и справедливостью. Политика конкретного коммерческого IMC из общей модели не следует.

Параметры конфигурации задают ограничения, а контроллер применяет их при выборе момента выдачи ACT, RD, WR и других команд. Часть настроек также используется для программирования регистров самой DRAM.

> Row locality — повторное обращение к данным уже активной строки. Bank-level parallelism — возможность обслуживать разные банки с перекрытием операций.

06. Команды и состояния банка

Состояние банка определяет, какую команду можно выдать следующей и сколько придётся ждать. ACT открывает строку, RD и WR обращаются к её данным, а PRE закрывает строку.

6.1. Основные операции DRAM

ACTIVATE, ACT, выбирает строку в указанном банке и запускает её активацию. Для обычного нового доступа банк должен быть закрыт и готов. После ACT нужно выдержать row-to-column delay (интервал от активации строки до доступа к столбцам), прежде чем выдавать RD или WR.

READ, RD, выбирает данные из активной строки. WRITE, WR, задаёт место записи в активную строку; сами данные приходят по DQ позднее, согласно write latency (задержке записи). PRECHARGE, PRE, завершает работу с активной строкой и подготавливает банк к новому циклу. REFRESH, REF, запускает предусмотренную режимом операцию регенерации. Уточнения REFab и REFsb понадобятся в главе 12.

Команды и адреса передаются по шине CA (command/address), а данные — отдельно по DQ. ACT передаёт адрес строки и запускает её внутреннюю активацию. Выдачу данных наружу задаёт READ. RD и WR используют адрес столбца и параметры пакета данных. PRE меняет состояние банка и не передаёт данные. READ сначала задаёт выбор банка и группы, затем часть адреса, определяющую столбец.

Рисунок 10. Состояния одного банка



Упрощённая модель без PD/SR. Для переходов действуют соответствующие ограничения по времени; при автоматической предзарядке банк закрывается по правилам режима.

Команда	Что требуется	Что она делает	Основные ограничения
ACT	Банк готов к активации	Выбирает и активирует строку	tRP, tRC, tRRD, tFAW
RD	Нужная строка активна	Запускает чтение выбранной порции	tRCD, интервал RD→RD, CL
WR	Нужная строка активна	Задаёт место записи данных, которые поступят позже	tRCD, CWL, интервалы WR
PRE	Можно закрыть строку	Закрывает строку и готовит банк	tRAS, tRTP, tWR
REF	Выполнены условия режима	Восстанавливает содержимое массива	tRP и ограничения регенерации

6.2. Сценарии доступа к строке A

Пусть запросу нужна строка A банка B0. Если A уже открыта, имеем row hit (попадание в открытую строку). Для чтения не требуется новая ACT: после соблюдения остальных ограничений контроллер выдаёт RD и ожидает начала передачи данных. В этой ситуации tRCD и tRP не должны заново добавляться к каждому чтению.

Если B0 закрыт и готов к активации, сначала требуется ACT A, затем интервал tRCD и команда RD. Назовём это доступом к закрытому банку, closed-bank access. Слова «закрыт» недостаточно, если интервал после PRE ещё не истёк: готовность к ACT нужно проверить отдельно.

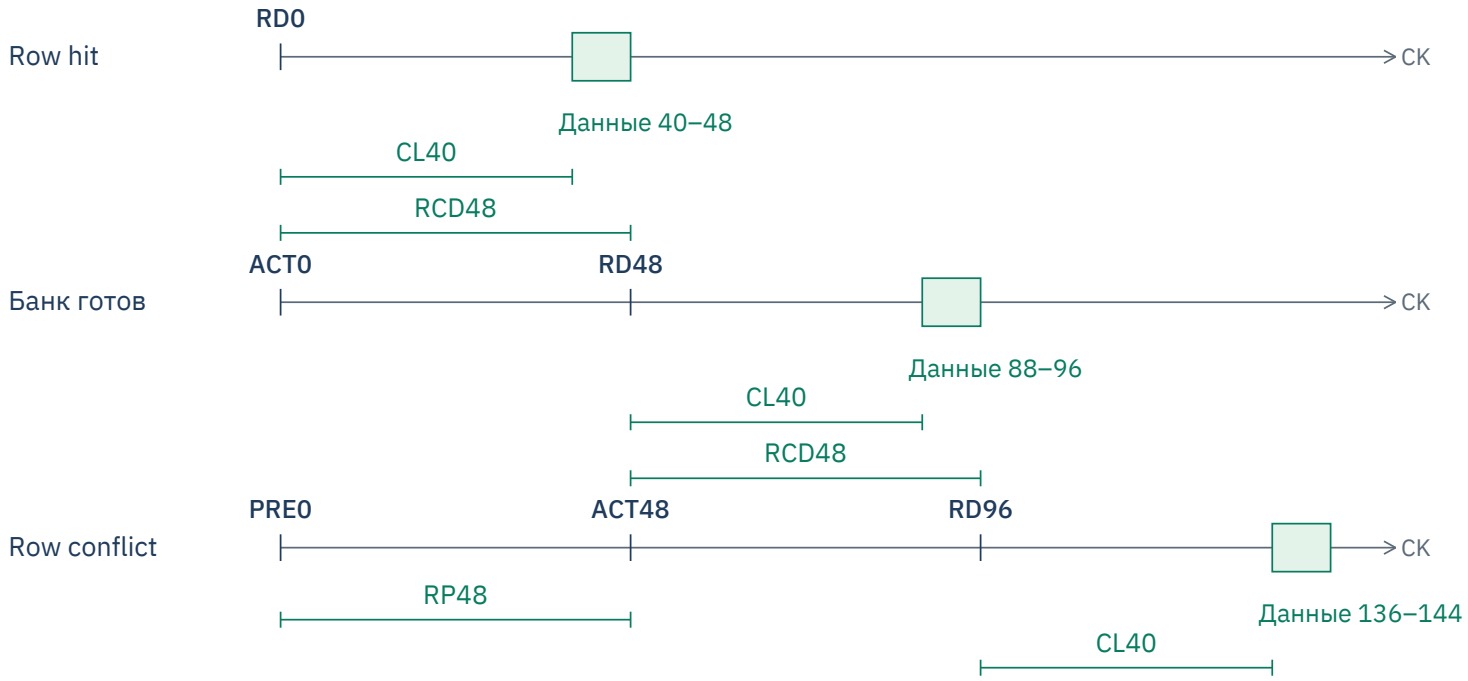
Если в B0 открыта другая строка X, возникает row conflict (конфликт строк). Контроллер должен подождать, пока можно выдать PRE, закрыть X, выдержать tRP и открыть A. Дальше путь совпадает с доступом к закрытому банку.

Row miss (промах по открытой строке) в этом руководстве означает отсутствие row hit. Различаем доступ к закрытому банку и row conflict: в первом случае новая PRE не нужна, во втором сначала нужно закрыть другую строку. В других источниках границы термина row miss могут отличаться.

На учебных схемах метки ACT, RD и WR стоят на последнем цикле соответствующей двухцикловой команды, а PRE — на цикле её приёма. Метка задаёт точку отсчёта интервалов; полная форма команды на шине CA не показана. CKn обозначает момент через n тактов CK от начала отсчёта. Поэтому стыкующиеся tRCD и CL складываются без дополнительного +1 CK.

Опорные циклы показаны в даташите SK hynix: рисунки 37–38, с. 152, и рисунок 141, с. 279.

Рисунок 11. Три пути к строке A банка B0



Учебный пример: банк B0 одного ранга; BL16; CL = 40, RCD = 48, RP = 48 (такты CK). PRE уже допустима, tRC к следующей ACT выполнен; шины свободны. Метки команд — нормативные точки отсчёта.

6.3. Временные ограничения на выдачу PRE

Закрытие открытого банка требует соблюдения нескольких временных ограничений. После ACT должно пройти не менее tRAS, после READ действует tRTP. Если были записи, должны завершиться передача данных и восстановление, ограниченное tWR. Для PRE должны выполняться все условия текущего обращения.

Данные последнего чтения могут передаваться по внешнему конвейеру и после завершения части внутренних действий. Для чтения действует RD→PRE; начало интервала восстановления после записи связано с окончанием приёма данных. Точки отсчёта разобраны в главе 11.

Auto-precharge, AP, позволяет запросить автоматическое закрытие после чтения или записи. При этом сохраняются требования tRAS, восстановления после записи и других минимумов. Устройство организует закрытие по правилам команды и режима, а контроллер должен определить, когда банк снова доступен. Обозначения RDA/WRA в схемах источников относятся к таким вариантам.

> Цепочки этой главы показывают зависимости одного обращения при свободных общих ресурсах. Полная задержка запроса CPU дополнительно включает очереди, ограничения интерфейса и путь данных в процессоре.

07. Тайминги: система временных ограничений

Тайминг связывает события в работе памяти. Чтобы правильно его прочитать, нужно знать точку отсчёта, следующее действие, ресурс, состояние, режим и единицы измерения.

7.1. Типы временных ограничений

В записи ACT→RD стрелка означает «от команды ACT до команды RD». Буква t в имени тайминга указывает на время; точные единицы задаёт документация. Минимум 48 тактов CK означает, что до истечения 48 тактов следующее действие запрещено. Позже его выполнить можно, если соблюдены остальные условия, включая предельное время обслуживания.

Многие тайминги задают минимальный интервал между командами, например PRE→ACT одного банка. CL описывает задержку выдачи данных, tWR — время восстановления после приёма данных записи. tFAW ограничивает число активаций в скользящем окне, а tREFI задаёт условия расписания обязательного обслуживания массива.

У одной пары команд могут быть разные ограничения. ACT→ACT в том же банке относится к tRC, а между разными банками — к соответствующим tRRD и дополнительным условиям. Для RD→RD имеют значение расположение банков и режим передачи. Поэтому рядом с интервалом на схеме нужно смотреть, к каким банкам и какому режиму он относится.

Для меток ACT, RD, WR и PRE используем правило из раздела 6.2. В частности, CL начинается на втором цикле READ; опорные циклы видны на рисунках 37–38, с. 152, даташита SK hynix.

Класс	Пример	Что нужно уточнить
Команда → команда	tRCD, tRP, tRTP	Банк, вид команд, опорный цикл
Команда → данные	CL, CWL	Начало данных/DQS, режим
Данные → действие	tWR, tWTR	Завершение записи и граница такта CK
Скользящее окно	tFAW	Область подсчёта и предыдущие события
Обслуживание массива	tREFI, tRFC	Режим регенерации, длительность и частота

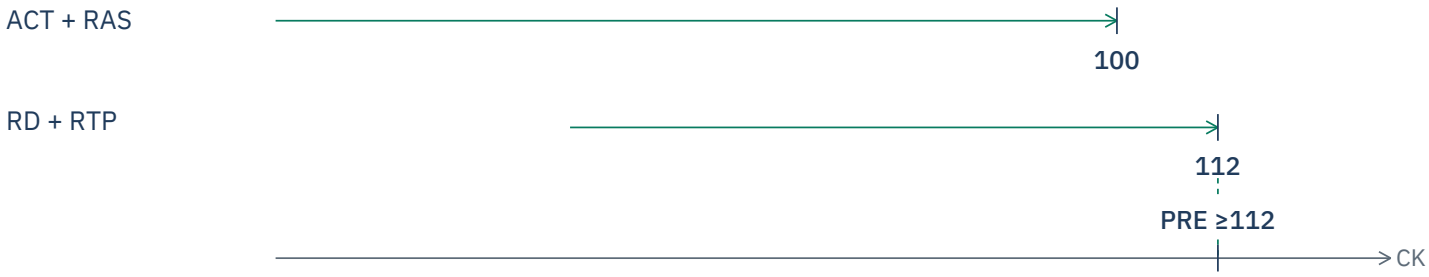
7.2. Совместное действие временных ограничений

Пусть tRAS разрешает PRE для этого банка не раньше 100-го такта, а последняя команда READ и tRTP — не раньше 112-го. Значит, PRE можно выдать начиная со 112-го такта. Обе границы заданы на одной временной шкале; выбираем более позднюю из них. Если первую границу уменьшить со 100 до 90, расписание этого обращения не изменится.

Сумма появляется там, где этапы действительно следуют друг за другом: PRE, ожидание tRP, ACT, ожидание tRCD, RD, ожидание CL. Даже такая сумма описывает только указанный путь. В другом банке часть этих действий может идти одновременно, а у row hit первые этапы отсутствуют.

Общая шина может потребовать ещё более позднего момента выдачи команды. Контроллер ждёт, пока будут выполнены все применимые ограничения, и только после этого команда становится допустимой. Планировщик выбирает запрос из очереди; момент выполнения зависит и от этого выбора.

Рисунок 12. Момент PRE определяет самая поздняя граница



Учебный пример для одного банка, время в тактах CK. Независимые границы: ACT + RAS = 100; RD + RTP = 112. PRE допустима начиная с CK112 — более поздней из двух границ.

7.3. Настройки BIOS и производительность программ

Введённое в BIOS число может отличаться от значения, которое запрограммировано после тренировки — автоматической настройки обмена при запуске. BIOS и код инициализации контроллера могут округлить его или согласовать с соседними полями. У Intel этот код называется MRC.

Запрограммированное значение задаёт одно из условий расписания. Действующее ограничение зависит от всей конфигурации, а результат в программе — от её обращений к памяти. Поэтому показание программы мониторинга само по себе ещё не объясняет причину паузы или величину выигрыша.

Названия primary, secondary и tertiary означают первичные, вторичные и третичные тайминги. Это привычные группы полей BIOS, состав которых зависит от платформы. Здесь параметры сгруппированы по действию: доступ к строке, активации, передача данных, смена направления, регенерация и служебные переходы.

08. Активация, чтение и закрытие строки

Открыть строку, дождаться данных и подготовить банк к новой строке — разные этапы. Для каждого нужен свой интервал; состав пути зависит от состояния банка.

8.1. tCL: интервал от READ до начала выдачи данных

CAS latency (задержка чтения), CL, задаёт задержку от нормативной точки команды READ до выдачи данных в соответствующем режиме. К моменту READ строка уже подготовлена. Команда выбирает порцию, после чего данные проходят внутренний путь до внешнего интерфейса. CL не включает открытие строки и не измеряет весь путь от CPU. В даташите SK hynix это показано на рисунке 34, с. 150.

CAS расшифровывается как Column Address Strobe, строб адреса столбца. Название сохранилось исторически; в DDR5 чтение задаёт команда READ.

После каждой рассматриваемой RD начало передачи данных определяется CL. Уменьшение CL может приблизить ответ. Интервал между соседними пакетами в непрерывном потоке также ограничивают tCCD, длительность пакета и расписание команд, поэтому он может остаться прежним.

Допустимые значения CL зависят от скоростного класса, режима и кодировки устройства. К началу выдачи должны быть готовы данные и строб. Даташит задаёт внешние ограничения, а для записи используется отдельный параметр CWL/write latency (задержка записи).

> tCL: READ → начало выдачи данных чтения; область — выбранный ранг/подканал и текущий режим чтения. Внешние DQS/DQ имеют отдельные электрические допуски. Связанные параметры: tRCD, tCCD, пакетная передача, переход от чтения к записи.

8.2. tRCD: интервал от ACT до READ или WRITE

tRCD, row-to-column delay (интервал от активации строки до доступа к столбцам), ограничивает ACT→READ или ACT→WRITE к активированной строке. Сначала заряд ячеек влияет на битовые линии, затем усилители считывания формируют состояние, пригодное для операции доступа к столбцам. Для ACT→WR опорные циклы показаны на рисунке 141, с. 279.

Параметр становится значимым при новой активации: доступе к закрытому банку и row conflict. Если нужная строка уже открыта и минимум давно выдержан, каждое следующее чтение не должно ещё раз ждать tRCD. Поэтому эффект зависит от доли обращений, которым требуется ACT.

BIOS может показывать отдельные tRCDRD и tRCDWR для ACT→RD и ACT→WR. Это настройки платформы; минимальные требования DRAM задаёт даташит. Соответствие этих полей регистрам устройства и фактические задержки проверяют по документации контроллера и результату тренировки.

Снижение tRCD позволяет раньше выдать команду чтения или записи. Если шина занята, контроллеру всё равно придётся ждать. При чтении новой строки tRCD и CL идут последовательно: сначала строка становится готова к RD, затем начинается задержка до выдачи данных.

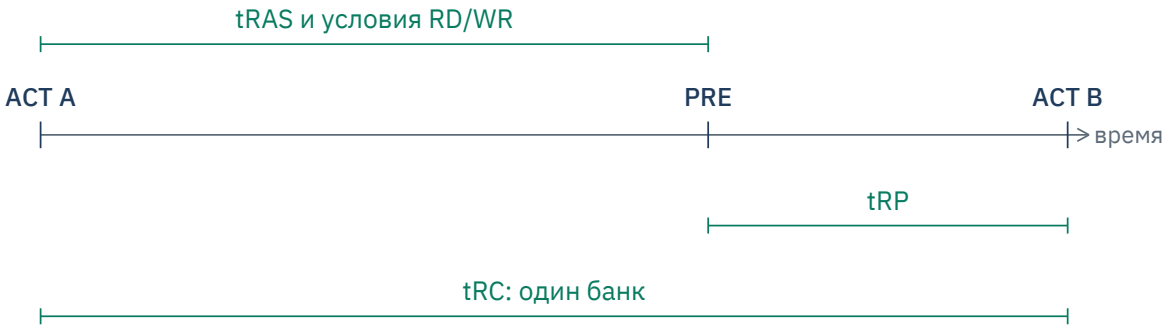
8.3. tRP: подготовка банка к следующей активации

tRP задаёт PRE→ACT для того же банка. После закрытия строки битовые линии должны вернуться к исходному состоянию, чтобы правильно различить следующий слабый сигнал. Снижение tRP сокращает эту часть смены строки, если PRE уже выдана с соблюдением всех ограничений. Стыковка READ, PRE и следующей ACT показана на рисунках 37–38, с. 152.

tRP начинается после PRE и не включает ожидание самой этой команды. При row conflict контроллер может сначала ждать tRAS, tRTP или восстановления после записи, затем выдать PRE, и лишь после этого начинается tRP. Если банк уже закрыт и готов, повторно ждать этот интервал не требуется.

Даташит DDR5 SK hynix различает PREpb для одного банка, PREsb для одинакового номера банка во всех группах и PREab для всех банков выбранного устройства. Во всех этих описаниях интервал обозначен tRP. Для имени tRPab из других семейств памяти или меню BIOS требуется установить соответствующее ограничение и его область.

Рисунок 13. Цикл одного банка: ACT → PRE → ACT



Условный масштаб. ACT→PRE: RAS; PRE→ACT: RP; ACT→ACT: RC. Показаны только команды и интервалы цикла строки.

8.4. tRAS: минимальное время активности строки

tRAS, row active time (время активности строки), задаёт минимальный интервал ACT→PRE одного банка. После активации строка должна оставаться активной достаточно долго для выполнения необходимых внутренних действий, включая восстановление. Данные могут быть доступны при обращении к столбцам ещё до того, как разрешено завершить весь цикл.

Документация задаёт максимальное время активности строки и требования регенерации. Для PRE контроллер сравнивает границу tRAS от ACT с границами от последних чтений и записей. Если tRTP либо восстановление после записи разрешает PRE позже, дальнейшее снижение tRAS не приблизит закрытие.

В последовательности ACT, RD через tRCD, PRE после tRTP сумма tRCD + tRTP задаёт одну из границ ACT→PRE. Собственное требование tRAS действует одновременно с ней. Поэтому записи tRAS = tRCD + tCL и tRAS = tRCD + tRTP не описывают общее правило: допустимый момент PRE определяют все условия этой последовательности.

8.5. tRC: полный цикл одного банка

tRC — минимальный интервал между последовательными ACT к тому же банку. Перед повторной активацией нужно дождаться, когда можно закрыть предыдущую строку, а затем завершить предзарядку. Поэтому спецификация связывает tRC с tRAS и tRP: их сумма задаёт нижнюю границу такой последовательности.

Для минимальных требований используется tRC(min) ≥ tRAS(min) + tRP(min). После перевода в такты отдельно проверяют nRC(min) ≥ nRAS(min) + nRP(min): неравенство должно соблюдаться и после округления. Фактический интервал между ACT может быть больше из-за чтения, записи, регенерации или занятого интерфейса. BIOS может вычислять tRC из других значений и не предоставлять отдельного поля.

Сокращение цикла банка помогает при нагрузке, которая часто меняет строки в одном банке, если именно этот цикл задерживает следующую ACT. Чтение уже открытой строки от этого не ускоряется. Интервалы ACT между разными банками задают tRRD и tFAW; им посвящена следующая глава.

> При моделировании PRE удобно использовать: момент PRE ≥ max(ACT + tRAS, RD + tRTP, опорный конец записи + tWR). Применяются только события данного банка и данного режима. Затем следующая ACT должна соблюдать tRP и собственный tRC.

09. Активации: RRD и скользящее окно FAW

Разные банки можно готовить параллельно. Частоту активаций ограничивают интервалы между ACT и число уже выданных активаций в скользящем окне.

9.1. Ограничения на выдачу ACT

Активация одновременно затрагивает множество ячеек и усилителей строки и нагружает внутренние ресурсы и питание устройства. Поэтому даже готовые банки нельзя активировать в произвольном количестве почти одновременно.

tRRD ограничивает интервал между ACT к разным банкам соответствующего устройства/выбранного ранга. В документации DDR5 используются tRRD_S для разных групп банков и tRRD_L для разных банков одной группы. Повторная ACT того же банка допустима при соблюдении tRC и после завершения предзарядки.

Разница S/L связана с общими внутренними ресурсами банков. Даташит задаёт интервалы для каждой области перехода.

> tRRD_S/L: ACT → ACT в другом банке; S — другая BG, L — та же BG. Уменьшение влияет на поток открытий строк. Для повторных чтений уже открытой строки эти ограничения не создаются заново.

9.2. Скользящее окно четырёх активаций

tFAW — Four Activate Window (окно четырёх активаций). Пятая учитываемая активация допустима после окончания окна, начавшегося на первой из предыдущих четырёх. После выдачи пятой проверка сдвигается: шестая сопоставляется уже со второй.

Обозначим моменты ACT как A1, A2, A3 и далее. Для последовательности, состоящей только из ACT, должно выполняться A5 − A1 ≥ tFAW, затем A6 − A2 ≥ tFAW и так далее. Одновременно продолжают действовать tRRD, готовность каждого банка, tRC при его повторном использовании и занятость командного интерфейса.

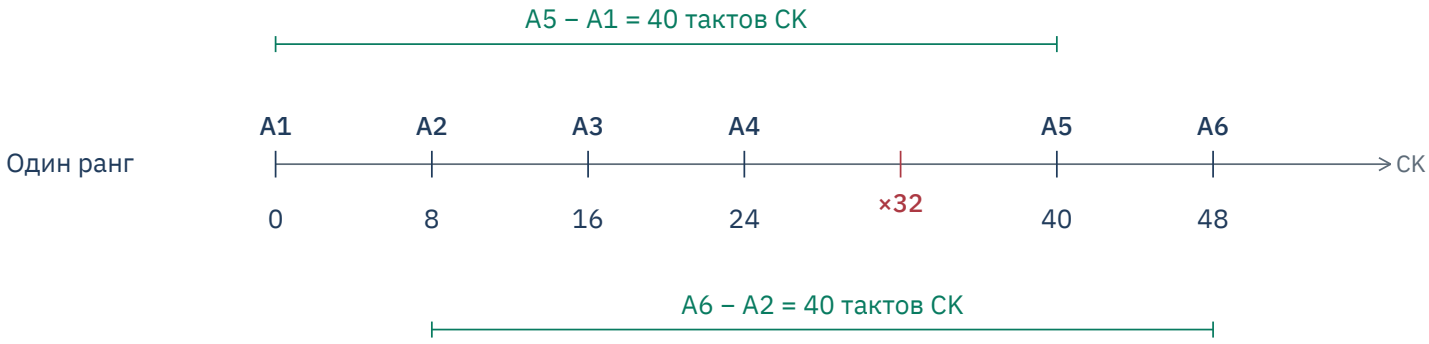
Представим учебную последовательность в одном ранге с окном 40 тактов CK и ACT через 8 тактов CK: 0, 8, 16, 24. Пятую нельзя выдать в 32: с первой прошло только 32 такта CK. В 40 окно первой уже истекло, и пятая становится допустимой, если прочие условия выполнены. Шестая допустима не раньше 48, поскольку теперь сравнивается с ACT в 8.

В этом примере от четвёртой до пятой ACT проходит 16 тактов CK при окне 40 тактов CK. FAW ограничивает частоту активаций; уже открытые банки могут оставаться активными и после сдвига окна.

A(i+4) − A(i) ≥ tFAW

следующая ACT ≥ max(граница RRD, граница FAW, готовность банка, ...)

Рисунок 14. Пятая и шестая ACT в скользящем окне



От A4(24) до A5(40) проходит 16 тактов CK при окне 40.

Пример по таблице 322 SK hynix: один ранг, x16 со строкой 2 КиБ, 6400 MT/s. tFAW = 40, tRRD_S = 8 (такты CK); BG0→1→2→3→0→1, разные готовые банки. REFsb/RFMsb исключены.

9.3. Соотношение FAW и 4 × RRD

Рассмотрим пять ACT к разным готовым банкам одного выбранного ранга, без команд регенерации REFsb и дополнительного обслуживания RFMsb между ними. Между первой и пятой ACT четыре промежутка: от первой ко второй, от второй к третьей, от третьей к четвёртой и от четвёртой к пятой. Если каждый промежуток не меньше общего нижнего предела, их сумма не меньше четырёх таких пределов. Когда tFAW не превышает сумму четырёх таких минимальных промежутков, окно уже соблюдено и не добавляет ожидания перед пятой ACT.

Предел каждого промежутка зависит от расположения банков: tRRD_S действует между разными группами банков, tRRD_L — между разными банками одной группы. Поэтому в смешанной последовательности нужно учитывать оба ограничения. Равенство tFAW = 4 × tRRD_S описывает лишь случай, когда четырёх коротких промежутков уже достаточно для окна; само по себе оно не задаёт допустимое значение tFAW для любого устройства.

В таблице 322 для DDR5-5600–6400 из даташита SK hynix 32Gb DDR5 SDRAM, Rev.1.2 (07.2025), на странице 473 PDF для DDR5-6400 заданы минимумы tRRD_S = 8 тактов CK и tRRD_L = max(8 тактов CK, 5 нс). При скорости 6400 MT/s один такт CK длится 0,3125 нс, поэтому ограничение tRRD_L соответствует 16 тактам. Для чипов x4/x8 со строкой 1 КиБ tFAW = max(32 такта CK, 10 нс), то есть 32 такта. Для x16 со строкой 2 КиБ tFAW = max(40 тактов CK, 12,5 нс), то есть 40 тактов.

Для x16 можно последовательно обратиться к группам BG0, BG1, BG2, BG3 и снова BG0, каждый раз выбирая другой готовый банк. Первые четыре ACT идут в моменты 0, 8, 16 и 24 такта CK. По короткому интервалу следующая ACT была бы допустима в момент 32, а возврат к BG0 уже удовлетворял бы длинному интервалу 16 тактов. Но окно tFAW = 40 тактов отодвигает пятую ACT до момента 40 — именно это показано на рисунке. Для рассматриваемых x4/x8 окно равно 32 тактам, поэтому при тех же переходах оно не добавляет задержку к четырём промежуткам по 8 тактов.

9.4. Область действия окна FAW и дополнительные события

Окно действует в пределах соответствующего устройства и выбранного ранга. На несколько банков выбранного набора действует одно общее ограничение FAW.

3DS (3D-stacked DRAM) — устройство с несколькими кристаллами в одном стеке. В его командной модели различают логические ранги, поэтому область окна и часть других интервалов уточняются отдельно; пример такого различия приведён в разделе 14.3.

В рассматриваемом режиме DDR5 с монолитным кристаллом REFsb и RFMsb тоже учитываются как активации для tFAW. На рисунке показана последовательность без регенерации и дополнительного обслуживания RFM, связанного с активностью строк (раздел 12.6). Для 3DS действуют дополнительные условия логического ранга.

Уменьшение действующего FAW может уплотнить поток активаций, если окно задерживало следующую команду. При обращениях к открытым строкам или более строгих ограничениях RRD и шины расписание может остаться прежним.

10. Доступ к столбцам и интервалы CCD

Открытая строка позволяет повторно обращаться к её столбцам. Скорость этого потока зависит от CL, интервалов между командами и занятости шины данных.

10.1. Объём данных на одну команду доступа

RD или WR выбирает порцию активной строки и запускает burst — последовательность передач данных. Burst length (длина пакета), BL, задаёт число передач на каждой линии DQ.

В примере BL16 и 32 полезных линии подканала дают 16 × 32 / 8 = 64 байта. Отдельное x8-устройство передаёт 16 байт, четыре устройства вместе — 64. Шестнадцать передач BL16 занимают восемь тактов CK.

DDR5 использует 16n-prefetch: данные готовятся внутри параллельно, а внешний интерфейс выдаёт их последовательно. В базовом режиме по каждой линии идут 16 передач. Отличие от предвыборки CPU разобрано в разделе A.4.

В режиме burst chop 8 (укороченной передачи), BC8, внешняя последовательность укорочена до восьми передач. Для 32 полезных линий это 32 байта и 4 такта CK. Выбор «на лету», OTF, означает выбор поддерживаемого варианта самой командой. Доступность и условия этих режимов проверяются в даташите устройства.

BL как фактическое число передач нужно отличать от расчётных RBL/WBL в формулах документации. Например, таблица 41 SK hynix сохраняет RBL = WBL = 16 для BC8 OTF, хотя передаётся восемь порций. При CRC, добавлении контрольных данных для обнаружения ошибок передачи, расчётные величины также могут измениться. Формулы применяют с учётом примечаний к режиму.

Объём пакета, байт = BL × полезная ширина DQ / 8
Длительность пакета = BL / 2 периодов тактового сигнала CK

Рисунок 15. BL16 на 32 полезных линиях



Учебный пример: один подканал, 32 полезные линии DQ; BL16 без CRC. По одному биту на линию за передачу: 16 × 32 бита = 64 байта. Шкала в тактах CK: последняя передача начинается в 7,5, полный слот заканчивается в 8.

10.2. Задержка первого ответа и непрерывная передача данных

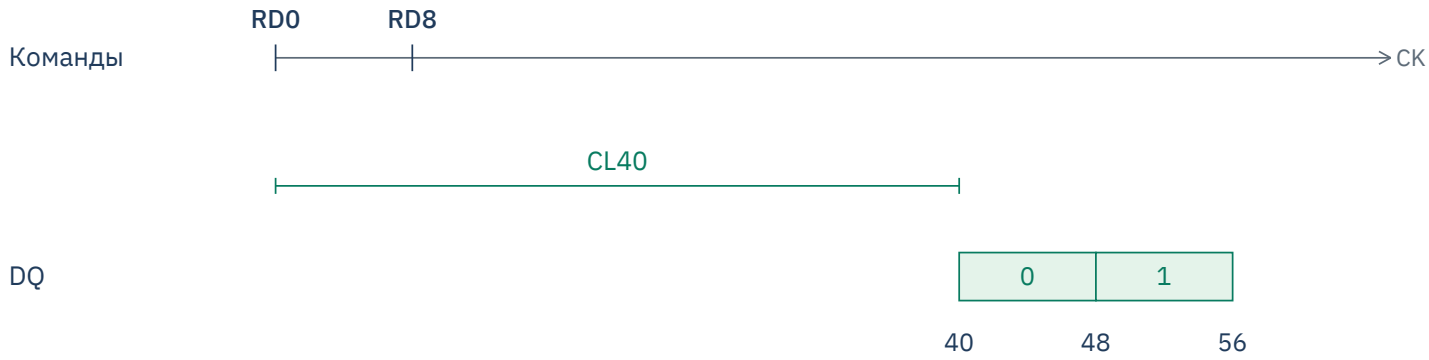
При двух чтениях из открытых строк вторая команда RD может быть выдана до появления данных по первому запросу. Это конвейер: операции перекрываются, но их пакеты данных должны передаваться по общей шине DQ без конфликтов. Время до первого ответа и интервал между ответами поэтому задаются разными ограничениями.

tCCD — семейство ограничений между столбцовыми командами. В таблице 324 рассматриваемого SK hynix для READ→READ S означает разные группы банков, M — разные банки одной группы, L — тот же банк. Эта расшифровка относится к указанной таблице; отдельного M-поля в BIOS может не быть. Для WRITE→WRITE и WRITE→READ используются собственные варианты и условия.

В таблице 322 выбранного даташита SK hynix значения M совпадают с соответствующими L. В более быстрых диапазонах таблиц 323–324 они различаются, хотя адресный смысл S/M/L в этих таблицах сохраняется. Совпадение чисел в одном диапазоне не делает M и L взаимозаменяемыми.

Пакет BL16 занимает восемь тактов CK. Между командами может потребоваться больший интервал из-за внутренних ограничений, расположения банков, направления данных и режима.

Рисунок 16. Задержка и плотность потока чтения



Учебный пример: один ранг, разные BG; BL16 без CRC; CL = 40, RD→RD = 8 (такты CK). Данные 0: CK40–48; данные 1: CK48–56. Без смены направления и конкуренции.

10.3. tCCD_L_WR и tCCD_L_WR2: условия применения при записи

При полной записи заменяется вся предусмотренная порция данных. При частичной записи часть прежних значений нужно сохранить. Для read-modify-write (чтения, изменения и записи), RMW, устройство сначала получает старые данные, объединяет их с новыми и записывает результат. Это внутренняя операция, которая может потребовать другого интервала между внешними командами. Например, маска записи может разрешить изменение одних байтов и сохранить другие.

tCCD_L_WR и tCCD_L_WR2 различают последовательности записей. В таблице 45 SK hynix для x8/x16 вторая BC8 либо частичная запись BL16 требует RMW и относится к WR, а полный BL16 без частичной записи — к WR2. Выбор зависит от режима и адресной области. Устройство выполняет RMW внутри, без отдельной внешней команды RD.

У этого различия есть физическая причина. В рассматриваемом устройстве on-die ECC объединяет 128 бит данных с 8 проверочными битами в одно кодовое слово — порцию, которую код защиты обрабатывает целиком.

Для x4 пакет BL16 передаёт только 4 × 16 = 64 бита, поэтому даже полная внешняя запись заменяет лишь половину внутреннего слова и требует RMW. Для x8 BL16 передаёт 128 бит; для x16 два таких кодовых слова обрабатываются параллельно. Поэтому «полный пакет» и «полное кодовое слово» — разные масштабы, а правила WR/WR2 нужно читать с учётом x4/x8/x16 и маски записи.

Более короткий WR2 действует при выполнении условий соответствующей последовательности. В BIOS эти различия могут быть представлены несколькими полями или скрыты в правилах контроллера.

Названия tCCD_M_WR и tCCD_M_WTR описывают отдельные случаи в таблице 323 и таблице 324 этого устройства. Суффикса sg в программе мониторинга недостаточно, чтобы однозначно сопоставить поле со всеми этими вариантами. При сопоставлении нужно различать тот же банк, разные банки одной группы и разные группы банков.

Семейство	Последовательность	Главное условие
tCCD_S / L / M	READ → READ	Расположение банков и условия устройства
tCCD_*_WR	WRITE → WRITE	Путь записи и внутренние условия
tCCD_L_WR2	WRITE → WRITE, особый случай	Выполнены условия для короткого интервала
tCCD_*_WTR	WRITE → READ	Write latency (задержка записи), пакет и интервал после записи
tCCD_WTRA	WRITE → READ AP	Дополнительное условие допустимости PRE

10.4. Чередование обращений к группам банков

Если последовательность обращений позволяет использовать разные группы и их короткие интервалы, контроллеру проще заполнить шину без промежутков. Это одна из причин, почему одинаковая частота и CL ещё не гарантируют одинаковую пропускную способность двух потоков адресов.

Если интервалы между столбцовыми командами оставляют промежутки между полезными пакетами, их уменьшение может уплотнить передачу. После достижения непрерывной передачи дополнительные уменьшения не сделают физическую шину шире. Для дальнейшего роста полезного потока нужны другие возможности: больше передаваемых битов в секунду, меньше смен направления или более удачное распределение запросов.

10.5. Коррекция ошибок, ECS и CRC (контроль передачи)

On-die ECC (внутренняя коррекция ошибок) исправляет одиночную ошибку внутри кодового слова DRAM перед выдачей прочитанных данных. Для защиты внешней передачи по DQ нужны средства на стороне контроллера и соответствующая организация памяти. Возможности системной ECC зависят от платформы и схемы защиты.

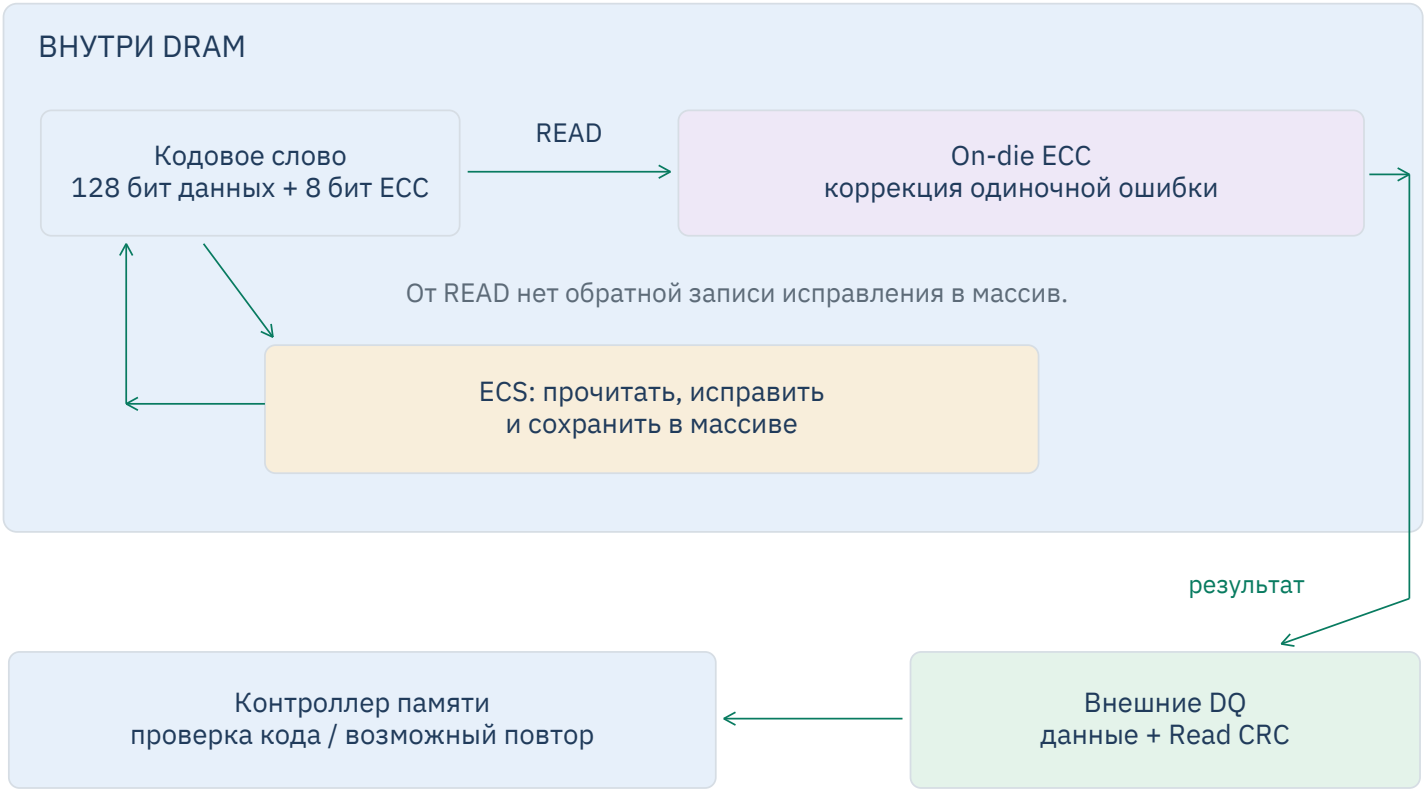
В рассматриваемом устройстве обычная READ выдаёт исправленный результат. Обратная запись исправления в массив при обычной READ не выполняется. Для записи исправленных данных служит Error Check and Scrub (проверка и исправление ошибок), ECS: внутреннее чтение, исправление и сохранение результата. Счётчики ECS учитывают результаты этого обхода с учётом настроек фильтра. Коррекции при обычном чтении в этот подсчёт не входят. Это различие задано в §4.36 даташита SK hynix, с. 300, и описании ECS на с. 301–305.

CRC, циклический избыточный код, проверяет передачу данных между контроллером и DRAM. При WRITE контрольный код формирует контроллер, при READ — память; несоответствие позволяет обнаружить ошибку. При Read CRC контроллер может повторить транзакцию; при Write CRC он отвечает за предусмотренный повтор и согласованность данных. Контроль чтения и контроль записи включаются отдельно. В рассматриваемом DDR5 Write CRC несовместим с маской данных: эти возможности нельзя считать одновременно действующими.

Включение CRC меняет расписание. В таблице 41 SK hynix для BL16 расчётное RBL или WBL с соответствующим контрольным кодом CRC равно 18 вместо 16; объём полезных данных остаётся прежним.

Таблица 150 SK hynix задаёт read CRC latency adder (надбавку к задержке чтения при контроле передачи) 2 такта CK при скорости выше 6000 и до 8800 MT/s, а в диапазоне 2933–6000 MT/s — нуль. Контроллер учитывает надбавку в программируемом CL через регистр режима MR0; к уже скорректированному CL её повторно не прибавляют.

Рисунок 17. Внутренняя коррекция и внешняя передача



CRC (контроль передачи) — только при включённом режиме.

Показан путь READ в рассматриваемом устройстве. Исправленные при чтении данные не записываются обратно в массив; для сохранения исправления служит ECS. Read CRC (контрольный код чтения) при включении формирует DRAM, проверяет контроллер. При Write CRC (контроле записи) направление обратное. Системный ECC — отдельный уровень защиты.

11. Запись, смена направления передачи и условия выдачи PRE

Команда WRITE, поступление данных, смена направления и восстановление ячеек — отдельные события. Для каждого действует своё ограничение.

11.1. Общая шина данных и два источника сигнала

При чтении данные по DQ выдаёт память, при записи — контроллер. При смене направления одна сторона освобождает линии для другой. Приёмнику нужны строб и данные с правильными временными запасами.

Смешанный поток чтения и записи требует пауз для смены направления — turnaround. Эти паузы, задержка от команды до данных и внутреннее восстановление записанных ячеек имеют разные точки отсчёта и назначение.

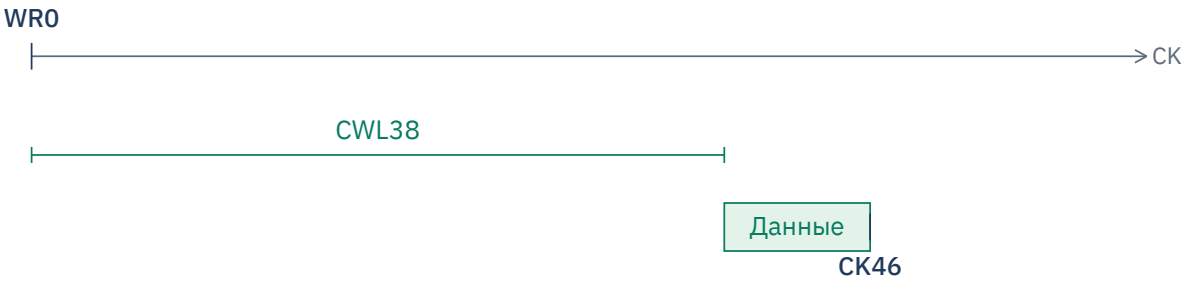
Ниже рассматриваются один выбранный ранг, BL16 и согласованные опорные точки команд. При переходе к другому рангу дополнительно учитываются переключение источника, терминация и условия приёма.

11.2. CWL: интервал от WRITE до начала передачи данных

CWL, CAS write latency (задержка записи), определяет задержку до начала приёма данных записи. В проверенном даташите отсчёт идёт от последнего цикла WRITE до первого эффективного нарастающего фронта DQS, без преамбулы. В основном примере затем передаются 16 порций данных на каждой линии. Символ WBL в формулах — расчётная величина из примечаний к таблице 41 SK hynix; исключения для BC8 OTF и CRC разобраны в разделе 10.1.

В рассматриваемом режиме DDR5 CWL = CL − 2. При CL = 40 получается CWL = 38 тактов CK; передача данных BL16 затем занимает ещё 8 тактов CK. Для следующего действия после пакета может понадобиться дополнительный интервал.

Рисунок 18. WRITE и приход данных



DQS: первый эффективный фронт в CK38

Учебный пример одного пакета записи: BL16 без CRC; CWL = 38 тактов CK. WR в CK0; первый эффективный фронт DQS — в CK38, начало последней передачи — в CK45,5, первый нарастающий фронт CK после последних данных — в CK46.

Уменьшение CWL сдвигает приём данных ближе к команде и одновременно меняет расчётные условия последующих переходов. Контроллер и DRAM должны согласованно использовать режим; отдельное видимое поле BIOS не гарантирует возможности выбирать его независимо от остальных настроек.

11.3. tWTR: интервал от завершения передачи данных записи до READ

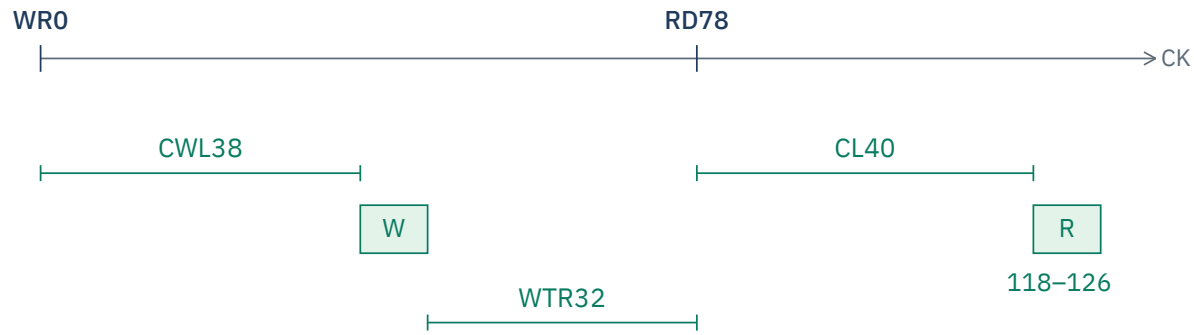
tWTR задаёт интервал до допустимого чтения после приёма данных записи. От команды WR сначала проходит CWL, затем передаётся пакет записи и выдерживается соответствующий интервал до READ. Опорная граница — первый нарастающий фронт тактового сигнала CK после последних данных записи в принятой диаграмме. В модели с отсчётом по тактам CK ей соответствует граница завершения пакета.

В таблице 322, таблице 323 и таблице 324 SK hynix полный интервал между столбцовыми командами записан как tCCD_*_WTR = CWL + WBL/2 + соответствующий WTR. Звёздочка обозначает область: короткий путь между разными BG, длинный и, где определён, промежуточный путь внутри BG.

В учебном примере CWL = 38 тактов CK, WBL = 16 и интервал после записи 32 такта CK дают WR→RD не меньше 78 тактов CK: 38 + 8 + 32 = 78. У поля Intel tWRRD могут быть другое представление интервала и поправки, поэтому его связь с этой формулой требует подтверждения.

Уменьшение WTR может сократить ожидание чтения после потока записей. Если планировщик пока выполняет другие записи либо читает другой ресурс, сокращение не обязано ускорить выбранный запрос. На больших однородных потоках переключений может быть мало, на смешанных — заметно больше.

Рисунок 19. От данных записи до READ



Первый нарастающий фронт CK после последней передачи записи: 46

Учебный пример: один ранг; BL16 без CRC; CWL = 38, WTR = 32, CL = 40 (такты CK). WR в CK0; данные записи в CK38–46; RD в CK78; данные чтения в CK118–126. Остальные минимальные интервалы выдержаны.

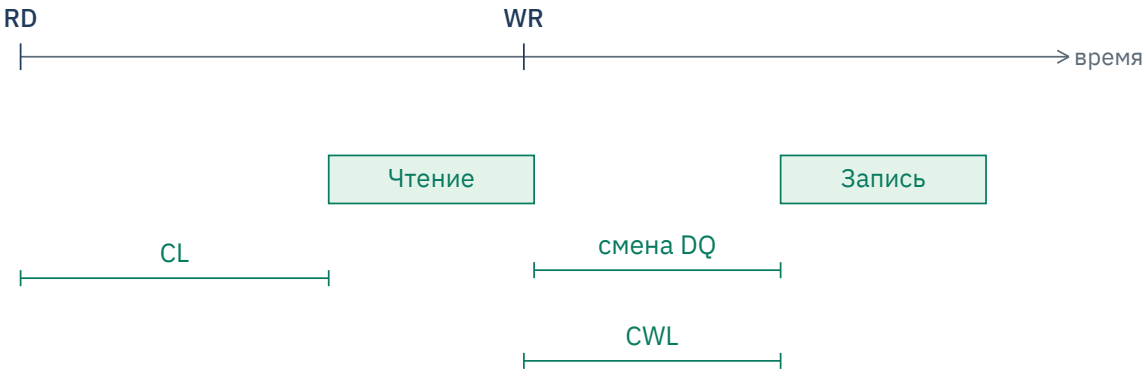
11.4. READ → WRITE: переключение с чтения на запись

После READ память выдаёт пакет, а затем контроллер должен начать свой пакет записи. Моменты выдачи команд необходимо выбрать так, чтобы внешние данные и DQS не конфликтовали. В расчёт входят read latency (задержка чтения), write latency (задержка записи), длина пакета, DQS offset (смещение строба), преамбула, постамбула и условия ранга.

tRTW, tRDWR и отдельные поля регистров могут описывать этот переход на разных уровнях. Универсальной формулы вида CL – CWL + константа для любого DDR5 из этого не следует. Для конкретного режима расчёт должен учитывать все перечисленные условия.

На диаграмме передача данных чтения заканчивается, проходит предусмотренный протоколом интервал передачи управления, затем начинается передача данных записи. Момент WR выбирается с учётом CWL: команду можно выдать до завершения передачи данных чтения, если последующая передача данных укладывается в допустимое расписание.

Рисунок 20. Смена направления после READ



Условия DQS: смещение, постамбула, преамбула

Условный масштаб: интервалы данных показаны отдельно от команд. Промежуток определяется DQS, преамбулой, постамбулой и режимом.

11.5. tWR и tRTP: интервалы до выдачи PRE

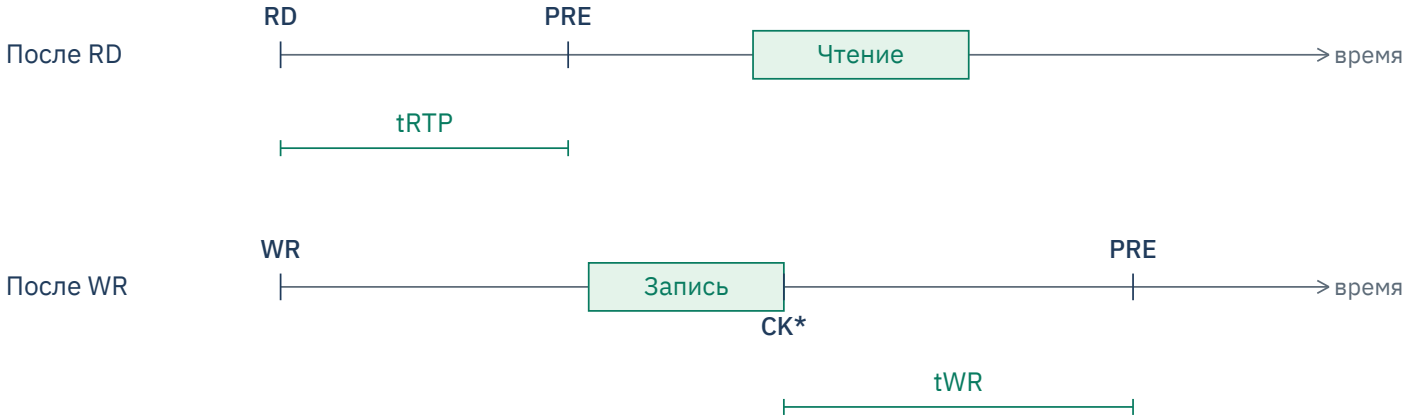
tWR, write recovery (восстановление после записи), отсчитывается от первого нарастающего фронта тактового сигнала CK после последних данных записи до допустимой PRE того же банка. Записанная информация должна успеть надёжно закрепиться в массиве. Этот интервал нужен для подготовки массива к закрытию после завершения внешнего пакета.

Полный путь WRITE→PRE в модели с отсчётом по тактам CK включает CWL + WBL/2 + tWR. Но PRE должна также соблюдать tRAS от ACT. Если минимальный интервал tRAS истекает позже, уменьшение tWR не передвинет PRE; если восстановление после записи заканчивается позже, именно оно определяет закрытие.

tRTP, read-to-precharge (интервал от чтения до предзарядки), задаёт внешний интервал READ→PRE того же банка без дополнительного суммирования CL и BL/2. В конвейерном пути данные уже могут идти к выходу, когда массиву разрешено начать предзарядку.

tWR и tRTP влияют прежде всего на возможность завершить работу с текущей строкой и перейти к другой либо к обслуживанию, требующему предзаряженного состояния. Они не приближают автоматически начало уже запланированного пакета чтения. Кодировки их значений проверяются отдельно от арифметического минимума в наносекундах.

Рисунок 21. Две разные точки отсчёта до PRE



CK* — первый нарастающий фронт тактового сигнала после данных

Сверху tRTP отсчитывается от RD; PRE может предшествовать внешним данным. Снизу tWR отсчитывается от первого нарастающего фронта тактового сигнала CK после последних данных записи. Масштаб условный, tRAS проверяется отдельно.

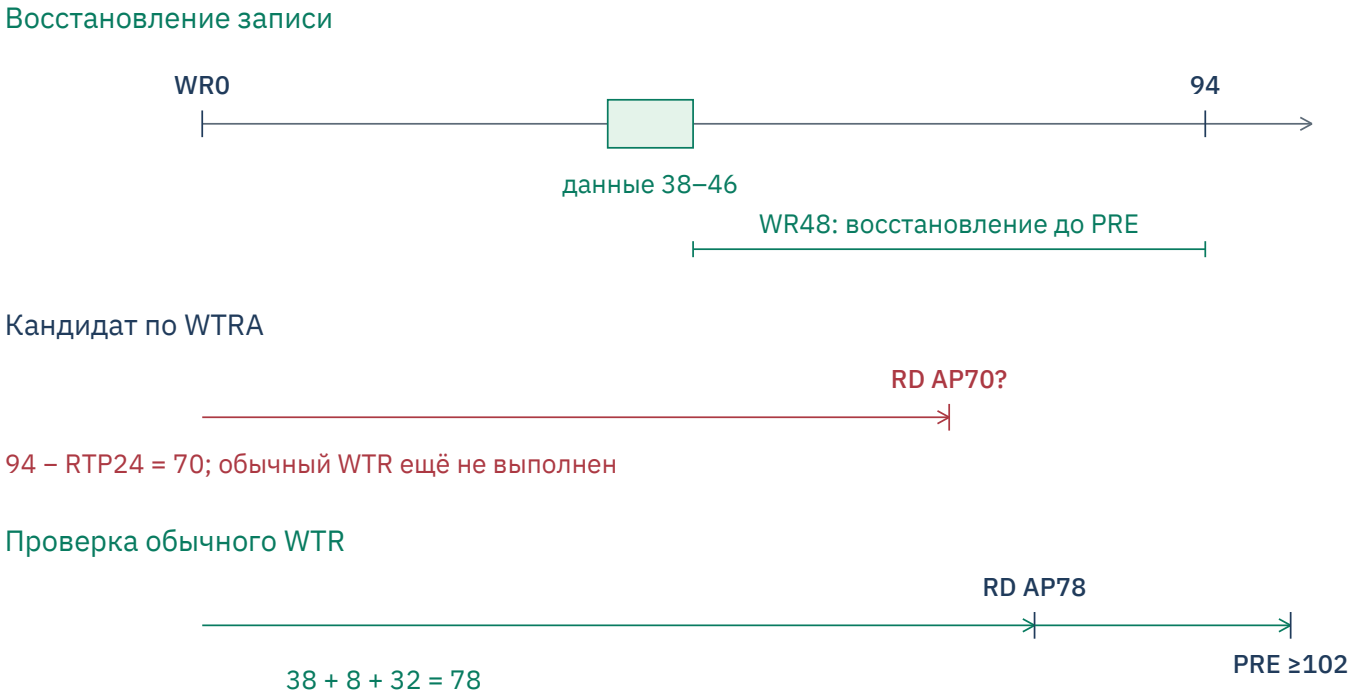
11.6. Ограничение tCCD_WTRA

Если за WRITE следует READ с автоматическим закрытием того же банка, будущая автоматическая PRE должна дождаться восстановления предыдущей записи. Поэтому появляется дополнительная нижняя граница tCCD_WTRA ≥ CWL + WBL/2 + tWR − tRTP в условиях таблицы 324 SK hynix и примечания 2 к ней.

tRTP вычитается потому, что часть ожидания до PRE будет выдержана после READ. Контроллер одновременно проверяет обычное ограничение WRITE→READ, tRAS и остальные правила режима.

Например, при WR в CK0, CWL = 38, WBL = 16, tWR = 48 и tRTP = 24 восстановление записи требует PRE не раньше CK94. По этой границе READ AP допустима не раньше CK70: 38 + 8 + 48 − 24 = 70. При обычном WTR = 32 интервал WRITE→READ составляет уже 78 тактов CK. Выбирается более поздняя граница, CK78; граница tRTP после неё наступает в CK102. В этом примере tRAS и прочие условия считаются выполненными.

Рисунок 22. READ AP должна пройти обе проверки



Учебный пример: WRITE и READ AP одного банка; BL16 без CRC, WBL = 16. Интервалы в тактах CK: CWL = 38, WR = 48, RTP = 24, WTR = 32. WR в CK0; tRAS и прочие условия выполнены. По WTRA команда RD AP допустима в CK70, но обычный WTR отодвигает её до CK78. PRE допустима не раньше CK102.

12. Регенерация и сохранность данных

Регенерация сохраняет данные в массиве. Её частота, длительность и набор занятых банков зависят от режима обслуживания.

12.1. Периодическая регенерация ячеек DRAM

Retention — время, в течение которого ячейка сохраняет различимое состояние при заданных условиях. Из-за утечки заряда это время ограничено и неодинаково для всех ячеек. Повышение температуры может усилить утечку, поэтому требования к регенерации связаны с температурным диапазоном устройства.

ACT восстанавливает выбранную строку, но программа может долго не обращаться к другим строкам. Контроллер обязан организовать регенерацию независимо от потока обычных обращений. Внутренний счётчик и логика DRAM выбирают соответствующие строки; процессор не посылает отдельный адрес каждой обновляемой ячейки.

Слишком редкое обслуживание уменьшает запас сохранности данных. Хороший сигнал на внешней шине не помогает удержать заряд внутри ячеек.

12.2. tREFI и tRFC: максимальный средний интервал и минимальная пауза

tREFI задаёт максимально допустимый средний интервал между командами регенерации в данном режиме, то есть частоту обслуживания. tRFC задаёт минимальную паузу после команды перед следующими разрешёнными действиями.

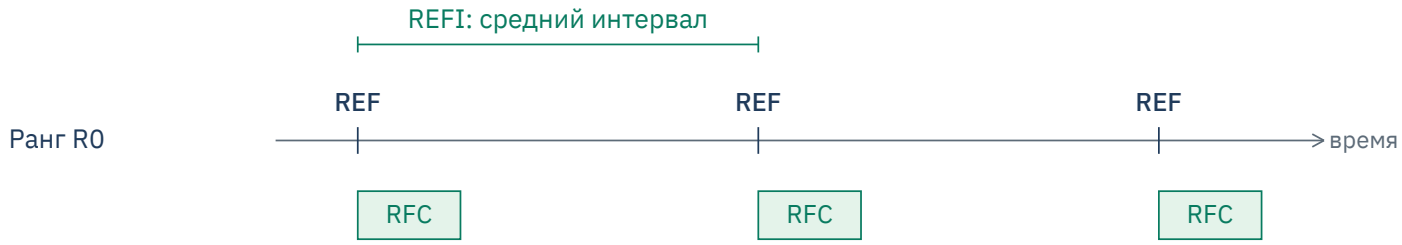
Перед регенерацией всех банков, REFab, все затрагиваемые банки устройства должны быть подготовлены: предзарядка завершена, tRP выдержан. После REFab они недоступны для обычных активных обращений в течение соответствующего tRFC. Ограничение действует в выбранном устройстве/ранге.

Увеличение настроенного tREFI разрешает контроллеру реже выполнять регенерацию. Уменьшение tRFC сокращает паузу, которую выдерживает контроллер. Длительность внутренней регенерации при этом остаётся прежней. При более редком обслуживании ячейки должны дольше удерживать заряд, а при более короткой паузе внутренние операции должны успеть завершиться.

Помимо интервала между командами и длительности одной операции есть полное окно регенерации массива. Для SK hynix 32Gb DDR5 SDRAM, Rev. 1.2, в Normal Refresh с командами REFab при 0 °C ≤ Tcase ≤ 85 °C базовое окно tREF равно 32 мс: за него требуется 8192 команды. Одна REF может обслуживать несколько внутренних строк; внутренний счётчик продвигает обслуживание по массиву.

Табличные 3,9 мкс — округлённое представление tREFI, поэтому произведение 8192 × 3,9 мкс не требуется приравнивать точно к 32 мс. Для расчёта расписания сохраняют табличное ограничение 3,9 мкс. При FGR и температурном учащении действуют соответствующие режиму правила. Окно и интервалы заданы в §4.13.4, таблице 66, с. 190; внутреннее обслуживание описано в §4.13, с. 184.

Рисунок 23. Интервал регенерации и занятость банков



Остальные ранги/подканалы могут иметь другое расписание.

Периодический REFab для выбранного ранга, условный масштаб. tRFC отсчитывается от REF; tREFI задаёт средний интервал и допускает ограниченные переносы команд.

12.3. Обычный режим регенерации и FGR

В обычном режиме для REFab действует tRFC1. Fine Granularity Refresh (регенерация более короткими и частыми операциями), FGR, дробит обслуживание: в проверенном режиме REFab выполняется чаще, а для отдельной операции действует tRFC2. Средний интервал tREFI2 равен половине tREFI1; длительность tRFC2 не обязана быть половиной tRFC1.

В даташите SK hynix для диапазона 0–85 °C средний интервал между REFab в обычном режиме составляет 3,9 мкс, а в FGR — 1,95 мкс. Выше 85 и до 95 °C интервалы сокращаются до 1,95 и 0,975 мкс. Температура здесь — TCASE микросхемы; показания датчика DIMM или концентратора SPD нельзя без проверки приравнивать к ней.

Удвоение частоты регенерации из-за температуры и включение FGR — независимые причины: одинаковый интервал не доказывает одинаковый режим или tRFC.

Для кристалла 32 Гбит в таблице 67 SK hynix указаны tRFC1 = 410 нс, tRFC2 = 220 нс, tRFCsb = 190 нс. Для 16 Гбит значения другие: 295, 160 и 130 нс. Это плотность одного кристалла. Суммарные 32 или 64 ГБ комплекта не определяют его RFC: нужна организация микросхем. В примере ёмкости модуля из раздела 3.2 одинаковые 32 ГиБ получены из устройств разной плотности; их параметры регенерации нужно выбирать по устройствам.

Режим FGR задаётся через регистр режима. Рассматриваемое устройство не поддерживает переключение «на лету».

Более короткие паузы могут занять больше времени в сумме. Для 32 Гбит при 0–85 °C доля времени недоступности выбранного набора банков при равномерном REFab составляет 410 / 3900 ≈ 10,51% в обычном режиме и 220 / 1950 ≈ 11,28% в FGR. Отдельная пауза сокращается с 410 до 220 нс, а суммарная доля таких пауз немного растёт. Расчёт не учитывает перекрытие работы и не показывает процент замедления программы.

Режим/команда	Длительность	Средний интервал, 0–85 °C	Область
Обычный режим, REFab	tRFC1	tREFI1 = 3,9 мкс	Все банки выбранного набора
FGR REFab	tRFC2	tREFI2 = 1,95 мкс	Все банки выбранного набора
FGR REFsb	tRFCsb	tREFI2 / n	Один номер банка во всех BG

n — число банков в группе рассматриваемого устройства.

12.4. REFsb: регенерация одноимённых банков

REFsb выбирает одинаковый номер банка в каждой группе: для банка № 1 обслуживаются банки № 1 всех BG выбранного устройства. В рассматриваемом DDR5 эта команда доступна в FGR.

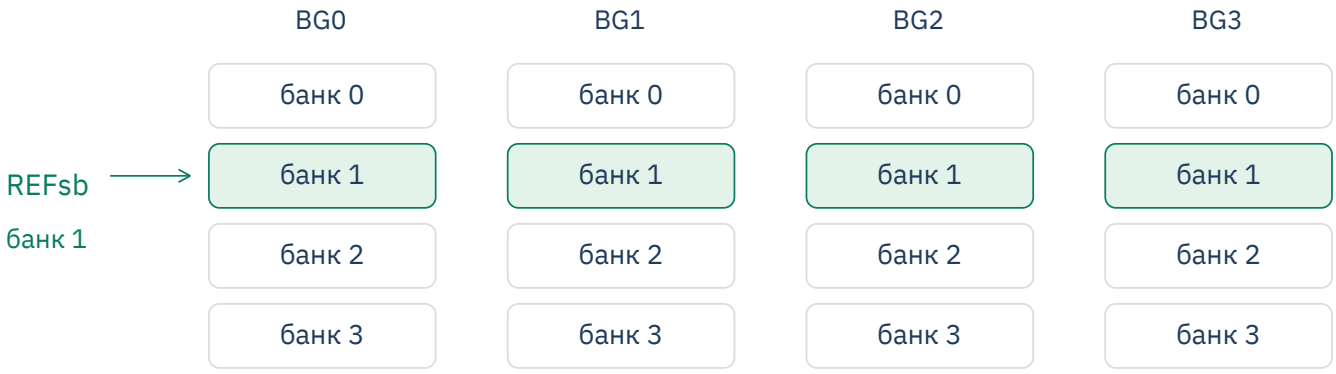
Остальные банки могут обслуживать запросы, но тоже подчиняются ограничениям регенерации. В частности, tREFSBRD задаёт интервал от REFsb до ACT для других банков, а сама REFsb учитывается в FAW для устройства с монолитным кристаллом.

Для продвижения внутреннего счётчика нужно обслужить набор всех номеров банков. Повтор одного и того же номера раньше завершения набора не заменяет обслуживание остальных. При n = 4 средний интервал между командами REFsb в приведённом FGR-примере равен 1,95 / 4 = 0,4875 мкс. Этот интервал задаётся между отдельными командами REFsb; полный цикл включает обслуживание всех номеров банков.

При смешивании REFab и REFsb важен полный набор номеров банков. Если после REFsb для № 0 и № 1 выдать REFab, не завершив набор, эти REFsb не продвинут глобальный счётчик регенерации дополнительно к REFab. Порядок номеров внутри полного набора может меняться. После REFsb обслуженные банки закрыты, и для следующего доступа к строке нужна ACT.

Чтобы сопоставить tRFCpb в BIOS или утилите с DDR5 tRFCsb, нужно знать отображаемое аппаратное поле и режим. Одного имени для этого недостаточно.

Рисунок 24. REFsb: одинаковый номер банка во всех группах



Сокращённая схема: четыре BG по четыре банка, выбран банк № 1. Остальные банки доступны с ограничениями, включая tREFSBRD и общий интерфейс.

12.5. Планирование регенерации и ограничения отсрочки

Контроллер может переносить регенерацию на более удобный момент. Спецификация ограничивает такие переносы и досрочное обслуживание: нужно учитывать выполненные и отложенные команды и соблюдать заданные временные окна.

В проверенном документе обычный режим допускает до четырёх отложенных REFab, FGR — до восьми. Соответствующие предельные интервалы между соседними REFab составляют 5 × tREFI1 и 9 × tREFI2. Отложенные команды остаются долгом и должны быть выполнены с соблюдением минимальных интервалов. Войти в автономный режим регенерации self-refresh, SR, можно с допустимым долгом, но его величина сохраняется после выхода; новые переносы не должны приводить к превышению общего предела. В этом режиме DRAM сама поддерживает данные; переход и выход разобраны в разделе 13.2. Сохранение долга задано в §4.13.6 даташита SK hynix, с. 192.

Например, если перед входом в SR отложены две REFab обычного режима, после выхода сохраняется долг в две команды. Кроме него, выход SRX создаёт отдельное требование дополнительной регенерации, разобранное в разделе 13.2. Поэтому число разрешённых переносов, уже накопленный долг и требования при смене режима учитываются раздельно.

Расчёт tRFC/tREFI из раздела 12.3 оценивает долю недоступности выбранного ресурса при равномерном REFab без перекрытия работы. Для REFsb эту дробь тем более нельзя считать простоем всего канала: другие банки и ранги могут работать.

12.6. Температура, RFM и PRAC

Короткий тест холодной памяти не охватывает все условия удержания данных. При длительной нагрузке температура меняется, а при другом наборе обращений могут проявиться ошибки слабых ячеек. Один быстрый тест ещё не подтверждает, что при сокращении расходов на регенерацию данные сохраняются в этих условиях.

Refresh management, RFM (управление дополнительной регенерацией), — дополнительное обслуживание, связанное с активностью строк и требованиями устройства. Его длительность задаёт tRFM. RFM помогает предотвращать повреждение данных из-за частых активаций: многократные активации строк могут нарушать состояние соседних ячеек. Обычная периодическая регенерация по-прежнему нужна для удержания заряда. Вариант RFMsb (Same Bank Refresh Management) даёт время для такого обслуживания одноимённых банков — с одним выбранным адресом BA — во всех группах банков. В рассматриваемом устройстве эта команда допустима только в FGR (§4.42, с. 327). RFM не заменяет периодические REF; внутренний алгоритм конкретного кристалла здесь не раскрывается.

При PRAC, Per Row Activation Counting (подсчёте активаций каждой строки), DRAM считает активации отдельных строк. Через сигнал back-off (запрос приостановки) она требует от контроллера приостановить обычное обслуживание и выдать RFM. Поддержку PRAC нужно подтвердить и для устройства, и для контроллера.

Self-refresh и условия выхода из него разобраны в разделе 13.2.

13. Служебные режимы DRAM

Память умеет снижать потребление, сохранять данные автономно и подстраивать свой интерфейс. Разберём, что происходит при этих переходах и когда после них снова разрешены команды.

13.1. Power-down: условия входа, пребывания и выхода

Power-down (пониженное потребление), PD, уменьшает потребление памяти. Вход обозначается PDE, выход — PDX. В отличие от self-refresh, режим не запускает автономную регенерацию: контроллер должен вовремя выйти и выполнить регенерацию. Поэтому задана и максимальная длительность пребывания в PD. Уже начатые внутренние операции могут завершаться после входа.

Между последней командой и PDE, а также между PDX и следующей командой действуют свои интервалы. tPD ограничивает пребывание в режиме, tXP — время после PDX до разрешённых команд с указанными в документации исключениями. tCPDED относится к задержке отключения прохождения команд при входе. Эти параметры относятся к входу в PD, пребыванию в нём и выходу.

Семейство PDEN задаёт минимальные интервалы от команд до PDE. tACTPDEN относится к ACT, tPRPDEN — к PRE, tREFPDEN — к REF. tRDPDEN учитывает завершение чтения; tWRPDEN и tWRAPDEN — запись и восстановление после неё; tMRRPDEN и tMRWPDEN — чтение и изменение регистра режима.

В таблице 57 SK hynix READ→PDE задан как CL + RBL/2 + 1, а WRITE→PDE — как CWL + WBL/2 + WR + 1 при соответствующих условиях. Суммы учитывают поступление и передачу данных, а для записи ещё и восстановление.

13.2. Self-refresh: автономная регенерация данных

Self-refresh, SR, поддерживает сохранность данных в массиве внутренней регенерацией и может обходиться без внешнего тактового сигнала CK, но требует сохранения питающих напряжений. Перед SRE все банки должны быть закрыты, tRP и остальные интервалы выдержаны, текущие пакеты завершены.

DLL (delay-locked loop) — схема подстройки задержки для временного согласования сигналов с опорным тактом. После смены режима ей может понадобиться время, чтобы восстановить рабочее состояние.

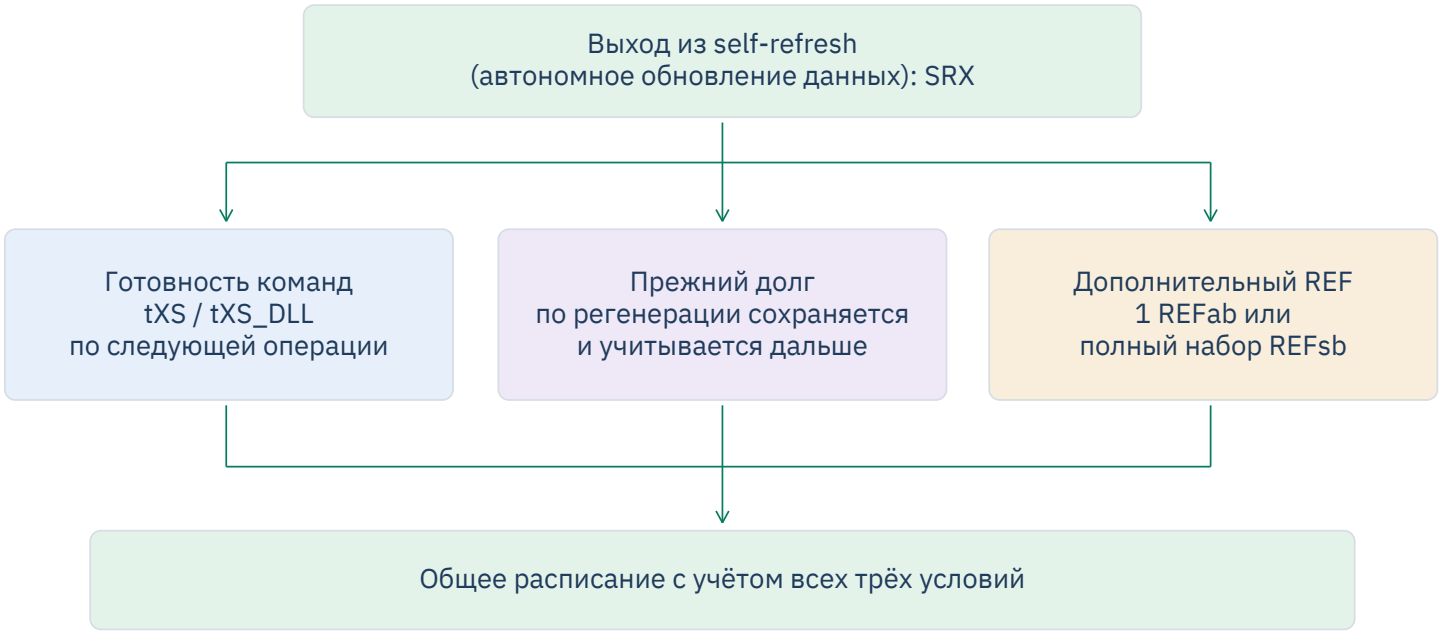
После выхода SRX перед ACT, MRW и другими командами, которые не требуют готовности DLL, выдерживают tXS. Для RD, WR и MRR требуется tXS_DLL. В проверенном варианте с монолитным кристаллом tXS привязан к tRFC1, а tXS_DLL — к tDLLK. Эти интервалы не складывают автоматически: проверяют требования именно следующей команды.

tCKSRX относится к стабильности тактового сигнала CK перед выходом SRX. Похожие названия в BIOS — tCKE, tCKCKEH, tXPDLL или tSR — могут быть историческими либо платформенными представлениями; без регистра и документации их точная трактовка остаётся неопределённой.

Согласно проверенному даташиту, после SRX требуется ещё одна операция регенерации сверх регулярного расписания: одна REFab либо полный набор n REFsb, где n — число банков в группе. Она не входит в расчёт среднего tREFI и не заменяет очередную периодическую регенерацию. Если её откладывают за пределы первого tREFI после выхода, она участвует в предельном числе отложенных команд. Требование описано в §4.9, с. 174, а дополнительные условия FGR — в §4.13.7, с. 193–194, даташита SK hynix.

В FGR незавершённый набор REFsb или нечётное число REFab перед SRE могут потребовать дополнительного обслуживания по отдельным правилам. Само ожидание tXS не заменяет этого дополнительного обслуживания.

Рисунок 25. Три независимые проверки после SRX



После SRX отдельно проверяют интервалы выхода, прежний долг по регенерации и дополнительную регенерацию. Для незавершённого FGR-набора действуют дополнительные правила.

13.3. Регистры режима: MRR и MRW

Mode registers (регистры режима), MR, хранят настройки режима устройства. MRR читает регистр, MRW изменяет его. В таблице 17 SK hynix обе операции требуют закрытых банков; для 3DS у MRW есть дополнительные условия по логическим рангам.

tMRR задаёт период таких чтений, tMRW — период записей, tMRD — задержку после программирования до разрешённых действий. Для изменения режима нужно привести устройство в нужное состояние и дождаться применения настройки. MRR и MRW обращаются к регистрам режима; данные программы читаются и записываются командами RD и WR.

13.4. ZQ-калибровка и осциллятор DQS

ZQ-калибровка настраивает выходные цепи и терминацию. Терминация, в том числе ODT внутри устройства, согласует линию и уменьшает отражения сигнала. ZQCal Start запускает фоновую калибровку. После tZQCAL команда ZQCal Latch принимает результат: к этому моменту операции на DQ должны закончиться, а в течение tZQLAT шина CA остаётся в состоянии Deselect (без выбора команды). Во время фоновой калибровки канал может продолжать обмен с соблюдением ограничений режима.

Период повторного запуска со стороны платформы отличается от этих длительностей. Точная привязка BIOS-полей tZQCSPeriod и tZQOPER к регистрам здесь не установлена.

Осциллятор DQS отслеживает изменения задержки внутреннего пути DQS. Контроллер может использовать результат измерения для поддержания настроек. tOSCOM и tOSCOA ограничивают доступ к результату в ручном и автоматическом режимах. Какому из них соответствует BIOS-поле tOSCO, без документации установить нельзя.

14. Параллелизм и расписание команд

Банки могут выполнять внутренние операции параллельно. Передачу данных из этих потоков по общей шине контроллер согласовывает во времени.

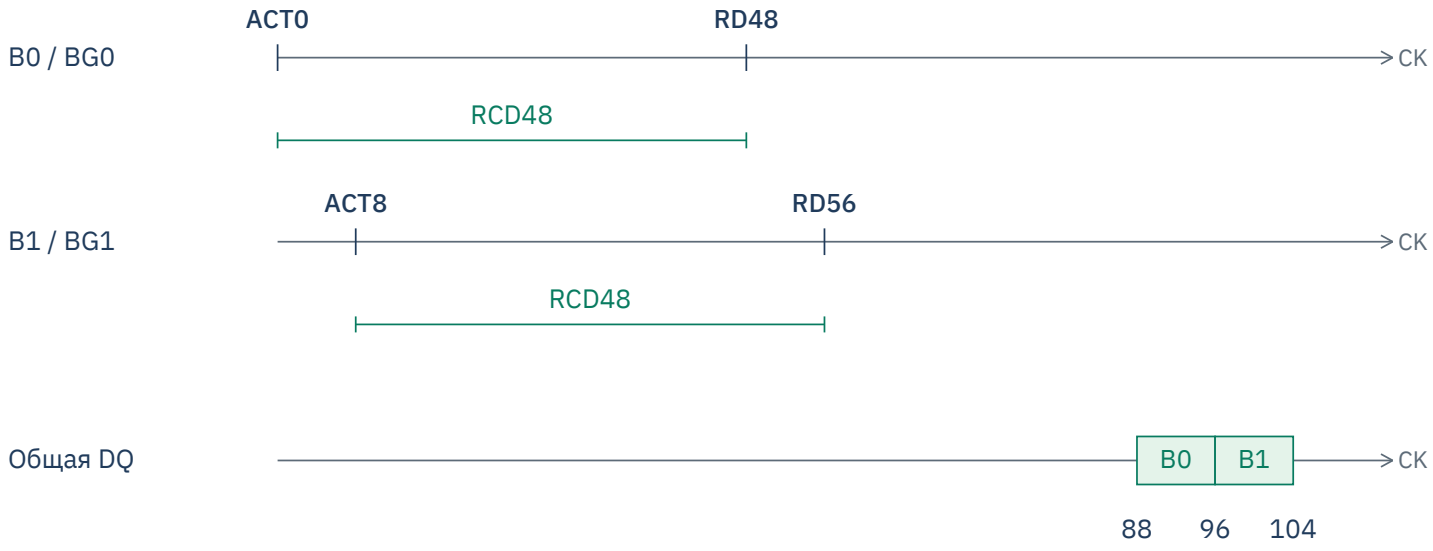
14.1. Работа двух банков на общей временной шкале

Рассмотрим два разных банка в разных BG одного ранга. B0 начинает активацию в CK0. В CK8 можно начать активацию B1, если выполнены tRRD, tFAW и остальные условия. При принятом для примера tRCD = 48 первый RD разрешён в CK48, второй — в CK56. Подготовка второго банка начинается на 8 тактов CK позже первой и 40 тактов CK идёт параллельно с ней.

Пусть CL = 40, BL16, а интервал между командами READ для выбранных разных BG равен 8 тактов CK. Тогда данные B0 занимают интервал CK88–96, B1 — CK96–104. Пакеты занимают шину последовательно, хотя подготовка банков частично шла параллельно.

Независимые состояния банков позволяют контроллеру перекрывать часть внутренних задержек полезной работой другого банка. По общей линии DQ пакеты передаются последовательно.

Рисунок 26. Подготовка двух банков с перекрытием



Учебный пример: B0/B1 в разных BG одного ранга; BL16 без CRC; RCD = 48, CL = 40, RRD_S = 8, RD→RD = 8 (такты CK). Ограничение FAW соблюдено; пакеты последовательно занимают общую DQ.

14.2. Чередование обращений, локальность и очередь запросов

Чередование, interleaving, распределяет обращения между банками, рангами или подканалами, чтобы заранее готовить ресурсы для следующих обращений. Насколько это помогает, определяют адресная карта и поток запросов.

Row locality (повторное использование открытой строки) помогает сократить число смен строк. Слишком частое переключение между строками одного банка добавляет PRE/ACT и мешает плотному потоку данных. С другой стороны, сосредоточение всех запросов в одном банке может ограничить параллелизм. Планировщик сочетает обе возможности в пределах имеющейся очереди.

В зависимой цепочке чтений адрес следующего элемента может находиться в данных предыдущего. Тогда контроллер узнаёт его только после завершения предыдущего чтения. Поток независимых запросов, напротив, позволяет держать несколько операций в работе. Поэтому две нагрузки с одинаковым суммарным объёмом чтения способны по-разному использовать один и тот же DIMM.

Контроллер может накопить несколько WR и выполнить их подряд, сократив число смен направления DQ. Запрос на чтение из того же ресурса тогда ждёт окончания серии и допустимого перехода WRITE→READ. Так возникает выбор между меньшими потерями при смене направления и меньшей задержкой отдельного чтения. Размер очереди, приоритеты и длина серии зависят от контроллера.

14.3. Параллелизм рангов и подканалов

Другой ранг предоставляет собственные массивы и открытые строки. Пока один занят внутренней операцией, контроллер иногда может обслужить другой. Но смена ранга может потребовать дополнительных интервалов, а общие линии данных всё равно передают согласованный поток.

Отдельный подканал имеет свой интерфейс команд и данных и позволяет независимо выполнять больше операций. Общий поток могут ограничить очереди IMC, межсоединения CPU или число параллельных запросов в самой нагрузке.

Регенерацию нужно вписать в этот же график. REFab закрывает доступ к выбранному набору банков на заданный интервал, REFsb оставляет часть банков доступной при соблюдении ограничений. Производительность зависит от того, есть ли в это время другая полезная работа, команды для которой контроллер действительно может выдать.

Углубление: логические ранги 3DS

В 3DS различают SLR, тот же логический ранг внутри стека, и DLR, другой логический ранг. Эти обозначения не заменяют dr — переход к другому рангу в терминологии Intel, разобранный в разделе 16.1. В таблице 68 SK hynix tRFC*_dlr ограничивает переход к регенерации другого логического ранга и при указанных условиях короче полного tRFC исходного ранга. Истечение короткого интервала не означает, что первый ранг закончил регенерацию и доступен для обычного обращения. Следующая ACT в другом логическом ранге подчиняется отдельному правилу.

15. Задержка, пропускная способность и нагрузка

Задержка показывает, сколько ждёт отдельный запрос, а пропускная способность — сколько данных проходит за секунду. Одна настройка может влиять на эти показатели по-разному.

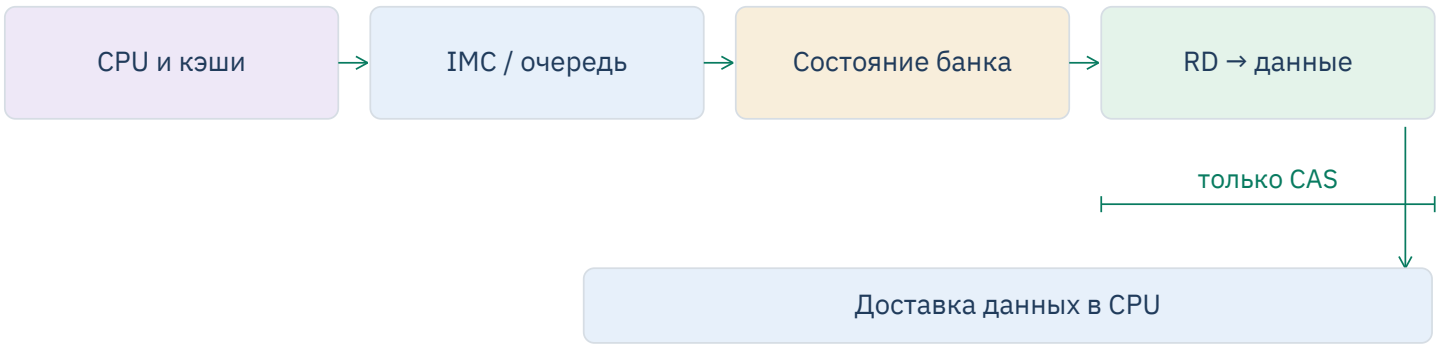
15.1. Задержка CAS и полное время выполнения запроса

В формуле CL × tCK, где tCK — длительность такта, время отсчитывается от READ, уже выданной памяти. Обработка запроса программы началась раньше: она включала поиск данных в кэшах, прохождение по внутренним соединениям, ожидание в очередях и выбор команды. Если потребовалась новая строка, добавились операции в банке. После появления данных на интерфейсе нужно доставить их потребителю в процессоре.

Даже при одинаковых CL и частоте два чтения могут иметь разную задержку. Одно попало в открытую строку при свободной шине, другое ждёт смены строки, окончания записи или регенерации. От потока обращений зависят и среднее время, и распределение задержек.

Формула tRP + tRCD + CL описывает только упрощённую последовательность действий при конфликте строк, когда PRE уже допустима, а остальные ресурсы свободны. Сумма всех первичных таймингов не описывает каждый запрос; к тому же часть условий действует одновременно: команда должна дождаться самой поздней из временных границ, рассчитанных от соответствующих событий.

Рисунок 27. Что остаётся вне интервала CAS



Этапы логические, масштаб не временной. Состав пути зависит от кэшей, состояния банка, режима платформы и нагрузки.

15.2. Пропускная способность интерфейса

Теоретическая пропускная способность интерфейса равна скорости передач, умноженной на полезную ширину. Для 6000 MT/s и 32 бит получается 24 ГБ/с в десятичном исчислении. Два таких подканала дают суммарно 48 ГБ/с, если оба непрерывно передают полезные пакеты.

Командные интервалы, смена направления, подготовка строк и регенерация оставляют на шине паузы. Кроме того, программа может не успевать подавать независимые запросы.

Время работы двунаправленной DQ делится между чтением и записью: предельную пропускную способность подканала нельзя одновременно получить в каждом направлении. Объём полезных данных программы тоже может отличаться от трафика интерфейса из-за кэш-линий, предвыборки и дополнительных передач.

6000 × 10^6 передач/с × 32 бит / 8 = 24 × 10^9 байт/с

Пусть каждый следующий адрес становится известен только после завершения предыдущего обращения. Одно обращение возвращает блок 64 байта, его полная задержка условно равна 60 нс, а перекрытия обращений нет. Тогда поток возвращаемых блоков ограничен следующим расчётом.

64 байта / 60 нс ≈ 1,07 ГБ/с

Здесь считаются все 64 байта возвращаемого блока; полезная алгоритму часть может быть меньше. Независимые запросы позволяют перекрывать ожидание и плотнее занимать интерфейс. Intel Memory Latency Checker использует зависимые чтения для измерения задержки; 60 нс в нашем примере — условное значение.

15.3. Параметры памяти на критическом пути запроса

Для зависимого чтения с частыми сменами строк особенно заметны этапы доступа к новой строке и системный путь. Для потока из открытых строк важнее плотность столбцовых команд и занятость DQ. Для смешанных чтений и записей — распределение операций по направлениям и смена направления. Для частых активаций — RRD/FAW и готовность банков.

Уменьшение тайминга помогает, пока именно он задерживает нужное действие. Затем ограничением может стать другой параметр. Даже заметное ускорение обмена с памятью иногда мало меняет общее время программы.

Под нагрузкой задержка может расти из-за очереди, даже когда память передаёт много данных в секунду. Поэтому максимальную пропускную способность и минимальную задержку обычно измеряют в разных сценариях. Intel MLC, например, отдельно измеряет задержку без нагрузки и при разных уровнях генерируемого трафика.

Сценарий	Что стоит проследить	Что ещё влияет
Частые row conflict (конфликты строк)	Допустимость PRE, RP, RCD, row cycle (цикл строки)	Очередь и занятость общей DQ
Поток row hit (попаданий в открытую строку)	CCD, пакет, общая DQ	Число независимых запросов
Чтения + записи	Смена направления и группировка операций одного направления	Между WR и READ нужно учесть CWL, передачу пакета и WTR
Много ACT	RRD, FAW, готовые банки	Организация DRAM и требования устройства
Плотная очередь	Параллелизм, конкуренция, регенерация	Характер запросов и их распределение по ресурсам

15.4. Интерпретация результатов тестирования

Тест задаёт конкретные адреса, зависимости, число потоков, долю чтений и записей, работу предвыборки и объём данных относительно ёмкости кэшей. Результат относится к этой нагрузке и конфигурации.

Тест чтения видит данные уже после on-die ECC. Поэтому отсутствие несовпадений не доказывает, что внутри массива не было исправленных одиночных ошибок; число видимых ошибок также нельзя напрямую приравнять к числу исходных ошибок ячеек. Счётчики ECS, если платформа даёт к ним доступ, относятся к определённому обходу и фильтрам. Само снижение пропускной способности не измеряет число исправлений ECC: для такого вывода нужны отдельные наблюдаемые признаки.

Теория позволяет предсказать, какой ресурс должен освободиться раньше: нужно проследить изменённый интервал и запросы, в которых он участвует. Практическая проверка покажет, произошло ли это и сохранилась ли корректность данных. Величину ускорения измеряют на выбранной нагрузке.

Более высокая частота может уменьшать длительность одного такта CK и увеличивать предельную скорость передачи данных, но режим контроллера, кодировки и требования в наносекундах способны менять результат. Поэтому при сравнении профилей XMP (наборов настроек памяти) нужно учитывать весь подтверждённый режим.

16. Контроллер Intel: третичные тайминги, RTL и параметры регенерации

Поля контроллера задают расписание работы DRAM и настройку обмена. Их связь со стандартными таймингами зависит от платформы.

16.1. Обозначения RDRD, WRWR, RDWR и WRRD

У Intel семейства tRDRD, tWRWR, tRDWR и tWRRD обозначают интервалы для четырёх последовательностей столбцовых команд: чтение→чтение, запись→запись, чтение→запись и запись→чтение. Суффикс уточняет адресную область. Это поля контроллера Intel, поэтому их определения привязаны к платформе. Далее используются TC_RDRD, TC_WRRD и TC_RDWR из документа ID 764981, Rev. 1.3; TC_WRWR — из ID 767624, Rev. 1.3.

sg означает same group, ту же группу банков; dg — different group, другую группу. Для рассматриваемого определения Intel dr означает другой ранг того же DIMM, dd — другой DIMM в канале. Переход dr относится к смене ранга; микросхемы внутри одного ранга работают совместно по разрядам.

Если в соответствующем канале установлен один DIMM, перехода к другому DIMM нет. Но для двух рангов этого единственного модуля переход dr всё ещё может использоваться. И наоборот, два 1R-модуля в одном канале создают переход dd, хотя каждый из них одноранговый. Конкретная разводка подканалов и термины регистра должны соответствовать платформе.

В рассмотренных регистрах Intel DDR5 эти интервалы измеряются в единицах tCK — длительности периода тактового сигнала CK, а RTL — в тактах контроллера QCLK. Области sg/dg можно сопоставлять с областями ограничений DRAM. Численные значения могут отличаться от tCCD_L/S или tWTR_L/S: контроллер может учитывать дополнительные условия и иначе представлять интервал.

Направление	Связь с теорией DRAM	Что проверять
tRDRD_*	Интервал между командами READ	BG/банк, пакет, режим
tWRWR_*	Интервал между командами WRITE	WR/WR2, частичная запись, область
tRDWR_*	READ → WRITE	Данные, DQS, смена направления
tWRRD_*	WRITE → READ	CWL, пакет, интервал после записи
*_dr / *_dd	Смена ранга / DIMM	Источник данных, ODT, топология

16.2. RTL, QCLK и тренировка памяти

Round-trip latency (задержка от выдачи команды до возврата данных), RTL, в рассматриваемом Intel-регистре измеряет время от READ до первых данных, вернувшихся в контроллер. В него входит время прохождения через интерфейс контроллера и плату, которое не учитывается в CL. Предшествующие очереди и уровни кэшей в RTL не входят.

Значение поля SC_ROUNDTRIP_LATENCY измеряется в тактах контроллера QCLK. При подтверждённых Gear и частотах tQCLK = Gear × tCK. Например, 30 тактов QCLK в Gear 2 при 6000 MT/s дают 30 × 2 × 2000 / 6000 = 20 нс. Это пересчёт поля для указанных условий. Определения SC_ROUNDTRIP_LATENCY, E020, и MC_BIOS_REQ, 5E00, приведены в Intel Datasheet Vol. 2, ID 767624, Rev. 1.3.

Утилита может пересчитать число или добавить смещение. Прежде чем применять формулу к её показанию RTL, нужно подтвердить, что она показывает значение поля в тактах QCLK без пересчёта и смещения. Обычные tCL/tRCD/tRP уже заданы в тактах CK, и умножать их на Gear не нужно.

IOL (название настройки задержки тракта ввода-вывода), RTL Offset (смещение RTL) и параметры тренировки выравнивают и настраивают тракт на уровне платформы. Публичное определение RTL не раскрывает их полную механику. Тренировка подбирает рабочие временные положения и запасы в пределах требований DRAM; одного расчётного равенства для этого недостаточно.

Для значения RTL в QCLK без пересчёта: время, нс = RTL × Gear × 2000 / MT/s

16.3. Параметры регенерации в контроллере

В документации регистра Intel TC_RFP, E438 (ID 764981, Rev. 1.3), такт для этих интервалов контроллера обозначен DCLK. Для перевода в физическое время нужна его документированная частота; равенство DCLK, CK и QCLK по названиям не предполагается.

tREFIx9 в Intel TC_RFP задаёт предельный промежуток по правилам контроллера и измеряется в единицах 1024 DCLK. Допустимый перенос регенерации зависит от режима (обычного или FGR) и баланса досрочно выполненных и отложенных команд; из имени поля его предел не выводится.

OREF_RI в рассматриваемом Intel-регистре задаёт в циклах DCLK длительность простоя ранга, после которой возможна opportunistic refresh (регенерация, которую контроллер назначает на свободное время ранга). Поле управляет расписанием. Его нулевое или иное значение не означает отключения обязательной регенерации массива.

Для tCA2CS, tMOD, tWRPRE и некоторых других названий в BIOS точная привязка к регистрам текущей платформы не установлена. Командный интерфейс, регистры режима, восстановление после записи и PRE объясняют связанные процессы, но для расчёта конкретного поля нужна документация платы и контроллера.

17. Сводная модель работы памяти и практическое применение

Параметр BIOS становится понятнее, когда за ним видны команда, ресурс и конкретный интервал ожидания.

17.1. Общая модель последовательности операций

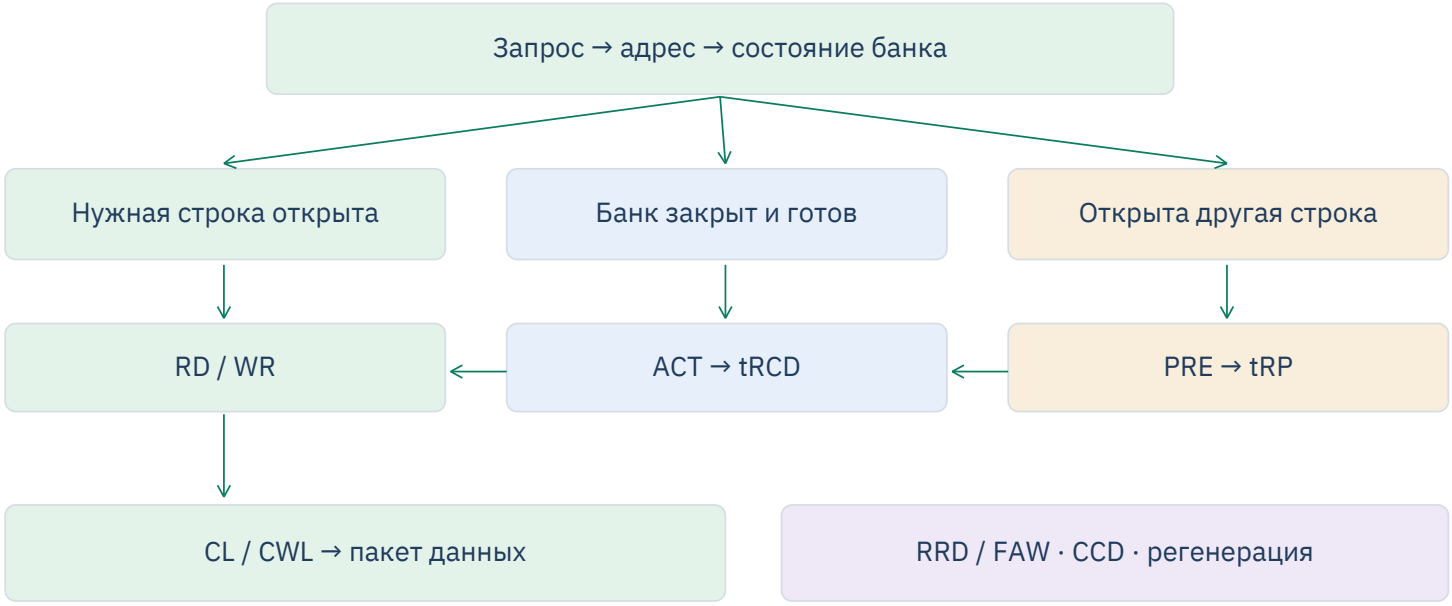
Запрос к DRAM сначала попадает в контроллер памяти. Адресная карта определяет ресурсы, а планировщик проверяет состояние банка и недавнюю историю команд. Если нужная строка активна, можно готовить команду чтения или записи. Если банк закрыт, нужна ACT. Если открыта другая строка, сначала требуется допустимая PRE и подготовка нового цикла.

Для новой строки tRCD определяет, когда разрешён доступ к столбцам. Для закрытия одновременно учитываются tRAS и условия последних чтений и записей; после PRE действует tRP, а перед повторной ACT того же банка нужно выдержать tRC. Между активациями разных банков действуют ограничения RRD и скользящего окна FAW.

RD и WR планируются с учётом CCD, направления, ранга и пакетной передачи. CL задаёт время до начала выдачи данных чтения, CWL — время до начала приёма данных записи. Для WR→RD нужно учитывать завершение записи и интервал после данных записи; для PRE после WR — восстановление. В других банках может идти подготовка к следующим операциям, пока общая DQ передаёт текущий пакет.

В это же расписание контроллер включает регенерацию и переходы между режимами устройства.

Рисунок 28. Общая карта команд и ограничений



$$PRE \geq \max(ACT + RAS,\ RD + RTP,\ CK^* + WR)$$

CK* — первый нарастающий фронт CK после последних данных записи; WR→RD: CWL + пакет + WTR.

Логические связи, без масштаба времени. Восстановление после записи и WTR начинаются после данных записи. Контроллер включает регенерацию в расписание запросов. Места разбора ограничений указаны в таблице ниже.

Группа ограничений	Где разобрана
Последовательные этапы и независимые границы	глава 7
Строка: CL, RCD, RP, RAS, RC	глава 8
Активации разных банков: RRD, FAW	глава 9
Столбцовые команды и пакет: CCD, BL	глава 10
Запись и смена направления: CWL, WTR, WR, RTP, WTRA	глава 11
Регенерация: REFI, RFC, REFsb, долг	глава 12
Режимы и служебные операции: PD, SR, MRR/MRW, ZQ	глава 13
Параллельная работа банков, рангов и подканалов	глава 14
Задержка запроса и поток данных	глава 15
Поля и единицы контроллера Intel	глава 16

17.2. Пять критериев анализа таймингов

От какого события идёт отсчёт? Какое следующее действие ограничено? Какого банка, группы, ранга или шины это касается? В каком режиме и в каких единицах задано значение? Какое другое условие может остаться более строгим после изменения?

Для расчёта по параметру платформы нужны публично подтверждённые событие отсчёта, область действия и единица измерения. Пока эти сведения неизвестны, подставлять число поля в формулу нельзя.

Я советую возвращаться к временной шкале, когда формула начинает казаться загадочной. Отметьте команду, начало и конец передачи данных, затем границы допустимости следующего действия. Такая схема быстро показывает, где нужны сумма, максимум, смена области либо окно, охватывающее несколько предыдущих событий.

17.3. Материалы для дальнейшего изучения

Практическое руководство «Настройка и оптимизация оперативной памяти DDR5 на платформах Intel LGA1700 / LGA1851» продолжает эту теорию: методы настройки, напряжения, тренировка и проверка стабильности. При работе с ним сопоставляйте каждое изменение с командой, ресурсом и ограничением, которые оно затрагивает.

Приложение A. DDR3 → DDR4 → DDR5: развитие архитектуры памяти

DDR3, DDR4 и DDR5 различаются скоростью передачи и устройством памяти. Менялись организация банков, размер внутренней выборки и распределение запросов по интерфейсам. Эти изменения определяют, сколько обращений память может обслуживать одновременно и как заполняется шина данных.

A.1. Общие принципы и различия поколений DDR

У DDR3, DDR4 и DDR5 сохраняется знакомая последовательность: открыть строку, обратиться к её данным, при необходимости закрыть её и перейти к другой. Ячейки теряют заряд, поэтому содержимое приходится восстанавливать. Знания об ACT, READ, WRITE, PRE и регенерации из глав 2 и 6 остаются полезными во всех трёх поколениях.

Меняется организация этой работы. В сравнительной таблице число банков относится к одному монолитному устройству DRAM, а ширина полезной шины — к обычному UDIMM без системной ECC. Связь между устройствами, рангом и подканалом разобрана в главе 3.

Для сравнения взяты Micron DDR3 2 Гбит, Micron DDR4 8 Гбит и SK hynix DDR5 32 Гбит. Число банков зависит и от плотности: например, у DDR5 8 Гбит из таблицы 1 SK hynix банков вдвое меньше, чем у рассматриваемых 32 Гбит. У x4/x8 это 8 BG по 2 банка, у x16 — 4 BG по 2.

Механизм / область	DDR3	DDR4	DDR5
Внутренняя предвыборка DRAM	8n (8 бит на линию DQ)	8n (8 бит на линию DQ)	16n (16 бит на линию DQ)
Банки одного чипа DRAM x4/x8 в выбранных даташитах	2 Гбит: 8 банков, без BG	8 Гбит: 4 BG × 4 = 16 банков	32 Гбит: 8 BG × 4 = 32 банка
Банки одного чипа DRAM x16 в тех же даташитах	2 Гбит: 8 банков, без BG	8 Гбит: 2 BG × 4 = 8 банков	32 Гбит: 4 BG × 4 = 16 банков
Полезный интерфейс UDIMM без системной ECC	Один путь 64 DQ	Один путь 64 DQ	Два подканала по 32 DQ
Основные режимы burst (пакетной передачи)	BL8; BC4 фиксированный / с выбором командой	BL8; BC4 фиксированный / с выбором командой	BL16; BC8 с выбором командой

BC — burst chop (укороченная передача). В фиксированном режиме BC4 длина задана заранее; при выборе «на лету» команда определяет полную или укороченную передачу в разрешённом режиме. В даташите DDR5 SK hynix BL32 доступен только для x4; общий пример в этом приложении использует x8.

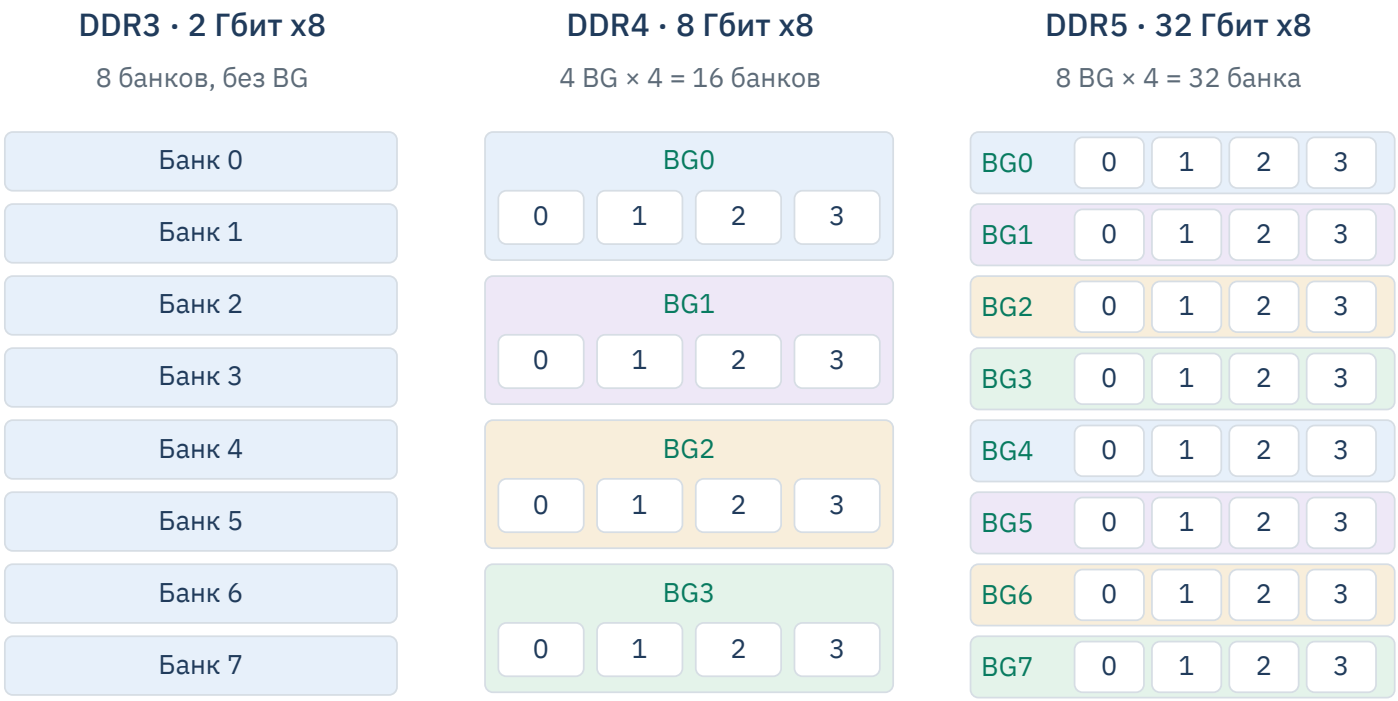
A.2. DDR3 → DDR4: изменения архитектуры при сохранении предвыборки 8n

Несколько банков позволяют готовить следующее обращение, пока другое ещё не завершено. Данные выбранных банков выходят через общий интерфейс устройства. В DDR4 банки объединены в группы банков, BG: для следующей команды имеют значение и готовность банка, и его группа.

В выбранных устройствах DDR3 восемь банков без BG. У DDR4 x4/x8 — четыре группы по четыре банка, вдвое больше банков; у DDR4 x16 — две группы и те же восемь банков.

В применимых ограничениях DDR4 буквы S и L различают переходы между разными и внутри одной BG. Например, tCCD связывает две столбцовые команды чтения, tRRD — две ACT к разным банкам. Для чтений в одной группе проверяется tCCD_L, в разных — tCCD_S; для ACT аналогично учитывается область tRRD_L/S. Кроме этих минимумов остаются занятость DQ, состояние банков и общие ограничения активаций.

Рисунок 29. Банки x8: три конкретные организации



Число банков указано для одного чипа DRAM.

По даташитам Micron DDR3 2 Гбит, Micron DDR4 8 Гбит и SK hynix DDR5 32 Гбит. В каждой колонке — одно монолитное x8-устройство; цифры внутри группы — номера банков. Все BG устройства используют его общий интерфейс DQ.

Чередование BG позволяет уплотнить поток команд: контроллер выбирает готовые к выполнению запросы в другой группе, пока для прежней ещё действует более длинный интервал. Для этого в очереди должны быть подходящие запросы. Зависимую цепочку чтений дополнительные банки распараллелить не могут.

DDR4 при этом сохранила 8n-prefetch (внутреннюю выборку восьми битов на каждую линию DQ). Новые возможности связаны с организацией банков и расписанием доступа. У более длинного пакета есть обратная сторона: по объяснению Micron, передавать им малые порции данных может быть неэффективно.

A.3. DDR4 → DDR5: переход к двум подканалам

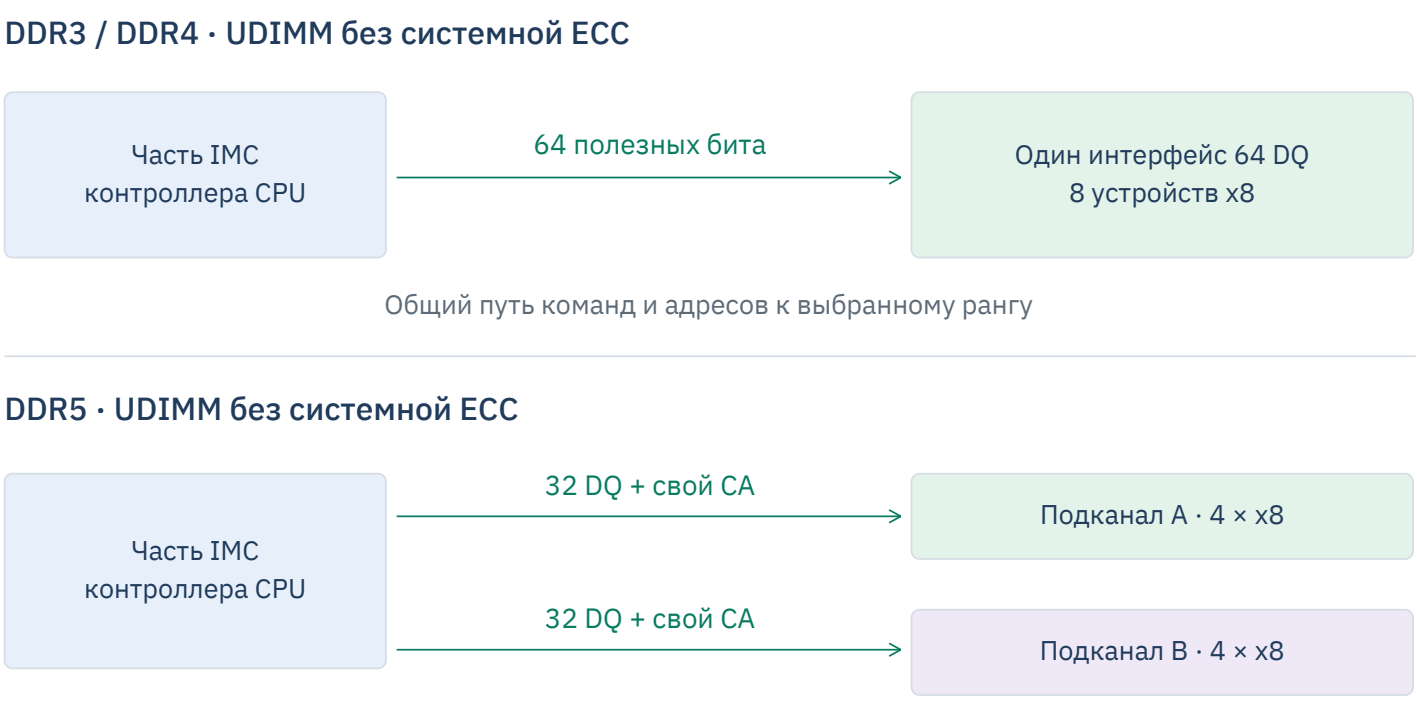
У обычного DDR3/DDR4 UDIMM без системной ECC данные выбранного ранга идут по 64 полезным линиям. В DDR5 такой модуль разделён на два подканала по 32 полезных линии. У каждого свой интерфейс команд и адресов и свой путь данных. Контроллер может планировать обращения к ним отдельно.

В примере с устройствами x8 для одной передачи по 64 линиям совместно работают восемь микросхем. В каждом 32-битном подканале DDR5 — четыре. Это объясняет ширину обмена; ёмкость модуля зависит ещё и от плотности устройств и числа рангов.

Второй подканал даёт ещё один путь для обращений: одна последовательность команд не обязана повторять другую. При наличии подходящих запросов передача в одном подканале может перекрываться с подготовкой или передачей в другом. Но планировщик, ресурсы CPU, питание и часть условий платформы остаются общими.

Два подканала DIMM и два канала контроллера — разные уровни организации. Число доступных полезных линий и схема их использования зависят от CPU, платы и установленных модулей.

Рисунок 30. 64 полезные линии и два подканала по 32



Обычные UDIMM без системной ECC, с устройствами x8 и одним выбранным рангом в каждом интерфейсе. Линии сгруппированы. У DDR5 два пути команд/адресов и данных; они подключены к общему IMC.

A.4. Передача 64 байт по 32 линиям данных при BL16

Более узкий путь может передать тот же блок за большее число передач. Для полного пакета DDR3/DDR4 возьмём 64 полезные линии и BL8: восемь передач по восемь байт дают 64 байта. Для одного подканала DDR5 возьмём 32 линии и BL16: шестнадцать передач по четыре байта дают те же 64 байта.

BL, burst length (длина пакета), обозначает число передач на каждой линии DQ. Объём в байтах и длительность в полных периодах тактового сигнала CK рассчитываются отдельно. За один такт CK происходят две передачи. Поэтому BL8 занимает четыре такта CK, а BL16 — восемь тактов CK внешнего пакета без дополнительных режимов. Число байтов получается только после учёта ширины полезного интерфейса.

Prefetch (внутренняя предвыборка) описывает другую сторону этой связи: сколько данных DRAM подготавливает внутри для последовательной выдачи. У базовой архитектуры DDR3/DDR4 это 8 бит на каждую внешнюю линию DQ, у DDR5 — 16. Для полного базового пакета эта внутренняя порция согласуется с BL8 или BL16. Укороченные BC4/BC8 меняют длину внешней передачи при прежней архитектуре внутренней выборки. В записи 4n/8n-prefetch числа означают четыре или восемь битов внутренней выборки на линию DQ.

Предвыборка CPU работает иначе: процессор заранее запрашивает адреса, которые могут понадобиться программе. 8n/16n-prefetch в DRAM описывает внутреннюю подготовку данных для выдачи по DQ.

Половина прежней ширины при удвоенной длине пакета сохраняет порцию 64 байта, а второй подканал может обслуживать другое обращение. Насколько полно используются оба подканала, зависит от потока запросов.

DDR3 / DDR4: 64 бита × BL8 / 8 = 64 байта
DDR5, один подканал: 32 бита × BL16 / 8 = 64 байта

> Сравниваются полные пакеты без дополнительных ECC-битов и CRC (контрольного кода передачи). 64 байта — выбранный учебный блок. Геометрия внутренней строки и детали пакетной передачи разобраны в разделе 3.4 и разделе 10.1; размер строки значительно больше этой порции.

Рисунок 31. Одинаковые 64 байта, разное число передач



Учебное сравнение полных пакетов: DDR3/DDR4 — 64 DQ, BL8; DDR5 — один подканал 32 DQ, BL16. Без дополнительных ECC-битов и CRC. Ширина полосы показывает объём — одинаковые 64 байта; длительность передачи определяется длиной пакета и периодом сигнала CK. Числа в ячейках — номера передач.

A.5. Обозначения таймингов и различия условий их применения

Основные ограничения знакомы по всем трём поколениям: строке нужно время на подготовку, банку — на закрытие, интерфейсу — на передачу данных. Их точные значения и события отсчёта зависят от протокола и режима.

Таблица ограничений описывает обычный доступ к выбранному банку. Для DDR3/DDR4 заданы additive latency AL (дополнительная задержка) = 0 и полные пакеты BL8, для DDR5 — BL16. Дополнительные задержки и CRC исключены. За момент команды принята её опорная точка на диаграмме производителя. У DDR5 READ занимает два командных цикла, а CL отсчитывается от второго.

Ограничение	Начальное → конечное событие	Область и смысл сравнения
CL	Опорная точка READ → номинальное начало выдачи данных	Выбранное чтение. При AL = 0 это CAS-часть read latency (задержки чтения); не весь запрос CPU.
tRCD	ACT → допустимое обращение READ/WRITE к открываемой строке	Тот же банк. Для DDR3/DDR4 AL = 0; правила DDR5 и раздельные поля BIOS уточняются в разделе 8.2.
tRP	PRE → следующая допустимая ACT	Тот же банк: завершение подготовки к новой строке.
tRAS	ACT → допустимое начало PRE	Тот же банк: минимальное активное время. Сам по себе этот минимум не отменяет условий после чтения/записи.
tRC	ACT → следующая ACT	Тот же банк: полный row cycle (цикл строки); не интервал активаций разных банков.

A.6. Изменения межбанковых ограничений и интервалов после записи

В DDR3 нет групп банков, а DDR4 и DDR5 различают переходы внутри BG и между BG. tRRD связывает ACT → ACT к разным банкам выбранного устройства/ранга; tCCD в этом примере — READ → READ. Для DDR4/DDR5 выбирается ограничение нужной области BG; общая шина DQ и окно tFAW действуют одновременно с ним. Подробные условия DDR5 — в главах 9 и 10.

tWR и tWTR начинаются не в момент выдачи WRITE. Для полного BL8 DDR3/DDR4 и BL16 DDR5 в выбранном базовом режиме опорой служит первый нарастающий фронт тактового сигнала CK после последних данных записи. Конечное событие tWR — допустимая PRE того же банка; для tWTR — допустимая READ в области выбранного устройства. В DDR4/DDR5 область BG дополнительно выбирает применимое ограничение WTR. Сначала идут write latency (задержка записи) и сами данные, затем соответствующий интервал.

У фиксированного BC4 и BC4 с выбором укороченной передачи самой командой в рассмотренных DDR3/DDR4 различается внешний отсчёт восстановления записи. Поэтому здесь сравниваются полные пакеты. Опорные события DDR5 подробно показаны в главе 11.

tRTP связывает внутреннюю READ с разрешением PRE того же банка. В DDR3/DDR4 при включённой AL внешняя READ выдаётся раньше внутренней, поэтому интервал от внешней READ до PRE включает AL. В таблице раздела A.5 принята AL = 0.

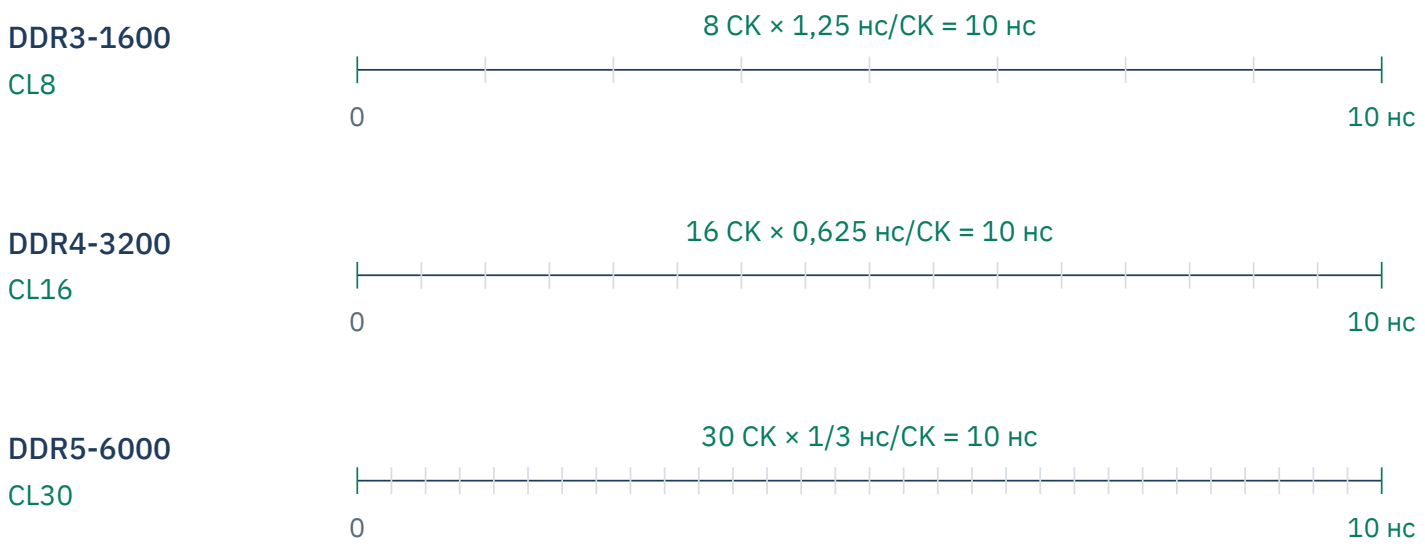
У DDR5 есть дополнительные ограничения для разных последовательностей записи и режимов; они разобраны в разделе 10.3. Соответствие этих ограничений полям контроллера рассматривается в главе 16.

A.7. CL8, CL16 и CL30: примеры задержки CAS 10 нс

Число CL нужно сравнивать вместе с длительностью такта CK. По формуле из главы 4, tCK = 2000 / MT/s в наносекундах. Для подобранных примеров DDR3-1600 CL8, DDR4-3200 CL16 и DDR5-6000 CL30 интервал CAS одинаков: 10 нс.

Примеры подобраны для арифметического сравнения CAS, а не как JEDEC-профили или рекомендации настроек. Расчёт выполнен для базового режима с AL = 0 у DDR3/DDR4. В рассматриваемой DDR5 read DQS offset (смещение строба чтения) меняет положение строба, сохраняя положение данных и этот CL-интервал.

Рисунок 32. Разное число тактов CK укладывается в одни 10 нс



Начало: опорный READ. Конец: номинальное начало данных.

Общая линейная шкала физического времени. Учебные DDR3-1600 CL8, DDR4-3200 CL16 и DDR5-6000 CL30: AL = 0 для DDR3/DDR4, без добавочных режимов задержки; опорная точка DDR5 — второй цикл READ. Мелкие деления — полные периоды тактового сигнала CK; единица «нс/CK» означает наносекунды за такт.

Учебный режим	Один такт CK	Интервал CAS	Передача 64 байт
DDR3-1600 CL8	1,25 нс	8 × 1,25 = 10 нс	64 DQ, BL8: 4 такта CK = 5 нс
DDR4-3200 CL16	0,625 нс	16 × 0,625 = 10 нс	64 DQ, BL8: 4 такта CK = 2,5 нс
DDR5-6000 CL30	1/3 нс	30 × 1/3 = 10 нс	32 DQ, BL16: 8 тактов CK = 8/3 ≈ 2,67 нс

A.8. Четыре показателя времени и скорости

Интервал CAS заканчивается в момент начала выдачи данных. Длительность передачи блока описывает последующий пакет: при двух передачах за такт CK это BL/2 полных тактов. Поэтому равные 10 нс до начала выдачи в предыдущем примере соседствуют с разным временем передачи 64 байт. Слоты считаются целиком; последняя передача начинается на полтакта раньше конца слота.

Обработка запроса CPU начинается ещё раньше: она может включать промахи кэша, ожидание в очереди, подготовку строки и прохождение через контроллер. Эти этапы не входят в CL. А устойчивая пропускная способность описывает длинный поток с учётом зависимостей программы и пауз на активации, смену направления и регенерацию.

В идеальном непрерывном потоке предел полезной пропускной способности в МБ/с равен MT/s × число DQ / 8; для ГБ/с результат делится на 1000. При заданных режимах это 12,8 ГБ/с для 64 DQ DDR3-1600, 25,6 ГБ/с для 64 DQ DDR4-3200 и 24 ГБ/с для одного подканала 32 DQ DDR5-6000. Два одновременно полностью занятых подканала дают 48 ГБ/с. Паузы и зависимости запросов уменьшают доступную программе пропускную способность; подробнее в главе 15.

A.9. Способы организации регенерации

Все три поколения нуждаются в регенерации. Меняются способы вписать обслуживание в поток работы. В рассматриваемой DDR3 обычная REF занимает все банки устройства, а self-refresh поддерживает данные во внутреннем режиме. Частота обслуживания зависит от температуры и конкретного устройства.

У рассмотренной DDR4 Fine Granularity Refresh (регенерация более короткими и частыми операциями) позволяет выбирать режимы 1×, 2× и 4×: более частое обслуживание с меньшим отдельным tRFC. Сокращение tRFC не обязано быть пропорциональным росту частоты команд, поэтому суммарное время обслуживания может не уменьшиться.

В DDR5 SK hynix из этого сравнения используются обычный режим и FGR с удвоенной частотой соответствующих REFab при одинаковой температуре. Кроме REFab для всех банков, в FGR доступен REFsb: он выбирает одинаковый номер банка во всех BG устройства. Остальные банки могут работать с ограничениями.

tRFC отсчитывается от принятой команды регенерации до разрешённых следующих действий в её области; tREFI описывает допустимое среднее расписание этих команд. У DDR4 и DDR5 различаются наборы режимов и правила их смешения. Температурное учащение регенерации и выбор FGR тоже действуют отдельно. Детали DDR5 — в главе 12.

A.10. Питание, защита данных и настройка интерфейса

Номинальные VDD/VDDQ снижаются от 1,5 В у DDR3 до 1,2 В у DDR4 и 1,1 В у DDR5. Это уменьшает одну из составляющих затрат на переключение. Температура модуля зависит также от ёмкости, числа устройств, скорости, загрузки, питания и охлаждения. VDD питает основные цепи DRAM, VDDQ — интерфейс ввода-вывода; в приведённых DDR4/DDR5 отдельная линия VPP используется для активации строк.

Обычный DDR3/DDR4 UDIMM получает необходимые напряжения питания от платы. В клиентском DDR5 основные напряжения формирует PMIC на самом модуле: типичный вход — 5 В, VDD/VDDQ/VPP формируются локально.

В DDR5 добавлена штатная on-die ECC. Она помогает исправлять ошибки внутреннего массива, но не защищает внешнюю передачу между DIMM и контроллером. Системная ECC требует отдельной поддержки памяти, контроллера и платформы. Работа ECC, ECS и CRC разобрана в разделе 10.5.

Настройка интерфейса, training, развивалась во всех трёх поколениях. DDR3 поддерживает write leveling (выравнивание строба записи по тактовому сигналу), DDR4 развивает настройку интерфейса, в том числе VrefDQ — опорного напряжения для приёма DQ. В DDR5 есть обучение команд/адресов, более сложные шаблоны чтения и дополнительные средства приёма сигнала. В даташите SK hynix сложные шаблоны связаны с повышением скорости обмена; DFE (decision feedback equalization, коррекция сигнала с обратной связью по уже принятым данным) помогает бороться с межсимвольными помехами.

Область	DDR3	DDR4	DDR5
Номинальные напряжения питания DRAM в выбранных документах	VDD/VDDQ: 1,5 В; DDR3L отдельно: 1,35 В	VDD/VDDQ: 1,2 В; VPP: 2,5 В	VDD/VDDQ: 1,1 В; VPP: 1,8 В
Питание обычного клиентского UDIMM	Линии от платы	Линии от платы	PMIC на модуле; типичный вход 5 В
Внутренняя коррекция	Выбранный даташит не заявляет on-die ECC (внутренняя коррекция)	В сравнении Micron: без on-die ECC (внутренняя коррекция)	on-die ECC (внутренняя коррекция) внутри DRAM; внешний путь остаётся отдельным
Системная ECC	Зависит от модуля, IMC и платформы	Зависит от модуля, IMC и платформы	Требует отдельной поддержки, несмотря на on-die ECC (внутренняя коррекция)

Номиналы в таблице питания и защиты относятся к линиям DRAM указанных вариантов памяти. CPU VDDQ и SA — отдельные напряжения.

A.11. Влияние архитектуры на производительность памяти

Поколение памяти задаёт архитектурные возможности: организацию банков, ширину и число интерфейсов, пакетную передачу и режимы обслуживания. Конкретный комплект определяет, какие из них доступны: здесь имеют значение ёмкость, плотность и ширина устройств, ранги, MT/s, тайминги и напряжения.

Результат зависит от всей платформы и потока запросов. Два комплекта DDR5 могут по-разному нагружать контроллер и использовать параллелизм. При сравнении разных поколений к этому добавляются различия CPU и плат; прирост FPS, пропускной способности или сокращение задержки устанавливается для конкретной нагрузки.

Общая логика доступа сохраняется: открыть строку, выдержать ограничения, передать данные и вовремя обслужить массив. У DDR5 меняются организация, события отсчёта и режимы, которые определяют расписание этих действий.

Приложение B. Сокращения и обозначения

Здесь собраны обозначения, встречающиеся в руководстве. Ссылки ведут к содержательным объяснениям; условия конкретного режима остаются в соответствующем разделе.

B.1. Сокращения, команды и регистры

Обозначение	Короткое объяснение	Определение
1T1C	Модель ячейки DRAM: один транзистор доступа и один конденсатор.	раздел 2.1
3DS	Организация DRAM со стеком кристаллов; ограничения могут зависеть от логического ранга внутри стека.	раздел 9.4
ACT / ACTIVATE	Команда активации строки выбранного банка.	раздел 6.1
AP	Автоматическое закрытие банка после чтения или записи с соблюдением условий допустимости PRE.	раздел 6.3
BC / BC4 / BC8	Укороченный пакет; число указывает количество внешних передач на линии DQ. Фиксированный режим и выбор командой различаются.	раздел A.1; раздел 10.1
BG	Группа банков с общими ограничениями для некоторых последовательностей команд.	раздел 3.3
BIOS / UEFI	Встроенное программное обеспечение платы, через меню которого доступны настройки.	раздел 1.4
BL / BL8 / BL16 / BL32	Число передач на каждой линии DQ в одном пакете; объём в байтах зависит также от ширины интерфейса.	раздел 3.7; раздел A.1
CA	Интерфейс команд и адресов между контроллером и DRAM.	раздел 6.1
CK	Тактовый сигнал интерфейса DRAM; tCK — его период.	раздел 4.1
CKD	Буфер, который распределяет тактовый сигнал CK на модуле CUDIMM или CSODIMM.	раздел 3.5
Command Rate	Режим выдачи и приёма команд на платформе; его назначение отличается от отношения частот Gear.	раздел 4.4
CPU	Процессор, выполняющий команды программы и работающий с её данными.	раздел 1.1
CRC	Контрольный код для обнаружения ошибок передачи между контроллером и DRAM.	раздел 10.5
CSODIMM / CUDIMM	Модули SO-DIMM / DIMM с буфером тактового сигнала CKD.	раздел 3.5
DCLK	Такт, которым документация TC_RFP задаёт интервалы контроллера. Не приравнивается автоматически к CK или QCLK.	раздел 16.3
dd / dg / dr / sg	Суффиксы Intel: другой DIMM; другая BG; другой ранг того же DIMM; та же BG.	раздел 16.1
DDR / DDR3 / DDR3L / DDR4 / DDR5	DDR означает две передачи данных за период CK; цифра указывает поколение. DDR3L — вариант DDR3 с пониженным номинальным напряжением.	раздел 4.1; раздел A.10

Обозначение	Короткое объяснение	Определение
DFE	Средство подавления межсимвольных помех в приёмнике данных.	раздел A.10
DIMM	Модуль памяти с устройствами DRAM, соединениями и служебными компонентами.	раздел 3.1
DLL	Узел подстройки задержки; для его готовности после выхода из режима может требоваться отдельное время.	раздел 13.2
DLR / SLR	Другой / тот же логический ранг в организации 3DS; область ограничения задаётся относительно исходной операции.	раздел 14.3
DQ / DQS	Линии данных / сопровождающий строб, по фронтам которого приёмник определяет момент приёма данных.	раздел 4.1
DRAM / SDRAM	Динамическая память с периодическим восстановлением заряда; SDRAM использует синхронный командный интерфейс.	раздел 1.4
ECC	Коррекция ошибок с использованием проверочных данных; область защиты зависит от её реализации.	раздел 3.6
ECS	Внутренний обход DRAM с чтением, исправлением ошибок и сохранением исправленного результата.	раздел 10.5
FGR	Режим регенерации более короткими и частыми операциями; температурное учащение задаётся отдельно.	раздел 12.3
Gear	Отношение частот интерфейса памяти и контроллера в рассматриваемой документации Intel.	раздел 4.4
I²C / I3C Basic	Двухпроводные служебные интерфейсы для обмена с компонентами модуля.	раздел 3.5
IMC	Встроенный в процессор контроллер памяти; принимает запросы и формирует расписание команд.	раздел 1.1
IOL / RTL Offset	Параметры настройки тракта и смещения RTL; их полная механика в используемом публичном определении RTL не раскрыта.	раздел 16.2
MLC	Intel Memory Latency Checker — программа для измерения задержки памяти при разных уровнях нагрузки.	раздел 15.3
MR / MR0 / MRR / MRW	Регистр режима DRAM; MR0 — регистр с номером 0; MRR читает регистр, MRW изменяет его.	раздел 13.3
MRC	Код инициализации памяти Intel, участвующий в настройке и тренировке интерфейса.	раздел 7.3
ODECC / on-die ECC	Коррекция одиночной ошибки внутри кодового слова DRAM. Обычная READ в описанном устройстве не сохраняет исправление обратно в массив.	раздел 10.5
ODT	Терминация внутри устройства для согласования линии и уменьшения отражений сигнала.	раздел 13.4
OREF_RI / TC_RFP	OREF_RI в регистре TC_RFP задаёт время простоя ранга перед возможной регенерацией в свободное время.	раздел 16.3
OTF	Выбор поддерживаемого варианта пакетной передачи самой командой, «на лету».	раздел 10.1

Обозначение	Короткое объяснение	Определение
PD / PDE / PDX	Режим пониженного потребления; вход в него; выход. Автономная регенерация в PD не запускается.	раздел 13.1
PHY	Физическая часть интерфейса памяти. В рассматриваемом Intel Gear связывает частоты DDR PHY и QCLK.	раздел 4.4
PMIC	Микросхема, которая формирует напряжения питания модуля и управляет их подачей.	раздел 3.5
PRAC	Подсчёт активаций каждой строки с запросом обслуживания через back-off и команды RFM.	раздел 12.6
PRE / PREab / PREpb / PREsb	Закрытие строки и подготовка банка. ab / pb / sb — все банки / один банк / одноимённые банки всех BG.	раздел 8.3
QCLK	Такт контроллера в рассматриваемой документации Intel; его период связан с tCK через подтверждённый Gear.	раздел 4.4
RAM	Оперативная память, хранящая используемые программой команды и данные.	раздел 1.1
RBL / WBL	Расчётная длина чтения / записи в формулах даташита. Она не всегда равна фактическому числу передач BC8.	раздел 10.1
RD / READ / WR / WRITE / RDA / WRA	Команды чтения / записи порции активной строки; RDA и WRA дополнительно запрашивают автоматическое закрытие банка.	раздел 6.3
REF / REFRESH / REFab / REFsb	Регенерация; REFab обслуживает все банки выбранного устройства, REFsb — одноимённые банки во всех BG.	раздел 12.4
RFM / RFMsb	Обслуживание по активности строк. RFMsb — одноимённые банки всех BG; в выбранном устройстве нужен FGR.	раздел 12.6
RMW	Внутреннее чтение прежних данных, объединение с новыми и запись результата при частичном обновлении.	раздел 10.3
RTL / SC_ROUNDTRIP_LATENCY	Время от READ до первых данных в контроллере; в регистре SC_ROUNDTRIP_LATENCY выражено в тактах QCLK.	раздел 16.2
SPD	Энергонезависимые сведения о модуле; концентратор SPD обеспечивает доступ к ним и служебным компонентам.	раздел 3.5
SR / SRE / SRX	Режим автономного обновления данных; вход в него; выход.	раздел 13.2
SSD	Твердотельный накопитель для долговременного хранения файлов.	раздел 1.1
TCASE	Температура корпуса микросхемы; показания датчика модуля могут отличаться от неё.	раздел 12.3
UDIMM	Модуль без регистра, буферизующего команды и адреса.	раздел 3.2
VDD / VDDQ / VPP; CPU VDDQ / SA	Первые три обозначения относятся к линиям питания DRAM; CPU VDDQ и SA — отдельные напряжения процессора.	раздел A.10
VrefDQ	Опорное напряжение приёма данных по DQ.	раздел A.10
XMP	Профили с наборами настроек памяти; при сравнении учитывается весь подтверждённый режим.	раздел 15.4
ZQ / ZQCal Start / ZQCal Latch	Калибровка выходных цепей и терминации; запуск калибровки и принятие её результата.	раздел 13.4

B.2. Параметры таймингов

Обозначение	Короткое объяснение	Определение
AL	Дополнительная задержка DDR3/DDR4; в сравнительных примерах принята равной нулю.	раздел A.5
CAS / CL / tCL	Историческое имя CAS и задержка от опорной точки READ до начала выдачи данных в выбранном режиме.	раздел 8.1
CWL	Задержка от последнего цикла WRITE до первого эффективного фронта DQS записи без преамбулы.	раздел 11.2
tACTPDEN / tPRPDEN / tREFPDEN	Минимальные интервалы от ACT / PRE / REF до допустимого входа PDE.	раздел 13.1
tCA2CS / tMOD / tWRPRE	Названия полей BIOS, точное соответствие которых регистрам рассматриваемой платформы не установлено.	раздел 16.3
tCCD_*_WTR	Полный интервал WRITE→READ с учётом CWL, расчётного пакета и интервала после записи; звёздочка обозначает область.	раздел 11.3
tCCD_L_WR / tCCD_L_WR2 / tCCD_M_WR	Ограничения WRITE→WRITE; выбор варианта зависит от адресной области, полноты записи и необходимости RMW.	раздел 10.3
tCCD_S / tCCD_M / tCCD_L	Интервалы READ→READ для разных BG / разных банков одной BG / того же банка в указанной таблице SK hynix.	раздел 10.2
tCCD_WTRA	Дополнительная граница WRITE→READ AP для завершения восстановления записи до автоматической PRE.	раздел 11.6
tCK	Длительность одного периода CK; для номинальной скорости MT/s равна 2000 / MT/s наносекунд.	раздел 4.1
tCKE / tCKCKEH / tSR / tXPDLL	Исторические или платформенные имена; точная трактовка требует документации соответствующего регистра.	раздел 13.2
tCKSRX	Требование к стабильности CK перед выходом из self-refresh.	раздел 13.2
tCPDED	Задержка отключения прохождения команд при входе в power-down.	раздел 13.1
tFAW	Скользящее окно четырёх учитываемых активаций; одновременно действуют tRRD и условия готовности банков.	раздел 9.2
tMRD / tMRR / tMRW	Задержка после программирования регистра режима / период чтений / период записей регистров.	раздел 13.3
tMRRPDEN / tMRWPDEN	Минимальные интервалы от чтения / изменения регистра режима до PDE.	раздел 13.1
tOSCO / tOSCOA / tOSCOM	tOSCOA и tOSCOM ограничивают доступ к результату осциллятора DQS; соответствие поля BIOS tOSCO не установлено.	раздел 13.4
tPD / tXP	Ограничение пребывания в power-down / интервал после выхода до разрешённых команд.	раздел 13.1
tQCLK	Длительность такта QCLK; при установленном отношении частот tQCLK = Gear × tCK.	раздел 4.4
tRAS / nRAS	Минимальный ACT→PRE одного банка / его значение в тактах CK; условия после чтений и записей действуют одновременно.	раздел 8.4

Обозначение	Короткое объяснение	Определение
tRC / nRC	Минимальный ACT→ACT того же банка / его значение в тактах CK.	раздел 8.5
tRCD / tRCDRD / tRCDWR	ACT→READ/WRITE; отдельные поля BIOS для чтения и записи требуют сопоставления с требованиями устройства.	раздел 8.2
tRDPDEN / tWRAPDEN / tWRPDEN	Интервалы от чтения / записи с AP / записи до PDE с учётом завершения данных и восстановления.	раздел 13.1
tRDRD_* / tRDWR_* / tWRRD_* / tWRWR_*	Поля Intel для переходов чтение→чтение / чтение→запись / запись→чтение / запись→запись.	раздел 16.1
tREF	Полное окно регенерации массива. В базовом примере Normal REFab при 0–85 °C оно равно 32 мс.	раздел 12.2
tREFI / tREFI1 / tREFI2	Максимальный средний интервал регенерации; индексы 1 и 2 относятся к обычному режиму и FGR.	раздел 12.3
tREFIx9	Поле TC_RFP, ограничивающее промежуток по правилам контроллера Intel; единица поля — 1024 DCLK.	раздел 16.3
tREFSBRD	Интервал от REFsb до допустимой ACT в остальных банках.	раздел 12.4
tRFC / tRFC1 / tRFC2 / tRFCsb	Минимальная пауза после регенерации; вариант зависит от команды, режима и плотности кристалла.	раздел 12.3
tRFC*_dlr	Переход к регенерации другого логического ранга 3DS; не завершает обслуживание исходного ранга.	раздел 14.3
tRFCpb	Название поля BIOS или утилиты; его соответствие DDR5 tRFCsb нельзя установить по одному имени.	раздел 12.4
tRFM	Длительность дополнительного обслуживания RFM.	раздел 12.6
tRP / nRP / tRPab	PRE→ACT того же банка / значение в тактах CK; соответствие отдельного имени tRPab требует проверки.	раздел 8.3
tRRD_S / tRRD_L	ACT→ACT разных банков: в разных BG / внутри одной BG.	раздел 9.1
tRTP	READ→PRE того же банка; CL и длительность пакета отдельно к нему не прибавляют.	раздел 11.5
tRTW / tRDWR	Обозначения перехода READ→WRITE; расчёт зависит от уровня представления, режима и условий интерфейса.	раздел 11.4
tWR	От первого нарастающего фронта CK после последних данных записи до допустимой PRE того же банка.	раздел 11.5
tWTR / tWTR_S / tWTR_L	Интервал от опорного конца записи до допустимой READ; область выбирается по расположению банков.	раздел 11.3
tXS / tXS_DLL / tDLLK	Ожидания после SRX для команд без готовности DLL / с её готовностью; tDLLK задаёт время установления DLL.	раздел 13.2
tZQCAL / tZQLAT	Длительность фоновой ZQ-калибровки / интервал после принятия её результата.	раздел 13.4
tZQCSPeriod / tZQOPER	Поля BIOS, точное соответствие которых регистрам в этом руководстве не установлено.	раздел 13.4

B.3. Единицы измерения и условные обозначения

Обозначение	Короткое объяснение	Определение
1R / 2R / 1Rx8 / 2Rx8	Один / два ранга; x8 дополнительно указывает ширину интерфейса устройства.	раздел 3.2
8n / 16n	Внутренняя предвыборка восьми / шестнадцати битов на каждую внешнюю линию DQ.	раздел 3.7
b / bit / B	Бит / бит / байт; восемь бит образуют один байт.	раздел 1.2
ceil / max	Округление вверх до целого / выбор наибольшего значения.	раздел 4.3
CKn	Граница через n тактов CK от принятого начала отсчёта.	раздел 6.2
Gb / Gbit / Гбит	Обозначения плотности в битах; в рассматриваемой организации DRAM 32 Гбит соответствуют 32×2^{30} бит.	раздел 1.2
KiB / MiB / GiB	КиБ / МиБ / ГиБ: 2^{10} / 2^{20} / 2^{30} байт.	раздел 1.2
MHz / МГц	Миллионы периодов сигнала в секунду.	раздел 4.1
MT/s	Миллионы передач в секунду на одной линии данных.	раздел 4.1
nCK	Число периодов тактового сигнала CK.	раздел 4.1
x4 / x8 / x16	Четыре / восемь / шестнадцать линий данных DQ одного устройства DRAM.	раздел 3.2
ГБ/с	Миллиарды байт в секунду; десятичная единица пропускной способности.	раздел 1.2
нс / мкс	Наносекунда — миллиардная доля секунды; микросекунда — миллионная.	раздел 1.2

Источники и документация

Требования к устройствам — из даташитов, определения регистров — из документации Intel, общие принципы — из исследовательских работ. Полный закрытый стандарт JEDEC напрямую не изучался; новых аппаратных измерений нет.

Устройство DRAM и единицы

• NIST. Prefixes for binary multiples. KiB, MiB, GiB и десятичные единицы. Проверено 07.10.2026.

• Adaptive-Latency DRAM. HPCA 2015. §§ 2.1–2.2: ячейка, считывание, восстановление и предзарядка.

• A Case for Exploiting Subarray-Level Parallelism in DRAM. ISCA 2012. § 2: банк, подмассивы и буфер строки.

• David T. Wang. Modern DRAM Memory Systems. 2005. §§ 4.4.1–4.4.2: нагрузка при активациях и окно FAW.

DDR3, DDR4 и DDR5

• Micron. 2Gb: x4, x8, x16 DDR3 SDRAM. Rev. S, 02.2016. Организация, команды, пакетная передача, питание и регенерация. Копия hstatic.

• Micron. 4Gb: x4, x8, x16 DDR3L SDRAM. Rev. R, 09.2018. С. 1: питание DDR3L. Копия LCSC.

• Micron. 8Gb: x4, x8, x16 DDR4 SDRAM. Rev. M, 10.2017. Организация, группы банков, команды, пакетная передача и регенерация. Копия DigiKey.

• Micron. TN-40-40: DDR4 Point-to-Point Design Guide. Rev. B, 10.2016. 8n, группы банков и электрический интерфейс. Копия oldfriend.url.tw.

• SK hynix. 32Gb DDR5 SDRAM, Rev.1.2, 07.2025. Организация, команды, тайминги, регенерация, ECC и тренировка. Копия UTTC.

• Micron. DDR5 DRAM. Архитектура и сравнение DDR4/DDR5.

Модули и служебные компоненты

• Kingston. DDR5 Memory Standard. MKF-954, 2022. С. 2–3: подканалы, 16n, питание и коррекция ошибок внутри кристалла.

• Micron. MTC16C2085S1UC UDIMM Addendum. Rev. F, 10.2021. С. 1, 7: организация обычного 2Rx8 UDIMM. Копия DigiKey.

• Kingston. KVR16N11/8. Техописание VALUERAM1445-001.D00: 64-битный модуль 2Rx8.

• Kingston. Карточка KVR16N11/8. Классификация DDR3 UDIMM без системной ECC.

• Kingston. KVR26N19S8/8. Техописание VALUERAM1595-001.A00: 64-битный модуль 1Rx8.

• Kingston. Карточка KVR26N19S8/8. Классификация DDR4 UDIMM без системной ECC.

• Rambus. DDR5 SPD Hub with Internal Temperature Sensor. SPD, служебная шина и датчик температуры.

• Rambus. DDR5 Client PMICs. Формирование напряжений и управление питанием модуля.

• Rambus. DDR5 Client Clock Driver. Роль CKD в CUDIMM/CSODIMM.

• NXP. I²C-bus specification, UM10204, Rev.7.0. Двухпроводная шина: SDA и SCL.

• MIPI Alliance. MIPI I3C Basic. Служебный обмен и уведомления по самой шине.

• Kingston. DDR5 Memory Standard: On-die ECC. Области внутренней и системной коррекции ошибок.

• Kingston Glossary. CAS Latency / CL. Расшифровка CAS.

Контроллер Intel

• Intel. Architectural Background: трансляция адресов. ID 775099, Rev. 2.0: адрес программы и физический адрес; без привязки к закрытой карте IMC.

• Intel Core 13-го поколения. SC_ROUNDTRIP_LATENCY, E020. Vol. 2, ID 767624, Rev. 1.3, 13.07.2023: RTL в QCLK.

• Intel Core 13-го поколения. MC_BIOS_REQ, 5E00. Vol. 2, ID 767624, Rev. 1.3, 13.07.2023: поле GEAR и отношение DDR PHY/QCLK.

• Intel Core 13-го поколения. TC_RDRD, E00C. Vol. 2, ID 764981, Rev. 1.3: чтение → чтение.

• Intel Core 13-го поколения. TC_WRRD, E014. Vol. 2, ID 764981, Rev. 1.3: запись → чтение.

• Intel Core 13-го поколения. TC_RDWR, E010. Vol. 2, ID 764981, Rev. 1.3: чтение → запись.

• Intel Core 13-го поколения. TC_WRWR, E018. Vol. 2, ID 767624, Rev. 1.3: запись → запись.

• Intel Core 13-го поколения. TC_RFP, E438. Vol. 2, ID 764981, Rev. 1.3: tREFIx9 в 1024 DCLK; OREF_RI в DCLK.

Планирование, задержка и надёжность

• Rixner et al. Memory Access Scheduling. ISCA 2000. §§ 1–3: планирование команд и локальность.

• Intel. Memory Latency Checker. Измерение задержки без нагрузки и под нагрузкой, а также пропускной способности.

• Vittal, Qureshi. BARD. 2025. § I и § II-D: группирование записей и ожидание чтений.

• Patel, Oliveira, Mutlu. HARP. 2021. § 1 и § 4.2: наблюдение ошибок после коррекции внутри кристалла.

• Canpolat et al. Understanding the Security Benefits and Overheads of Emerging Industry Solutions to DRAM Read Disturbance. 2024. Аннотация v3: PRAC, приостановка команд и RFM.