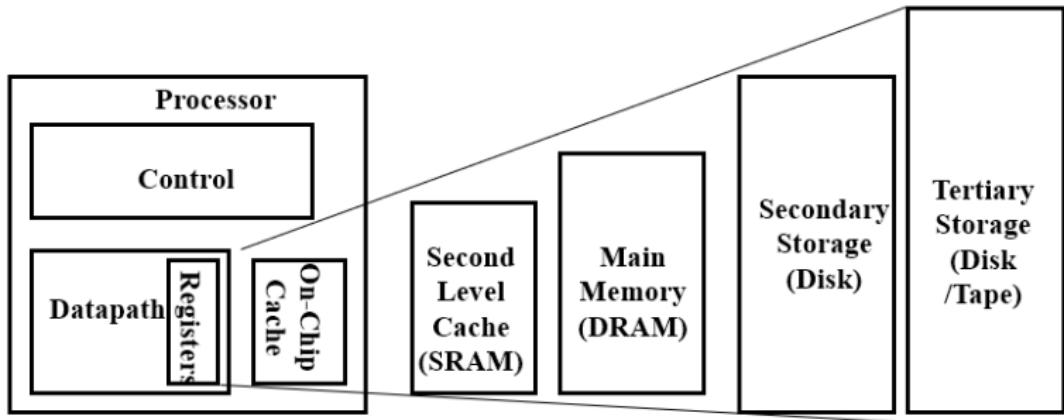


Первинна пам'ять: RAM, SRAM/DRAM, Кеш



©Олег Фаренюк,
Creative Commons - Attribution Share Alike (CC BY-SA)

Види пам'яті

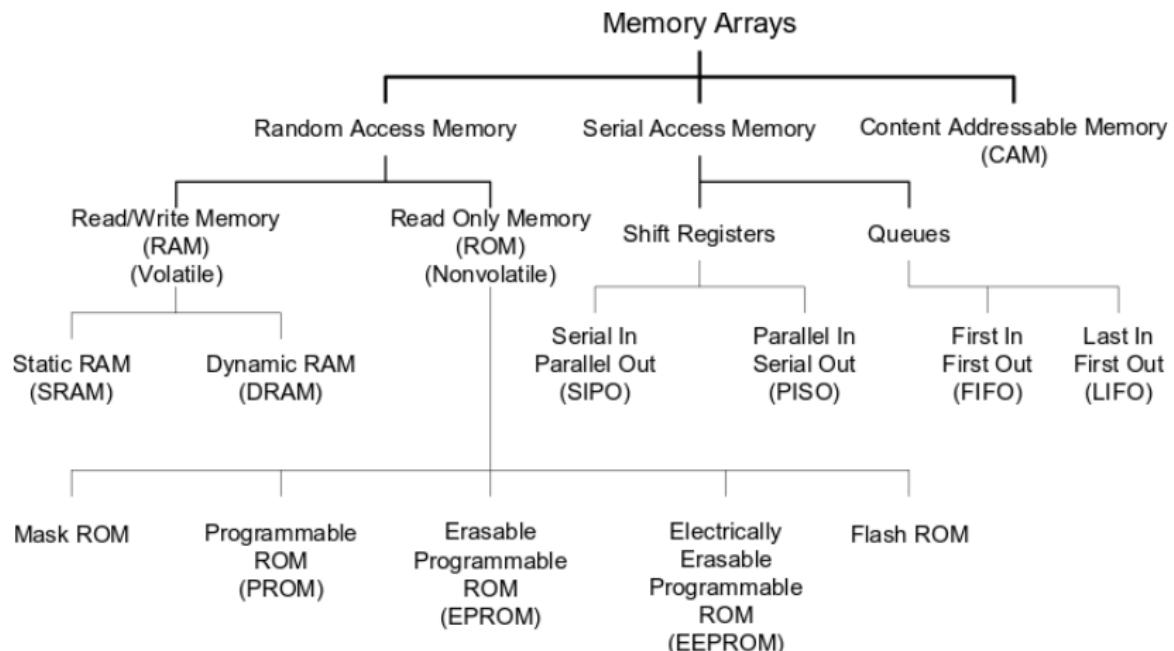


Speed : 1ns Xns 10ns 100ns 10 ms 10 sec

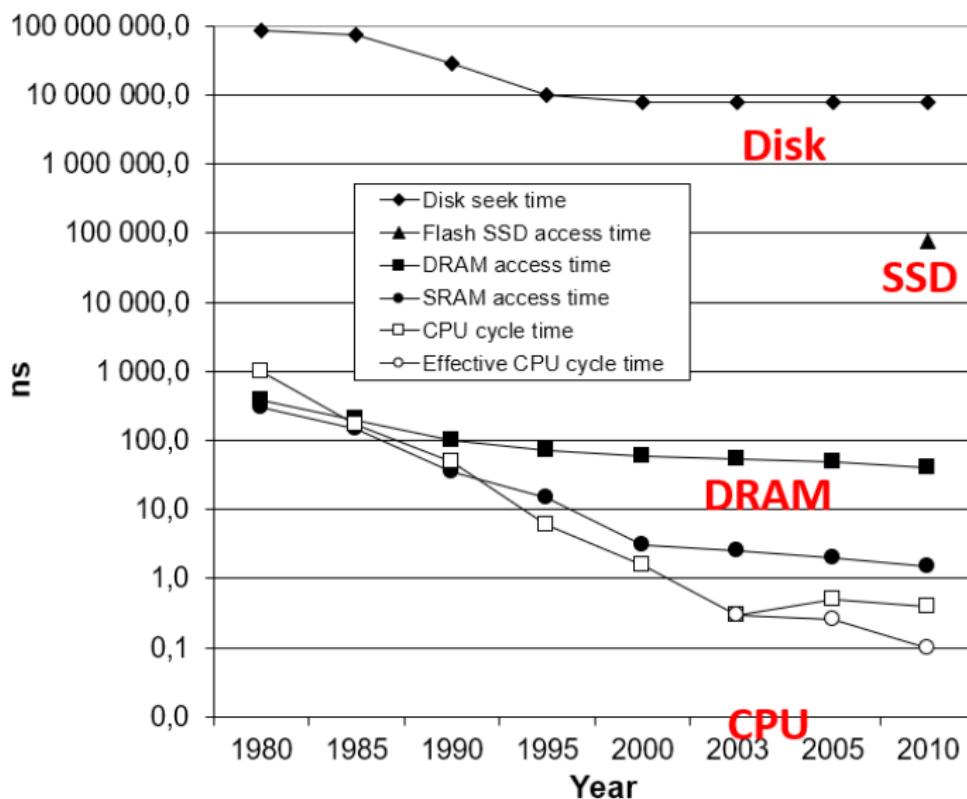
Size (bytes): 100 64K K..M M G T

- Взято тут: <https://slideplayer.com/slide/3641496/>.

Різновиди "електронної"/внутрішньої пам'яті



Швидкодія різних видів пам'яті



- Взято тут: <https://slideplayer.com/slide/13863089/>.

Кешування/буферизація

- Ввід-вивід значно повільніший, ніж робота із пам'яттю. Остання — значно повільніша, ніж робота процесора.
- Однак, зазвичай, звертання до даних володіють просторовою чи часовою локальністю.
- Часова локальність — дані, які були потрібні щойно, ймовірно, будуть потрібні і в найближчому майбутньому. (Приклад — лічильник чи акумулятор циклу).
- Просторова локальність: якщо щойно було використано якусь порцію даних, ймовірно, в найближчому майбутньому буде використано сусідню їх порцію. (Приклад — обхід масиву в циклі, послідовне читання файлу).
- Тому в комп'ютерній техніці широко використовується **кешування** — збереження щойно використаних, чи — близьких до щойно використаних (prefetching), даних у більш швидкій пам'яті.
- Кешування дозволяє зекономити на більш дорогих, однак, швидших, носіях, або й реалізувати основні об'єми пам'яті, недоступні для швидшої пам'яті, на більш повільній технології, майже без втрати продуктивності.

Види кешів

- Кеш пам'яті (іноді його називають кеш процесора) — "кешує" доступ до повільної основної пам'яті. Часто – багаторівневий.
- Translation lookaside buffer — TLB. Трансляція сторінок у типових реалізаціях віртуальної пам'яті вимагає звертання до таблиць у пам'яті. Якщо б таке звертання відбувалося фізично кожного разу – швидкодія була б катастрофічно низькою.
- Кеш диску – кешує в оперативній пам'яті¹ результати звертання до диску чи дані для запису. На відміну від кешу пам'яті та TLB, зазвичай керується програмно, ОС².
- Web-кеш – кешування на диску результатів, отриманих із мережі.
- Memoization – оптимізація, що полягає у збереженні результатів обчислень для повторного використання³.
- В якомусь сенсі, віртуальну пам'ять теж можна вважати кешем диску, а сам диск – кешем Всесвітньої мережі, Інтернету.

¹Тій чи іншій, включаючи власну пам'ять контролера диску.

²В деяких архітектурах, TLB чи й кешем теж повинна керувати ОС.

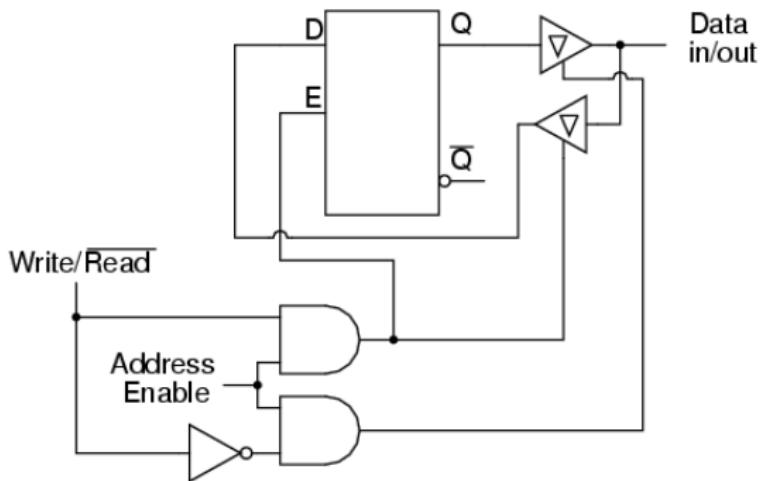
³Числа Фіbonacci на швидкість комусь доводилося розраховувати? З іншого боку, під час паралельних обчислень часто швидше просто розрахувати якусь величину ще раз, ніж звертатися до неї у пам'яті.

Cache Type	What is Cached?	Where is it Cached?	Latency (cycles)	Managed By
Registers	4-8 bytes words	CPU core	0	Compiler
TLB	Address translations	On-Chip TLB	0	Hardware
L1 cache	64-bytes block	On-Chip L1	1	Hardware
L2 cache	64-bytes block	On/Off-Chip L2	10	Hardware
Virtual Memory	4-KB page	Main memory	100	Hardware + OS
Buffer cache	Parts of files	Main memory	100	OS
Disk cache	Disk sectors	Disk controller	100,000	Disk firmware
Network buffer cache	Parts of files	Local disk	10,000,000	AFS/NFS client
Browser cache	Web pages	Local disk	10,000,000	Web browser
Web cache	Web pages	Remote server disks	1,000,000,000	Web proxy server

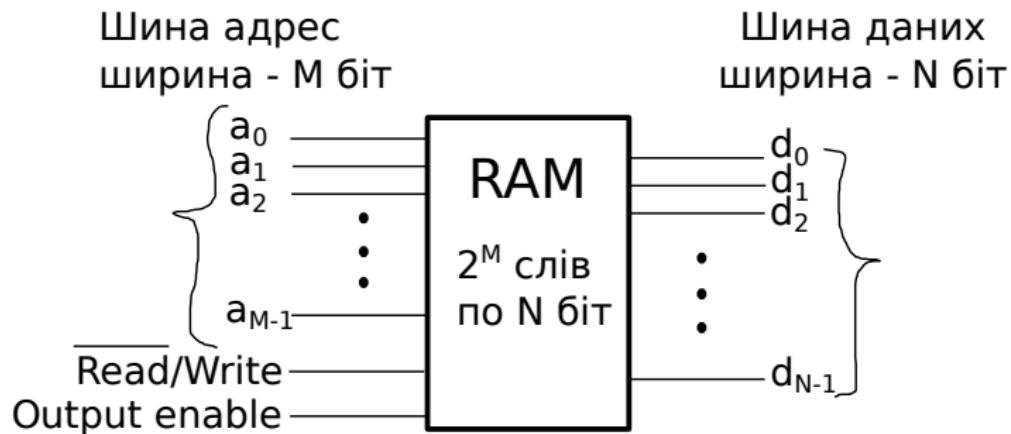
- Взято тут: <https://slideplayer.com/slide/13863089/>.

Однобітна комірка із спільним I/O

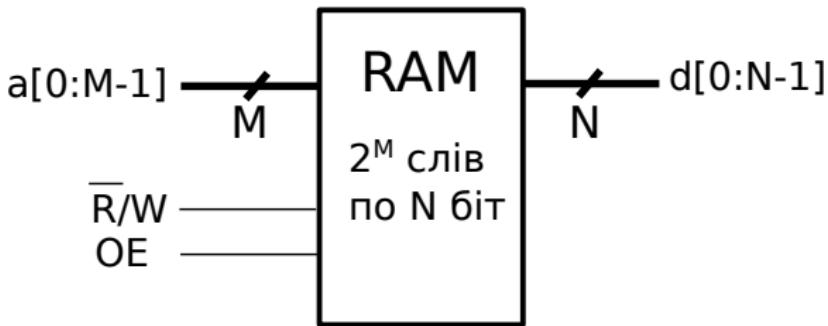
Memory cell circuit



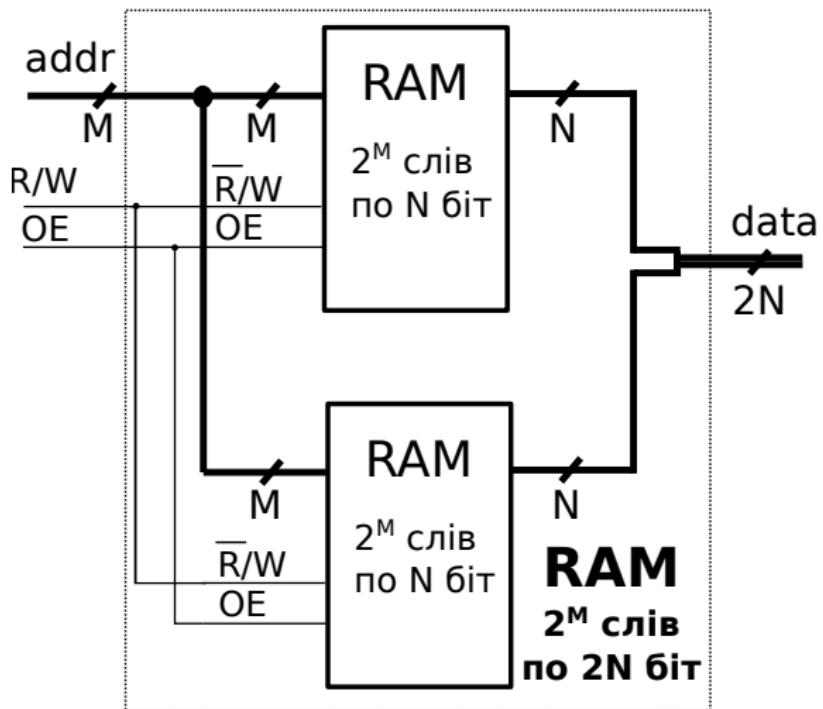
Інтерфейс модуля RAM



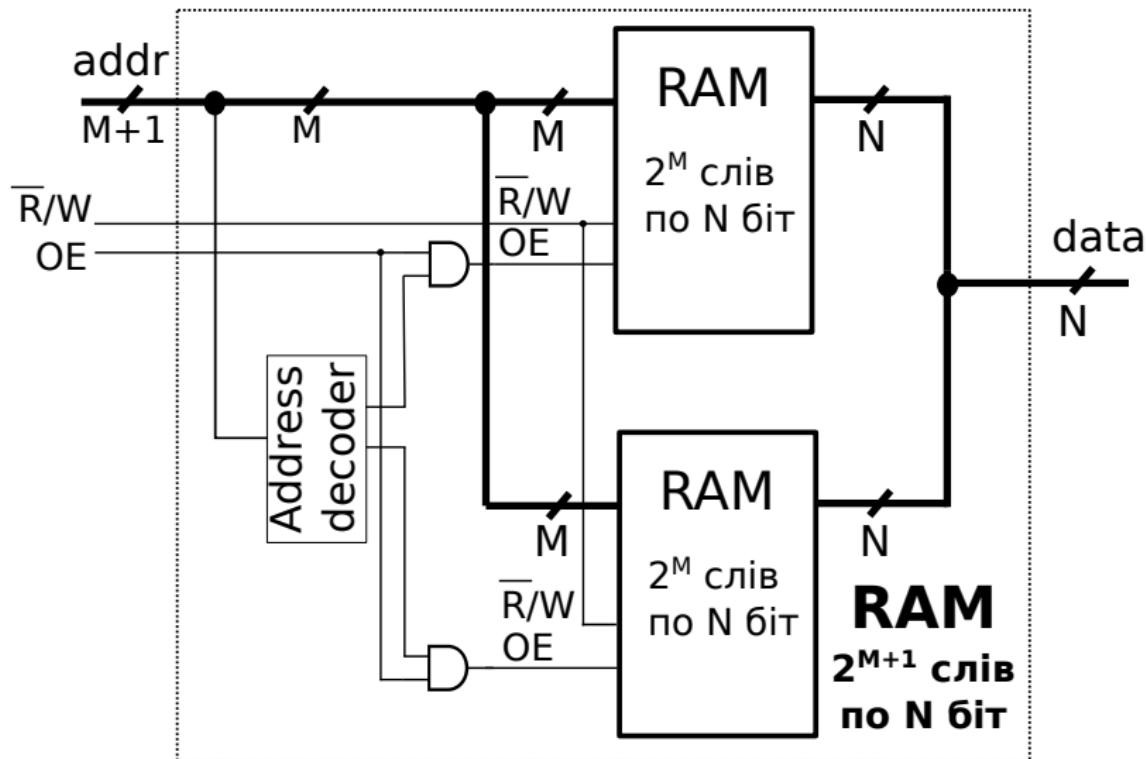
- Або в більш "дорослому" зображенні:



Розширяємо шину даних



Розширяємо шину адрес



Трансляція і MMU

- За відсутності MMU⁴ **логічна адреса**, використана машинною командою безпосередньо передається на адресні лінії пам'яті (іншими словами, на шину адрес, яка є частиною шини пам'яті).
- MMU виконує **трансляцію** – адреса, перш ніж потрапити на шину адреси, перетворюється згідно певних правил.
- Організація MMU може базуватися на сегментах і/або сторінках.
- Зараз домінує сторінкова організація пам'яті⁵.
- Для трансляції за сторінкової організації використовується спеціальна структура даних, **таблиця сторінок – page table**⁶.

⁴Memory management unit.

⁵Як не дивно, сторінкова пам'ять з'явилася раніше, в 1959/1962 (Atlas Computer), а сегментна – 1961, B5000. Джерело: https://en.wikipedia.org/wiki/Virtual_memory#History.

⁶Іноді також застосовують inverted page table. Приклад: PowerPC, UltraSPARC та IA-64.
https://www.usenix.org/legacy/publications/library/proceedings/osdi99/full_papers/dougan/dougan_html/node3.html.

Віртуальна пам'ять і MMU

- Надалі говоритимемо про сторінкову організацію пам'яті.
- За допомогою MMU можна реалізувати **віртуальну пам'ять**.
- Зараз домінує flat memory model чи linear memory model – пам'ять виглядає для програм неперервним блоком.
- Якщо MMU підтримує **відсутні**⁷ сторінки чи сегменти, за його допомогою можна також реалізувати підкачування сторінок – **swapping** чи **paging**.
- Зараз популярно кожному процесу виділяти свій, незалежний адресний простір. Однак, все ще існують системи, де всі процеси "живуть" в єдиному адресному просторі – IBM i (https://en.wikipedia.org/wiki/IBM_i).
- Напів-жартівлива гра, запропонована в 1972-74:
https://en.wikisource.org/wiki/The_Paging_Game.

⁷ В RAM.

Page fault

- Коли відбувається звертання до відсутньої в RAM сторінки, генерується переривання page fault – помилка відсутності сторінки. Тоді менеджер пам'яті ОС може знайти відповідну сторінку⁸, завантажити її в RAM і повторно виконати команду, що спричинила page fault – тепер звертання буде успішним.
- В результаті роботи цього механізму, ті ж логічні адреси процесу можуть відповідати різним фізичним у різні моменти часу.
- Частина сторінок можуть бути зафіксованими – якщо їх не можна переміщати й у фізичній пам'яті. Наприклад, коли з ними працює ще якась периферія, типу DMA.
- Також, може бути заборонено витісняти (swap) певні сторінки із RAM⁹. Наприклад, обробник page fault завжди має бути присутнім, та й взагалі, якщо обробники переривань дозволити витісняти – продуктивність дуже страждатиме. Це ж стосується й інших критичних частин ОС.

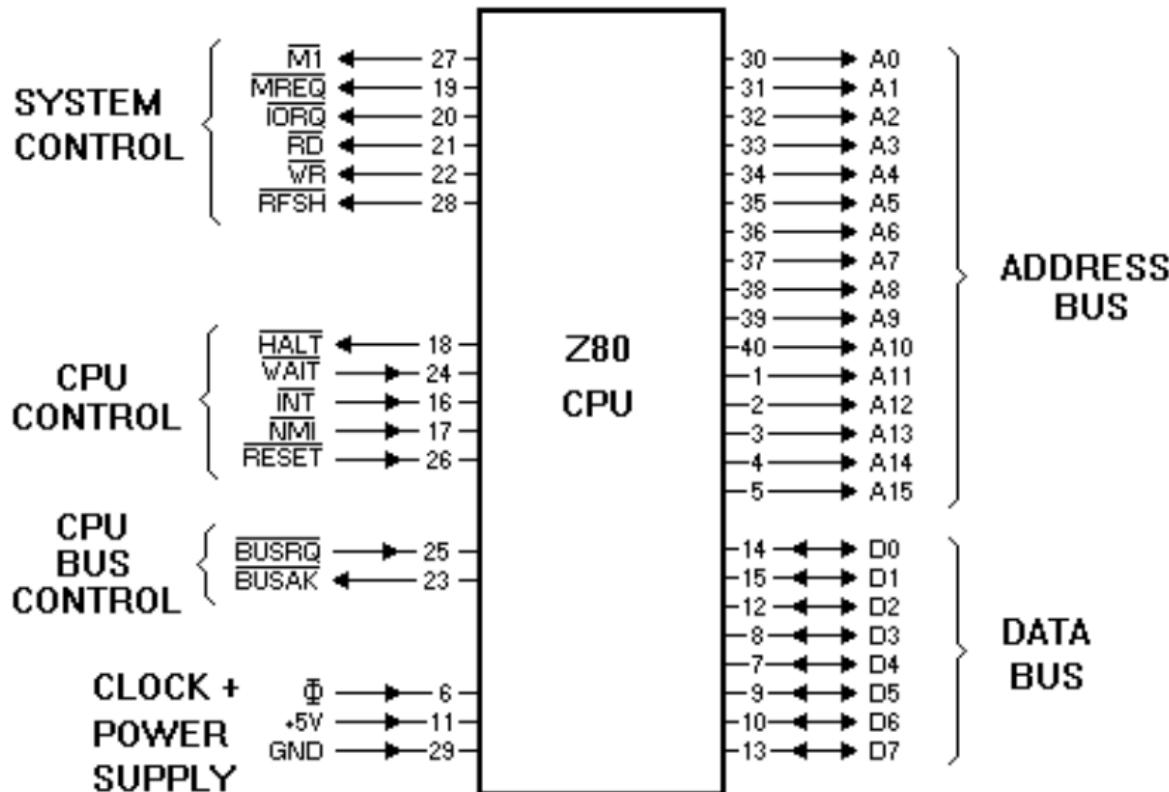
⁸ В припущеннях, що таке звертання коректне – процес має право звертатися до цієї сторінки. Де знайти? Якщо вона має бути порожньою – просто виділити програмі ще одну. Якщо її раніше було збережено на диск – прочитати з диску. Якщо у викинутій раніше сторінці був код – підвантажити його з виконавчого файлу, і т.д.

⁹ Їх переміщення може залишатися дозволеним.

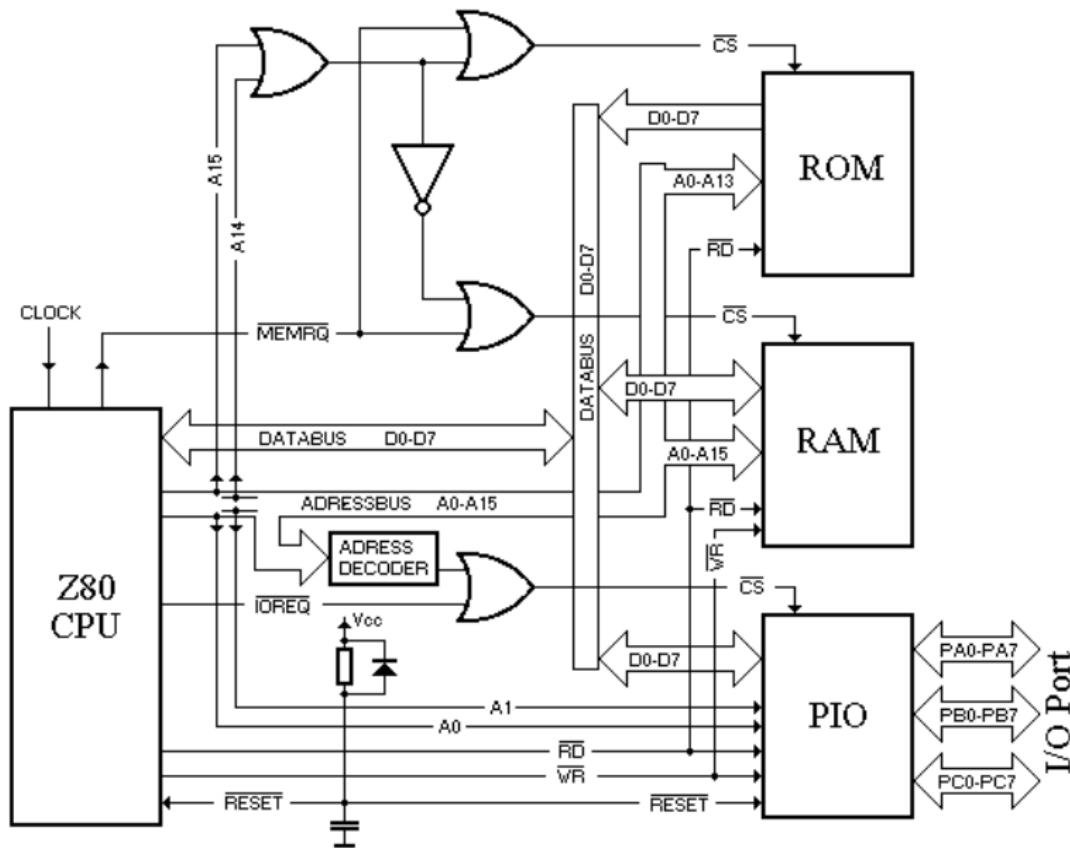
MPU

- MPU – memory protection unit.
- MPU – урізана версія MMU, не забезпечує трансляції, лише захист областей пам'яті.
- Наприклад, деякі ARM Cortex M мають MPU, але жодне не має MMU. Cortex R мають MPU, але не MMU. Cortex A – обладнані MMU.

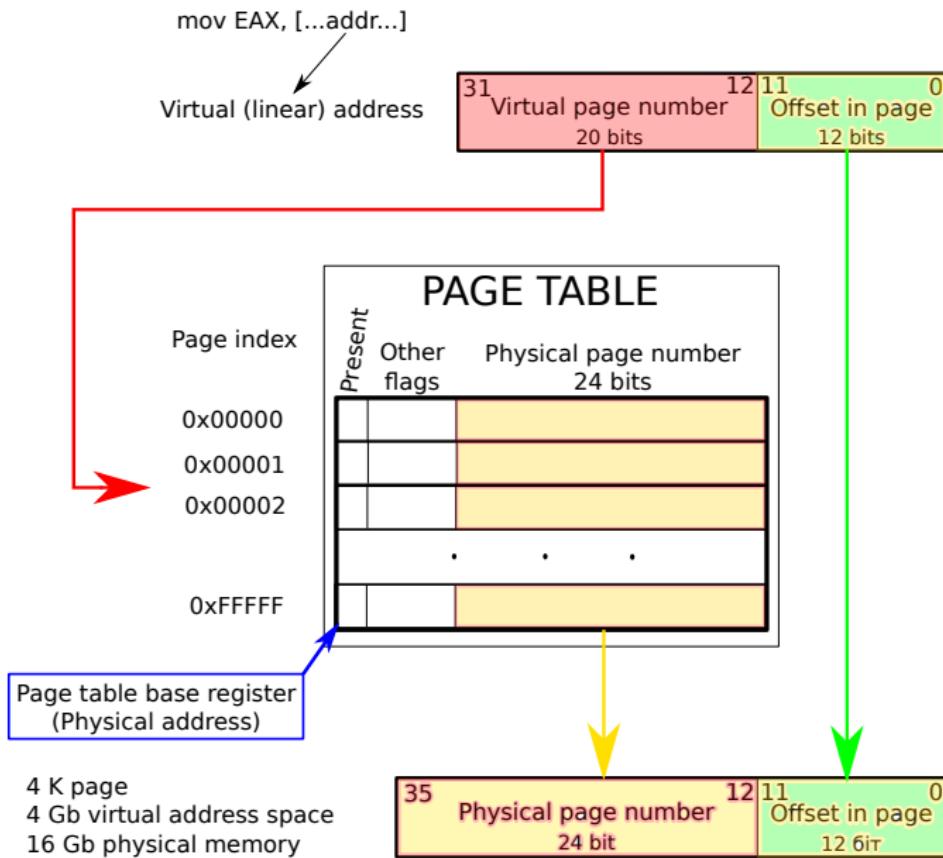
Z80 – приклад процесора без MMU



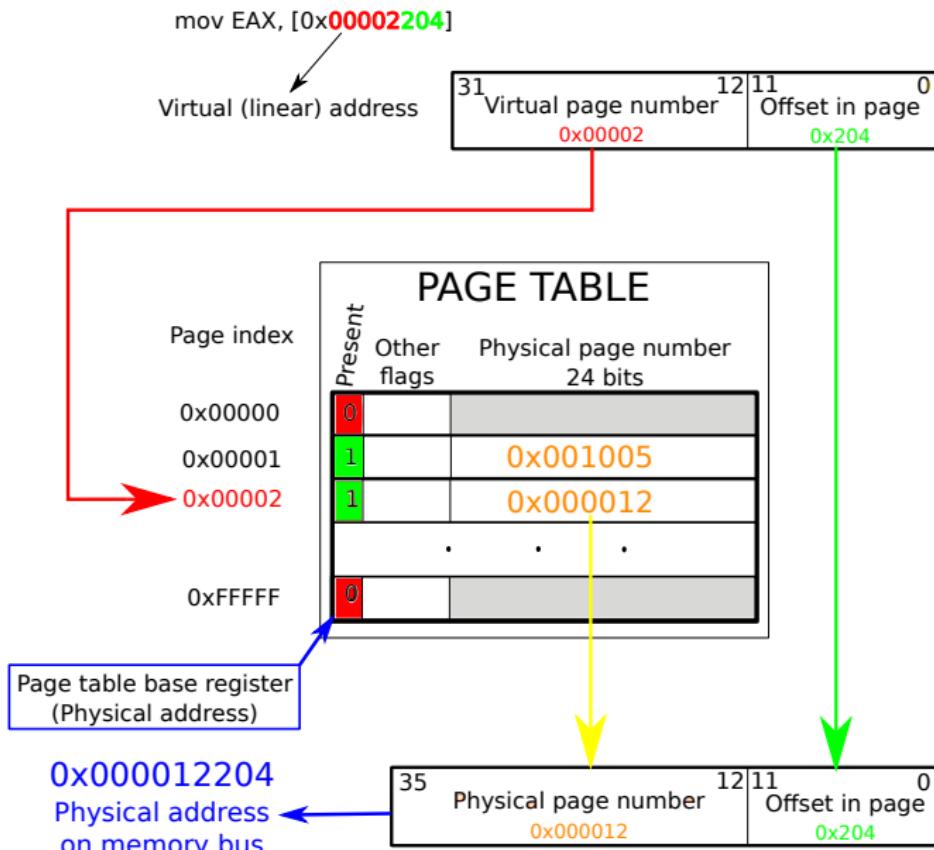
Комп'ютер на базі Z80 – без MMU



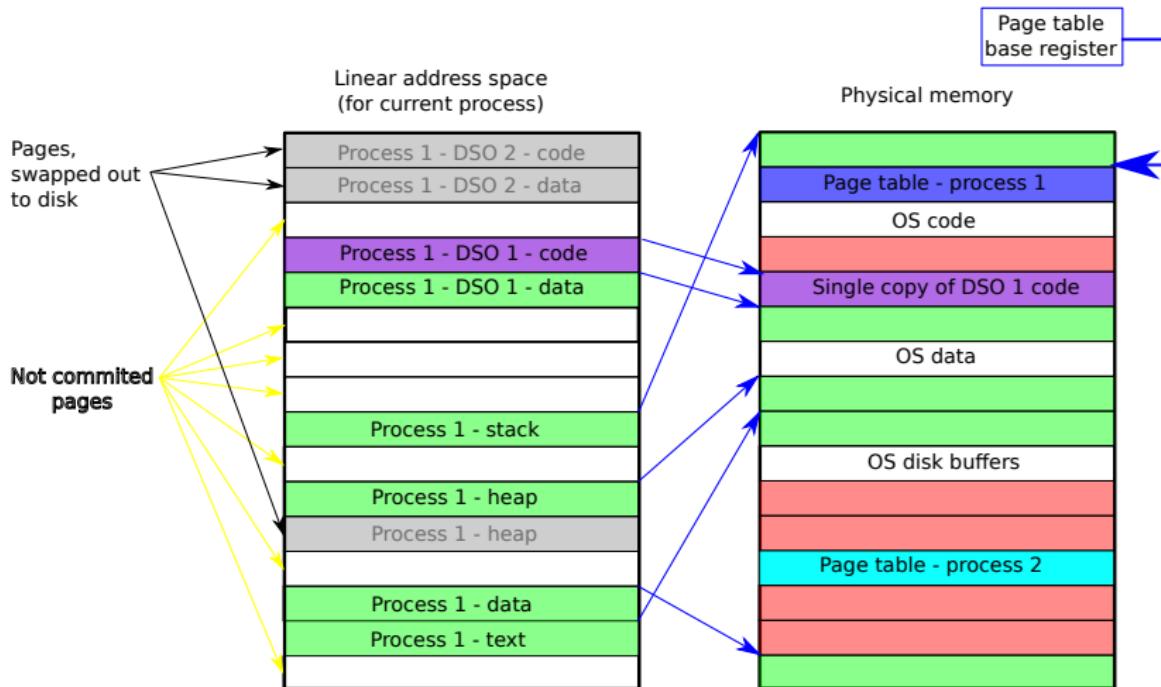
Приклад трансляції



Приклад трансляції

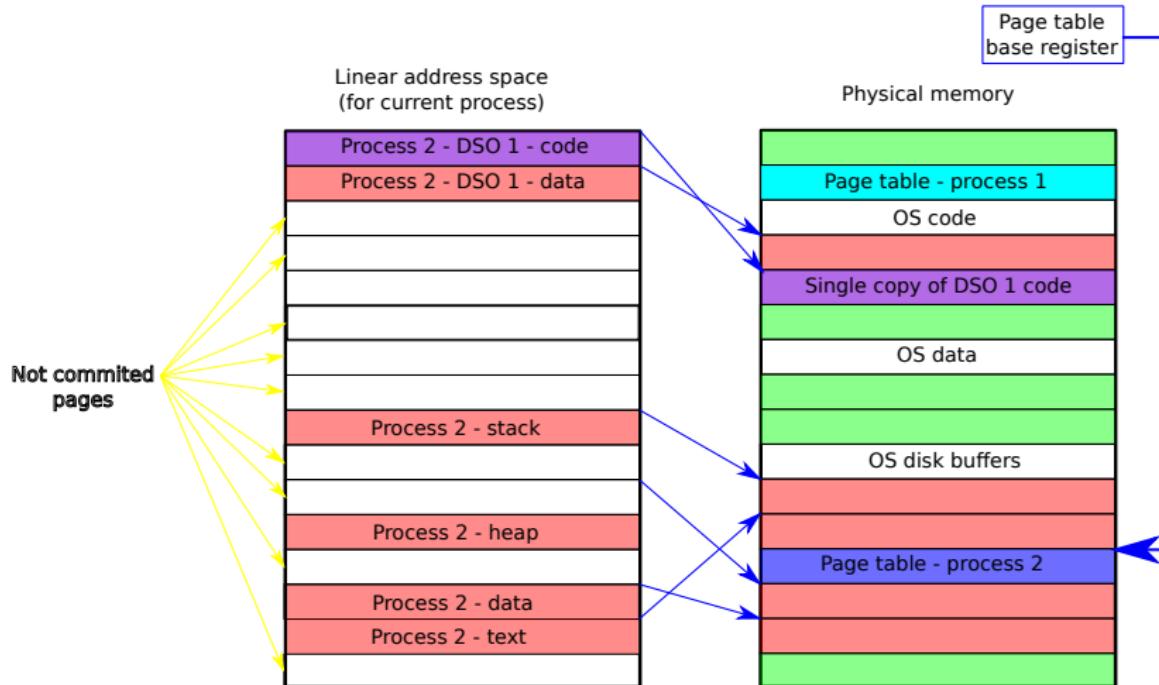


Організація віртуальної пам'яті процесу



- Запущені разом програми потребують більше фізичної пам'яті, ніж є. Однак, не вся вона потрібна одночасно. І, навіть якби її вистачило, раціональніше використати частину її під буфери – замість тримати в RAM рідко потрібні сторінки програм.

Зміна віртуальної пам'яті при переключенні на інший процес



- Зверніть увагу – код DSO (динамічної бібліотеки – dynamic shared object) є лише в одній копії. Але дані в кожного процесу свої.

Приклад – x86

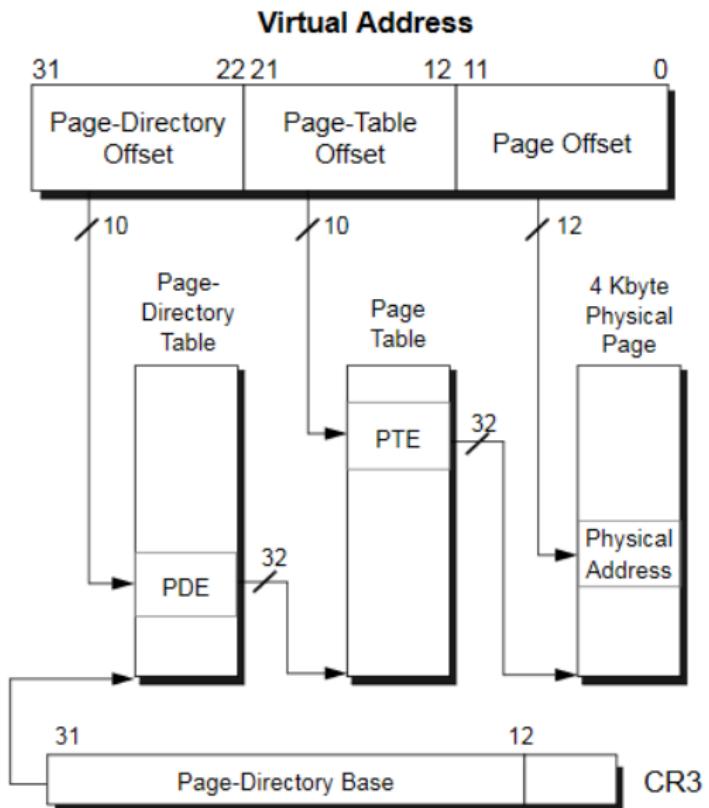
- Базовий розмір сторінок x86 – 4 Кб. В різних режимах підтримуються також 2 Мб, 4 Мб і 1 Гб сторінки.
- Однорівнева таблиця сторінок – надміру громіздка.
 - ◊ Для x86-32, елемент таблиці сторінок має 32 біти, 4 байти, для x86-64 – 64 біти.
 - ◊ Для відображення 4 Гб потрібно 2^{20} записів, або 4 Мб. Для кожного процесу.
 - ◊ Системи часу появи x86-32 зазвичай мали менше пам'яті всього.
 - ◊ Та й зараз це достатньо марнотратно – 4 Мб фіксованої пам'яті на процес.
- Тому Intel використала багаторівневу таблицю:
 - ◊ Регістр таблиці сторінок (в x86 він називається **CR3**) вказує на Page Directory Table – таблицю таблиць сторінок. Старша частина адреси служить індексом в цій таблиці.
 - ◊ Вона вказує на таблицю сторінок (одну із багатьох), середня частина адреси служить індексом у ній, а відповідний запис вже містить фізичну адресу сторінки.

- Збільшення фізичної пам'яті за межі 4 ГБ (так-зване РАЕ розширення), збільшення віртуального адресного простору (x86-64), підтримка віртуалізації¹⁰, додавали нові рівні таблиць.
 - ◊ Кількість таких рівнів таблиць в сучасних x86, зазвичай, 4.
 - ◊ Інтел в Ice Lake (Sunny Cove) реалізував 5 рівнів¹¹. Здається, це абсолютний рекорд, і можна адресувати до 128 Петабайт ($128 \cdot 2^{50}$, хоча це все ще менше за 16 екзабайт – $16 \cdot 2^{60} = 2^{64}$, доступних в 64-бітному адресному просторі).

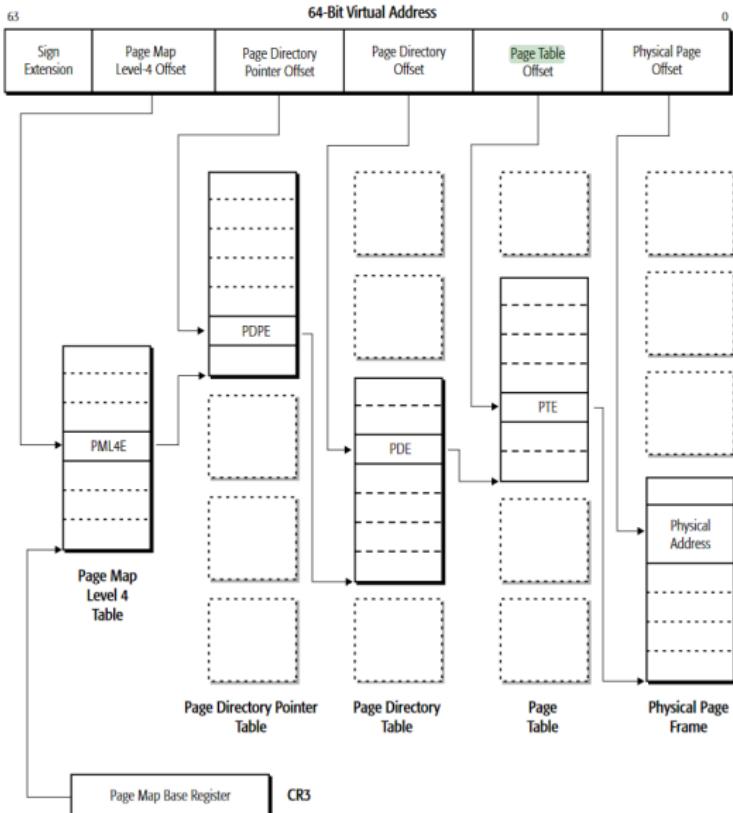
¹⁰ Див. також https://en.wikipedia.org/wiki/Second_Level_Address_Translation.

¹¹ https://en.wikipedia.org/wiki/Intel_5-level_paging.

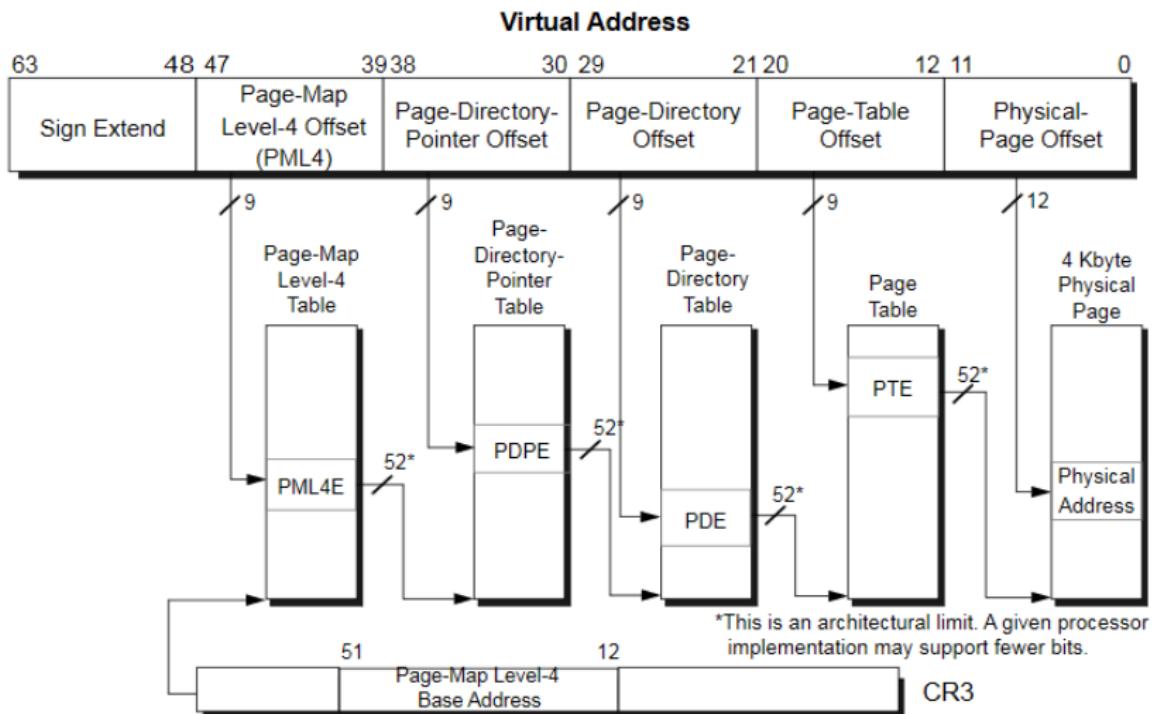
- Найпростіший варіант – x86-32, 4Кб сторінки, без РАЕ. Записи – 32 біти.



- x86-64, 4Кб сторінки, записи – 64 біти.



- x86-64, 4Кб сторінки, детальніше



Приклад – формат запису таблиці сторінок

63 62		52 51		32															
N	X	Available		Physical-Page Base Address (This is an architectural limit. A given implementation may support fewer bits.)															
31		12 11 9 8 7 6 5 4 3 2 1 0																	
Physical-Page Base Address		AVL G A D A P C W / U R / P T D T S W																	

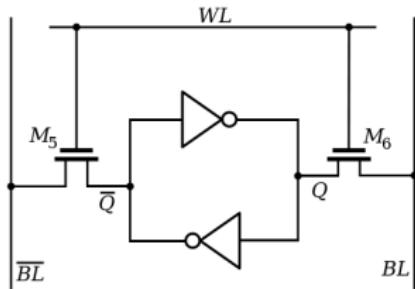
- Біт 0, P (Present) — вказує чи сторінка присутня в пам'яті. Якщо ні, P=0 — звертання до неї викличе #PF (page fault). Керує ним OS.
Якщо P=0, всі решта біти запису OS може використовувати на свій розсуд. Наприклад, зберігати інформацію про те, куди ж було збережено сторінку.
- Біт 1, R/W (Read/Write) — якщо дорівнює нулю, сторінка доступна тільки для читання, інакше також для запису.
- Біт 2, U/S (Supervisor/User) — рівень привілей для сторінки.
- Біт 3, PWT (Page-level write-through) — керує кешуванням сторінок. 0 — write-through, 1 — write-back.
- Біт 4, PCD (Page-level cache disabled) — забороняє кешувати дану сторінку.

- Біт 5, A (Accessed) — встановлюється в 1, якщо до даної сторінки звернулися. Очищається програмно. Може використовуватися OS, щоб визначати частоту використання сторінок і, наприклад, записувати на диск ті, що використовуються рідко.
- Біт 6, D (Dirty) — встановлюється, якщо в цю сторінку здійснено запис. Може використовуватися для керування спільним використанням сторінок по типу copy-on-write.. Очищається програмно.
- PAT (Page attribute table index) стосується кешування сторінок, подробиці надміру громіздкі, щоб їх тут наводити.
- Біт 8, G — 1 вказує що сторінка глобальна. Помічені цим бітом сторінки не очищатимуться в TLB при перезавантаженні регістра CR3. Це дозволяє не інвалідувати записи, для яких трансляція адрес не змінюється при переключенні задач, наприклад сторінки OS, відображені в пам'ять кожного процесу.
- Біти AVL/Available — можуть бути використані OS на свій розсуд, наприклад для збереження інформації про частоту використання сторінок або їх походження.
- Біт 63, NX – No Execute. Забороняє виконання коду для цієї сторінки.

- Огляд вище – дуже поверхневий.
- Детальніше див. офіційну документацію:
 - ◊ <https://www.amd.com/system/files/TechDocs/24593.pdf>, "AMD64 Architecture Programmer's Manual Volume 2: System Programming" від AMD,
 - ◊ <https://software.intel.com/en-us/articles/intel-sdm>, Volume 1 та Volume 3.
- Підручник по використанню, в режимі x86-32:
<http://indrekis2.blogspot.com/2011/08/x86.html>, зокрема урок 8.

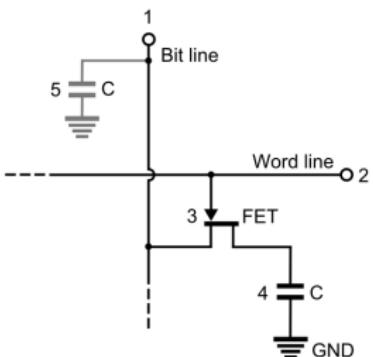
Види оперативної (первинної) пам'яті — Random Access Memory (RAM)

- SRAM – статична напівпровідникова пам'ять.
- Одна із реалізацій – на тригерах. Такий варіант ми розглядали на курсі ПОК.
- Однак, на практиці, використовуються спеціалізовані схеми. Еталонна комірка – 6 транзисторів, існують інші варіанти.
- Швидка, дорога, не потребує оновлення, енергоекономна.
- При вимкненні живлення – втрачає вміст.



DRAM

- DRAM – динамічна напівпровідникова пам'ять.
- Базується на конденсаторах, потребує значно менше транзисторів, ніж SRAM, відповідно – має більшу густину.
- Повільніша, дешевша за SRAM.
- Потребує регулярного оновлення вмісту¹².
- Менш енергоефективна, ніж SRAM.
- Як і SRAM, при вимкненні живлення – втрачає вміст.



- Симулляція: <http://www.falstad.com/circuit/e-dram.html>.

¹²Зараз оновленням займаються самі модулі пам'яті або чіпсет, раніше про оновлення мав турбуватися CPU.

Non-volatile random-access memory (NVRAM)

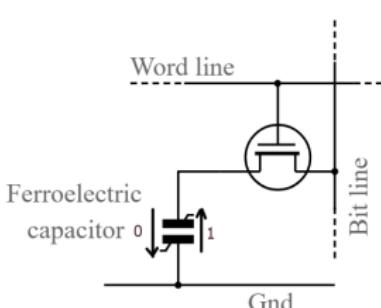
- NVRAM – збірна назва видів RAM, які не втрачають вміст при вимкненні живлення.
- EEPROM, та ж, що використовується як флеш-пам'ять¹³, буває, працює і в ролі RAM. Повільна, запис може бути на три порядки повільнішим за читання, обмежена кількість циклів запису¹⁴, потребує багато енергії для запису. Базується на транзисторах, які "залипають" у протилежних станах.
- FRAM – Ferroelectric RAM (сегнетоелектрична RAM). Швидка, час читання та запису – одинаковий, енергоекономна.
- Інші технології: Magnetoresistive RAM, Phase-change RAM, і т.д., поки мало представлені на комерційному ринку.
- Іноді також виділяють Pseudo SRAM (PSRAM) – DRAM, із вбудованими колами оновлення. Вона виглядає, як повільна SRAM, дозволяє обійтися без кіл та логіки взаємодії із та оновлення DRAM.

¹³ Остання, традиційно, дозволяє програмувати лише блоками – спрощуючи внутрішню будову.

¹⁴ Екзотичний варіант – SRAM, який зберігається на EEPROM при зникненні живлення, NOVRAM.

FRAM

- FRAM використовує сегнетоелектрики (ferroelectrics), спеціальні матеріали, що електрично поляризуються. Це явище, аналогічне до магнітної поляризації, яка використовується в магнітних носіях.
- Читання деструктивне, але не потребує живлення для збереження значень – сегнетоелектрик залишається поляризованим.
- В цілому – ідейно дуже схожа на RAM, тільки замість заряду конденсатора для запам'ятовування використовується його поляризація.
- Непоганий популярний огляд:
https://www.electronics-notes.com/articles/electronic_components/semiconductor-ic-memory/fram-ferroelectric-ram-operation-technology.php.



NVDIMM – 1

- NVDIMM – non-volatile dual in-line memory module¹⁵. Це NVRAM-модулі, сумісні¹⁶ із звичними DDRx-DRAM модулями, про які далі.
- Ідея – сумістити швидку DRAM із повільнішим пристроєм збереження, куди зберігається вміст DRAM при вимкненні. Для цього використовується енергія, збережена в самому модулі. Є розвитком модулів DRAM із акумуляторами – які достатньо проблемні.
- NVDIMM-F – (NAND) Flash-пам'ять, що парується із звичними DRAM DIMM.
- NVDIMM-N – (NAND) Flash+DRAM в одному модулі.
- NVDIMM-P – є розвитком NVDIMM-N, публікація стандарту запланована на 2020:

<https://software.intel.com/en-us/articles/enabling-persistent-memory-in-the-storage-performance-developer-guide>
разом із стандартом на DDR5 (див. далі). Включає безпосередній доступ до блочного non-volatile-пристрою, тобто, в якомусь сенсі є гібридом NVDIMM-F і NVDIMM-N.

¹⁵Що таке DIMM-модулі – говоритимемо в майбутньому. Якщо коротко – це стандарт на фізичну реалізацію модуля пам'яті.

¹⁶Чи, принаймні, схожими на них.

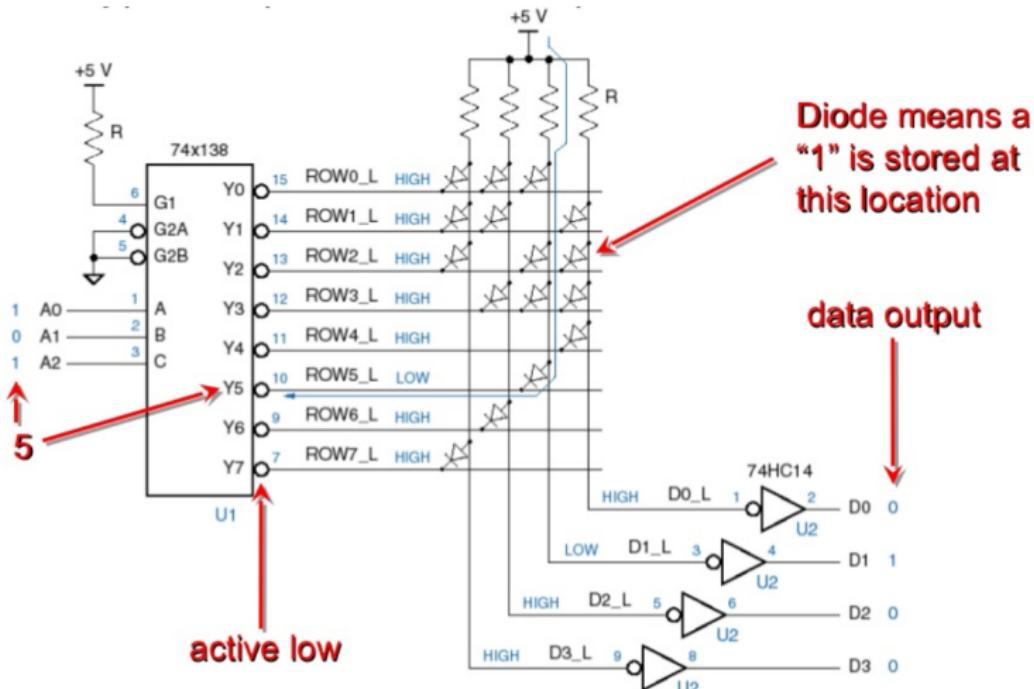
NVDIMM – 2

- Комерційні реалізації: нестандартний NVDIMM-X від Xitore¹⁷; Intel Optane (2015), Micron QuantX (2017), обов'язково за технологією 3D XPoint Phase-change memory. Дані стверджують <https://www.servethehome.com/intel-optane-hands-on-real-world-benchmark-and-test-results/> що серверна Optane – в 2.5 рази швидша за кращі SSD.
- Див. також <https://software.intel.com/en-us/articles/enabling-persistent-memory-in-the-storage-performance-developer-guide>, <https://habr.com/ru/company/raidx/blogs/350600/>, https://www.flashmemorysummit.com/English/Collaterals/Proceedings/2018/20180806_PreConC_Kulkarni.pdf та й <https://en.wikipedia.org/wiki/NVDIMM>.

¹⁷ Базується на ReRAM – Resistive random-access memory (мемристори).

Ідея ROM

- Варіанти ROM детальніше не розглядаємо – за їх різноманітністю.
- Ілюстрація принципу:



Екзотика – Content-addressable memory

- Вся пам'ять, яку ми розглядали, адресується номером комірки.
- Існує, хоча й не має широкого вжитку, інший підхід – асоціативна пам'ять, де адресація відбувається за допомогою вмісту. Звідси й назва: Content-addressable memory (CAM).
- Зараз використовується в маршрутизаторах, для швидкого пошуку в таблицях маршрутизації, кешах і т.д. – там де звичайна пам'ять не справляється.
- Крім звичної, бінарної, CAM, використовується також тернарна пам'ять, третій стан, зазвичай¹⁸, означає "don't care".
- Якщо її робити швидкою, складна та дорога (див. далі, розмову про асоціативність кешів – ефективна CAM має бути повноасоціативною).
- Існували комп'ютери, побудовані на базі такої пам'яті.
- Образно кажучи, CAM – така собі апаратна хеш-таблиця.
- Детальніше див. https://en.wikipedia.org/wiki/Content-addressable_memory і посилання там.

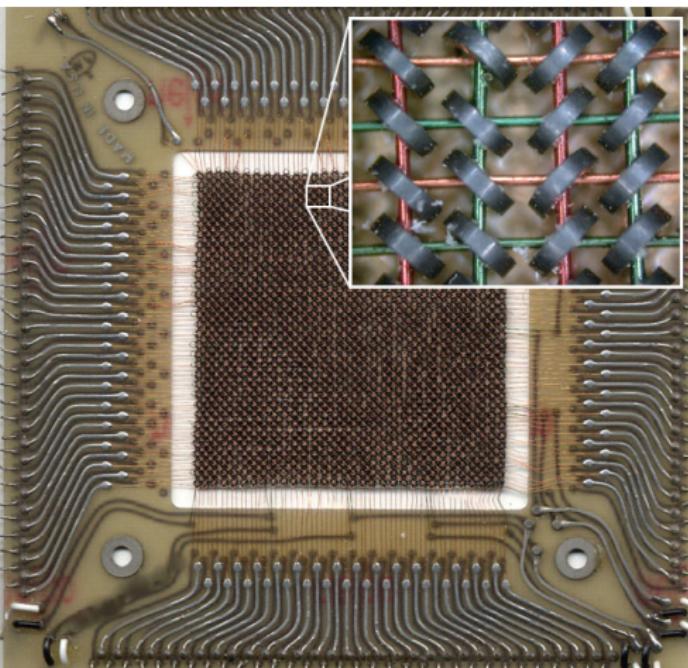
¹⁸Нагадаємо, що існує декілька тернарних узагальнень бінарної булевої алгебри.

Історія – magnetic core memory, пам'ять на магнітних осердях

- Підхід до RAM, який масово використовувався два десятиліття. Домінував, орієнтовно, в 1955-1975, потім був витіснений напівпровідниковою пам'яттю.
- Запам'ятує за допомогою намагнічування маленьких колечок із феромагнетика.
- Зберігає значення без живлення, але читання деструктивне.
- Термін "прошивання" походить із, власне, прошивання осердь провідниками – через кожен має пройти 3 або 4 провідника, лінії X, Y, S – sense і Z – inhibit, які об'єднані в новіших моделях.
- Ідея базується на гістерезисі. Як і в сучасних FRAM.

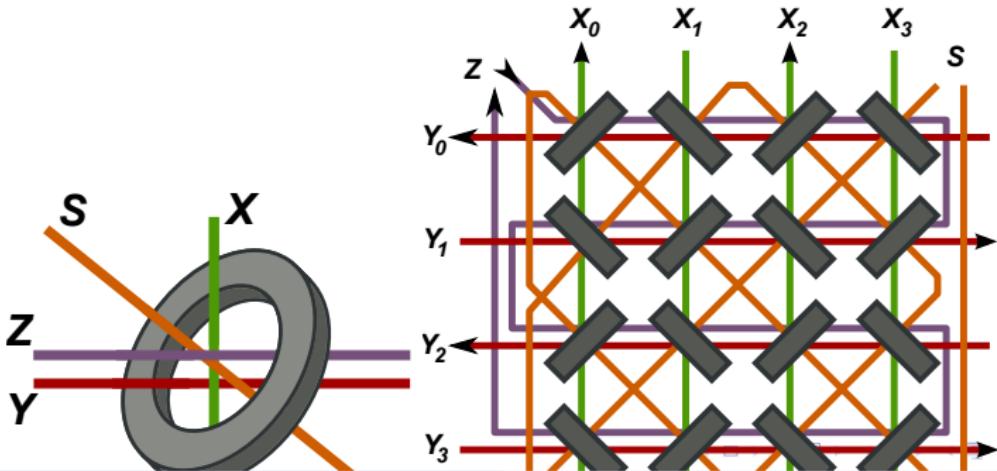
Історія – magnetic core memory, пам'ять на магнітних осердях

- 1024 біти, 1964, розмір – 10.8 см на 10.8 см, CDC 6600 – перший суперкомп'ютер¹⁹.



¹⁹3 MFLOPS, найшвидший комп'ютер в 1964-1969 роках.

- Щоб створити магнітне поле, достатнє для перемагнічування осердя, потрібно в провіднику в осерді створити струм певної сили. Менші значення не можуть змінити напрямок намагніченості.
- Якщо в лініях X та Y буде струм рівний половині необхідного значення, змінить свій стан лише осердя на їх перетині.
- Для читання в осердя записується 0 та вимірюється стрибок напруги на лінії S. Якщо він був – біт зберігав 1, інакше – 0.
- Для реалізації ROM на базі осердь, для бітів, рівних 1, S проходить через осердя, а для рівних 0 – повз нього. При тому, одне осердя може зберігати багато бітів.



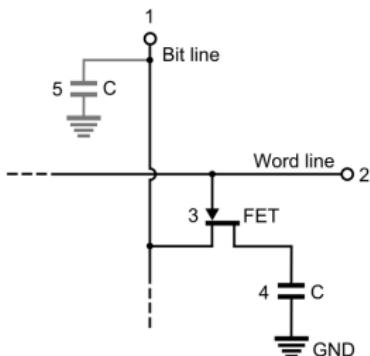
Посилання

- В блозі Ken Shirriff є багато цікавих статей про core memory і взагалі про древні комп'ютери.
- <https://www.righto.com/2015/08/examining-core-memory-module-inside.html> – IBM 1401 mainframe.
- <https://www.righto.com/2017/12/repairing-1960s-mainframe-fixing-ibm.html> – IBM 1401 mainframe.
- <https://www.righto.com/2015/10/repairing-50-year-old-mainframe-inside.html> – "Fixing the core memory in a vintage IBM 1401 mainframe".
- <https://www.righto.com/2020/03/the-core-memory-inside-saturn-v-rockets.html>
- <https://www.righto.com/2019/04/a-look-at-ibm-s360-core-memory-in-1960s.html> – "128 kilobytes weighed 610 pounds".
- <https://www.righto.com/2019/07/software-woven-into-wire-core-rope-and.html> – про ROM.
- <https://www.righto.com/2019/07/apollo-guidance-computer-dipsticks-and.html>
- <https://www.righto.com/2019/01/inside-apollo-guidance-computers-core.html>
- <https://www.righto.com/2020/03/inside-titan-missile-guidance-computer.html>

Посилання

- Деяка екзотика з цього ж блогу:
- <https://www.righto.com/2019/11/tros-how-ibm-mainframes-stored.html> – "TROS: How IBM mainframes stored microcode in transformers"
- <https://www.righto.com/2019/11/ibm-sonic-delay-lines-and-history-of.html> – "IBM, sonic delay lines, and the history of the 80?24 display"
- <https://www.righto.com/2014/12/inside-intel-1405-die-photos-of-shift.html> – "nside the Intel 1405: die photos of a shift register memory from 1970"

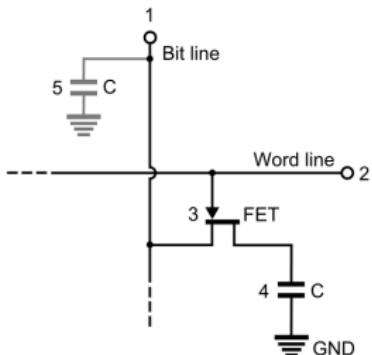
Подробиці роботи DRAM – 1 біт, читання



- Найпростіший варіант: Якщо подати на лінію 2 сигнал, транзистор відкриється.
- Якщо транзистор був зарядженим, лінією 1 протікатиме струм, інакше – не буде.
- Спеціальна схема (sense amplifier)²⁰ може виявити цей струм та представити його як звичні логічні рівні.
- При читання вміст комірки губиться – його потрібно записати назад. Займається цим теж sense amplifier.
- В реальних схемах все трішки складніше – використовується так-званий precharge, підтягування BL до напруги, по середині між рівнями 0 і 1. Тоді вимірюється напрямок струму.

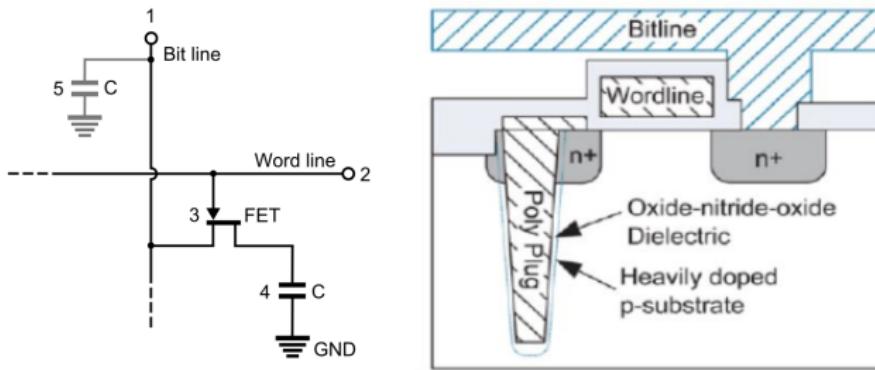
²⁰ Термін походить з епохи core memory.

Подробиці роботи DRAM – 1 біт, запис



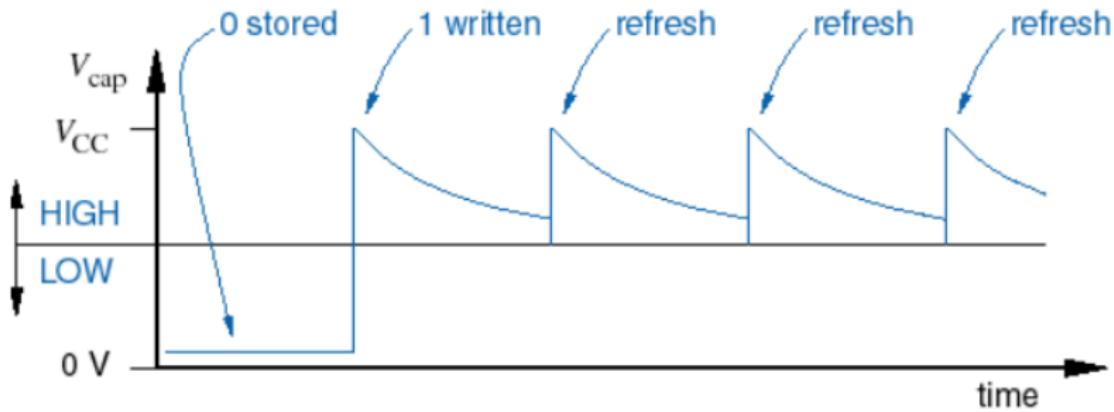
- Запис відбувається аналогічно – активується лінія 2, а на лінію 1 виставляється потрібне значення.

Подробиці роботи DRAM – 1 біт, фізична структура



Подробиці роботи DRAM – оновлення

- Ємність конденсаторів крихітна – щоб можна було їх багато вмістити. Згідно Urlich Drepper, 2007, заряджений конденсатор містить лічені десятки тисяч електронів. Тому, не зважаючи на величезний опір (тераоми), вони доволі швидко розряджаються.
- Сучасну пам'ять (DDRx SDRAM) слід оновлювати не рідше, ніж кожних 64 мс, при високих температурах – кожних 32 мс.
Подробиці див, наприклад: <http://utaharch.blogspot.com/2013/11/a-dram-refresh-tutorial.html>.

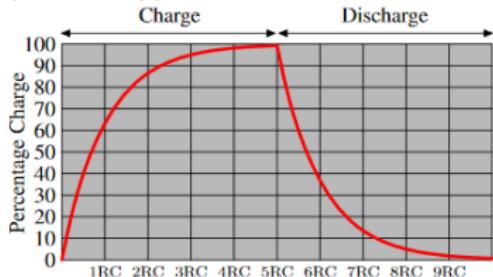


- Зображення із:

<https://www.slideshare.net/abhilash128/lec-21-16642228>

Подробиці роботи DRAM – оновлення

- При тому, як розряд, так і заряд конденсатора, потребує часу – це накладає обмеження на швидкодію пам'яті:



- RC – характерний час системи, тому його використано в ролі одиниці. Заряд конденсатора при зарядці:

$$Q(t) = Q_0 \left(1 - \exp\left(-\frac{t}{RC}\right)\right),$$

при розрядці:

$$Q(t) = Q_0 \exp\left(-\frac{t}{RC}\right),$$

де C – ємність конденсатора, R – опір кола, через яке він розряджається.

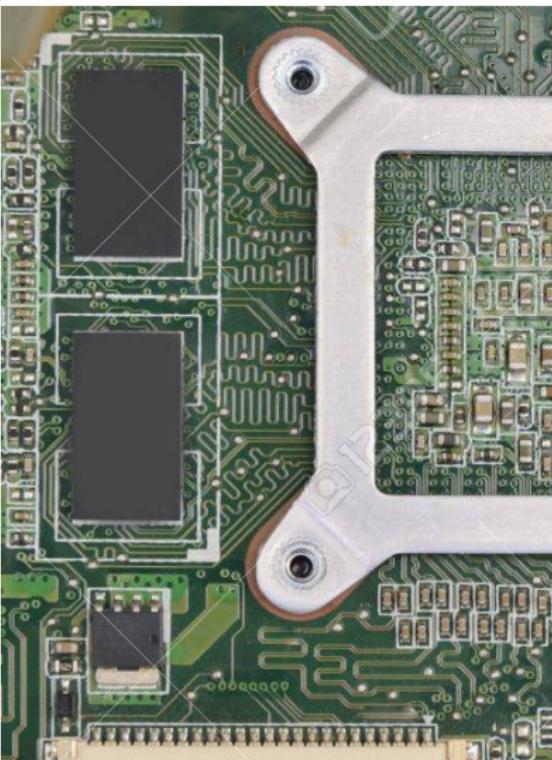
Подробиці роботи DRAM – 1 Gbyte

- За актуальних зараз об'ємів пам'яті, які навіть у користувачьких комп'ютерах доходять до перший десятків гігабайт, постає ще одна проблема: до кожного модуля слід підвести багато адресних пінів.
- Якщо ширина шини даних – байт, то для одного гігабайта слід мати декодер $30 \rightarrow 2^{30}$. А також, 2^{30} ліній – до кожного байта пам'яті.
- При чому, їх кількість росте експоненційно.
- Додаткова складність – сигнал по всіх 30 лініях повинен приходити (майже) одночасно. При тому, на поточних частотах, за один такт зміна електромагнітного поля, що рухається із швидкістю світла, встигає пройти не так багато²¹, різниця в довжині ліній стає дуже суттєвою проблемою – їх слід робити однаковими.
- Тому, на практиці, використовують окрему адресацію рядків та стовпців – пам'ять організована у вигляді матриці.

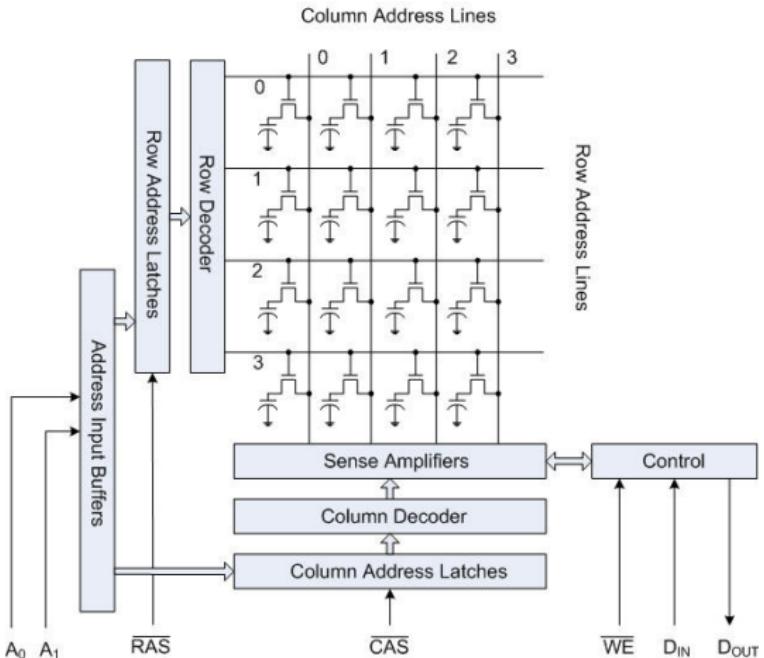
²¹ $c = 3 \cdot 10^8 \text{ m/s}$, $\nu = 2\text{Hz} = 2 \cdot 10^9 \text{ Hz}$, $T_{cycle} = \frac{1}{\nu} = 0.5 \cdot 10^{-9} = 0.5 \text{ ns}$. Тобто, за цей час світло встигає пролетіти $L = T \cdot c = 0.15 \text{ m} = 15 \text{ см}$. А швидкість світла в речовині ще нижча, хоча для високочастотного сигналу залежність складна.

Відступ

- Вирівнювання довжини доріжок на випадковій материнській платі з Інтернету:

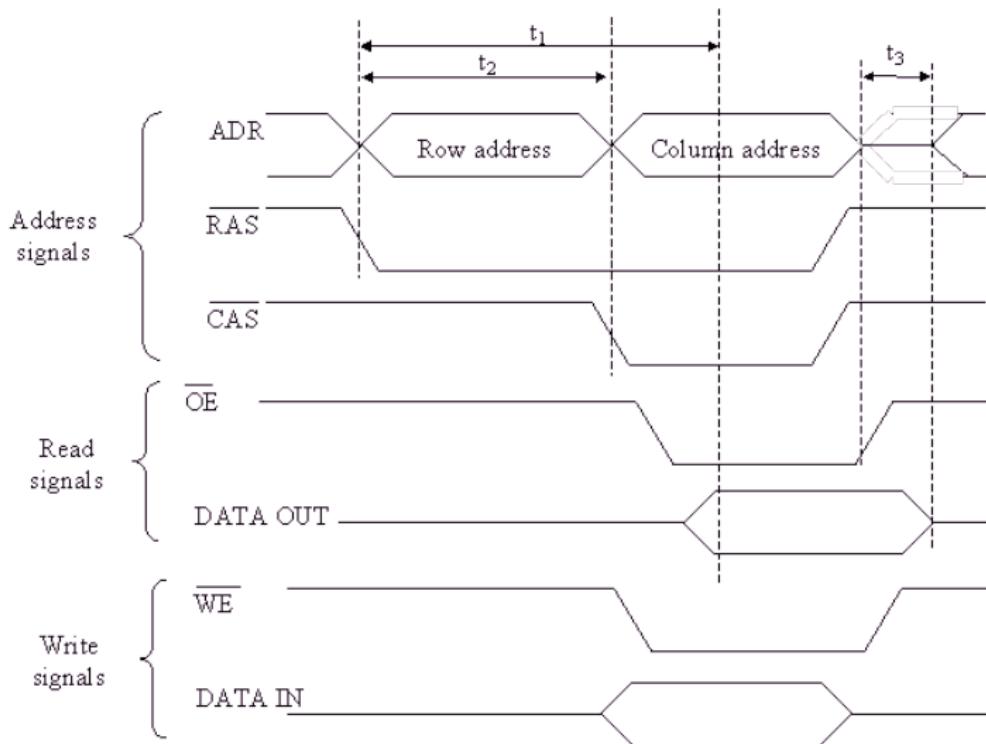


Зменшуємо кількість ліній



- На адресні піни подається адреса рядка.
- По сигналу /RAS, модуль пам'яті запам'ятує її.
- Потім на ті ж піни подається адреса стовпця, по /CAS – запам'ятується.
- Обмін відбувається після цього²².
- Решта контрольних сигналів – звичні.

²²Подробиці типу precharging та інші нюанси опускаємо заради простоти.

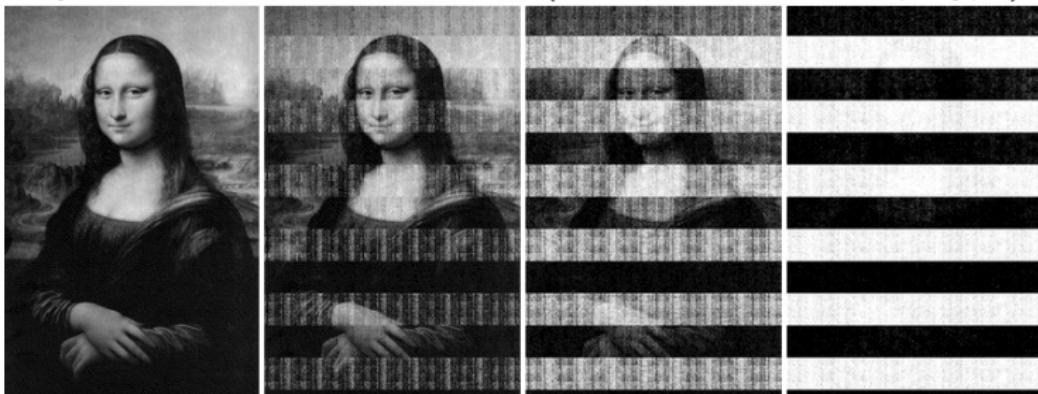


- Така пам'ять називається асинхронною DRAM.
- Зображення взято тут:
<https://edux.pjwstk.edu.pl/mat/264/lec/main93.html>.

Оновлення DRAM

- Оновлення, зазвичай, відбувається по рядках. Часто – один рядок кожних 64 мс, але деякі системи оновлюють всю пам'ять зразу.
- Працювати із рядком, що оновлюється, неможливо.
- ◊ У спеціалізованих різновидах пам'яті, де важлива реакція в реальному часі, оновлення може відбуватися у керовані зовнішнім таймером інтервали.
- ◊ Прикладом можуть бути відеопам'ять, яка може оновлюватися під час "vertical blanking interval" – часу між завершенням виводу останнього рядка пікселів і початком виводу першого. В епоху аналогових електронно-променевих дисплеїв це був час повернення електронного променя до початкової позиції.
- Подробиці тут не розглядатимемо. (Їх багацько).

- Якщо дуже потрібно – зловмисникам чи слідчим, вміст DRAM можна відновити протягом кількох хвилин після зникнення живлення – якщо їх сильно охолодити.
- ◊ Це так-звана "**Cold Boot Attack**", подробиці в оригінальній статті: https://www.usenix.org/legacy/event/sec08/tech/full_papers/halderman/halderman_html/. Без охолодження характерний час зникнення інформації складатиме секунди.
- Приклад із статті вище, зображення у пам'яті, через 5, 30, 60 секунд, 5 хвилин, без живлення (за нормальної температури):



- Охолоджена до -50°C , дозволяє зберегти вміст хвилинами, рідким азотом, -196°C – години, із кількістю збоїв меншою за 0.2%.

Асинхронна DRAM – 1

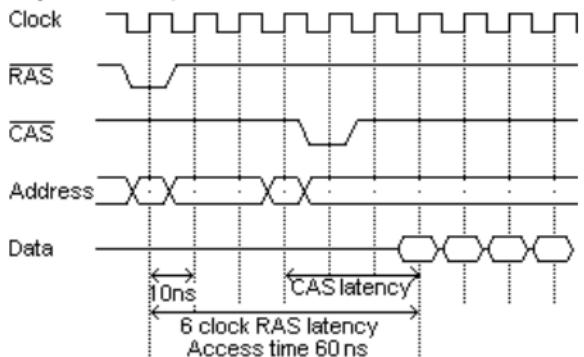
- Asynchronous DRAM – від 60-х і до середини 90-х. Тактуванням служили RAS/CAS сигнали.
- Page mode DRAM – варіація асинхронного DRAM, яка після вибору рядка (RAS), дозволяла читати кілька різних стовпців із цього рядка, повторним заданням адреси стовпця та імпульсом CAS.
- Розвиток цього підходу:
 - ◊ Fast page mode (FPM) – адресу стовпця можна подавати завчасно, до встановлення /CAS. Відповідні сигнали поширюються схемами, економлячи на цій затримці, але сама дія виконується по /CAS.
 - ◊ Static column – адреса стовпця веде себе асинхронно, не зберігається, а діє за принципом: змінилося значення на ній, за певний час – потрібний результат на виході.
 - ◊ Nibble mode – читання до 4-х послідовних адрес. Підхід схожий на використаний в DDR, але потребував окремих імпульсів на /CAS.

Асинхронна DRAM – 2

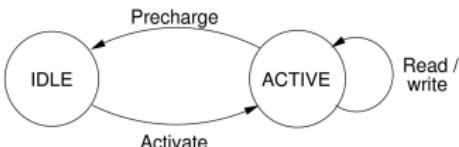
- Розвитком page mode RAM є Extended data out DRAM (EDO DRAM) – до FPM додала певне розпаралелення, попередні дані утримуються на шині даних, поки пам'ять розпочала цикл отримання адреси наступних. Отримане прискорення складало до 30%
- Burst EDO DRAM – пакетний режим, дозволяє опрацьовувати до 4 послідовних адрес.
- Список вище – далеко не вичерпний, наведено лише види пам'яті, що ілюструють використання важливих ідей щодо оптимізації.
- Деякі схеми асинхронного DRAM: https://www.csee.umbc.edu/courses/undergraduate/CMPE310/Fall07/cpate12/slides/html_versions/chap10_lect06_memory3.html.
- За подробицями див. також <https://edux.pjwstk.edu.pl/mat/264/lec/main93.html> та https://en.wikipedia.org/wiki/Dynamic_random-access_memory#Asynchronous_DRAM.

SDRAM

- **Synchronous dynamic RAM – SDRAM.** Практично вся сучасна DRAM є SDRAM.
- SDRAM додала сигнал тактування, а WE, RAS, CAS почали служити трьохбітовою командою модулю пам'яті:



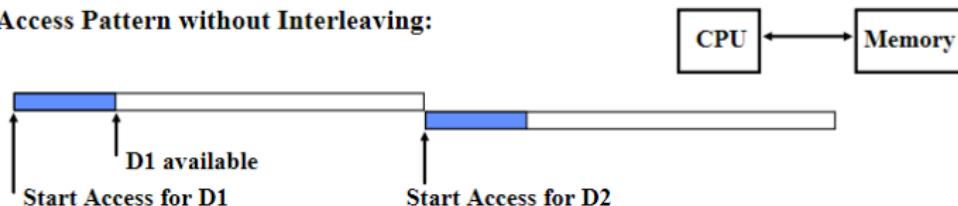
/CS	/RAS	/CAS	/WE	Address	Command
H	x	x	x	x	Command inhibit (No operation)
L	H	H	H	x	No operation
L	H	H	L	x	Burst Terminate: stop a read or write burst in progress
L	H	L	H		column: Read from currently active row
L	H	L	L		column: Write to currently active row
L	L	H	H		row: Activate a row for read and write
L	L	H	L	x	Precharge (deactivate) the current row
L	L	L	H	x	Auto refresh: Refresh one row of each bank, using an internal counter
L	L	L	L	mode	Load mode register: Address bus specifies DRAM operation mode.



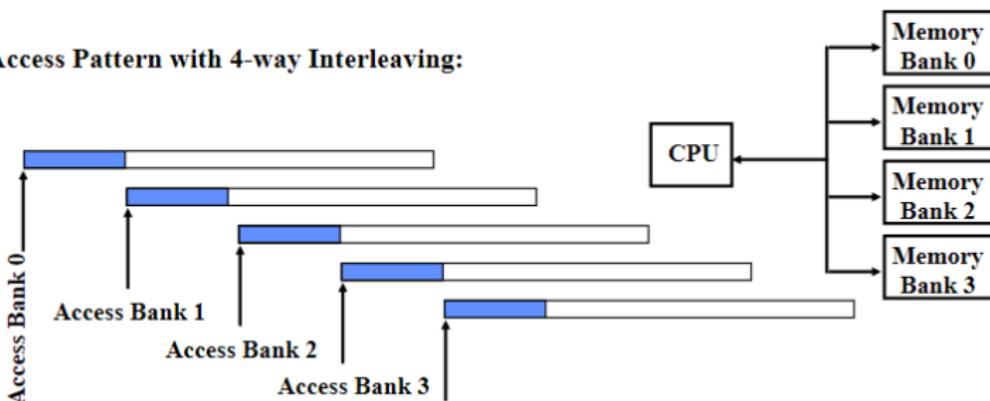
Банки SDRAM

- Підтримується pipelining – нові команди можна подавати ще до того, як завершилися попередні.
- Ще однією важливою новинкою SDRAM була підтримка банків: за допомогою трьох додаткових бітів команди (BA0-BA2), можна паралельно працювати із різними банками:

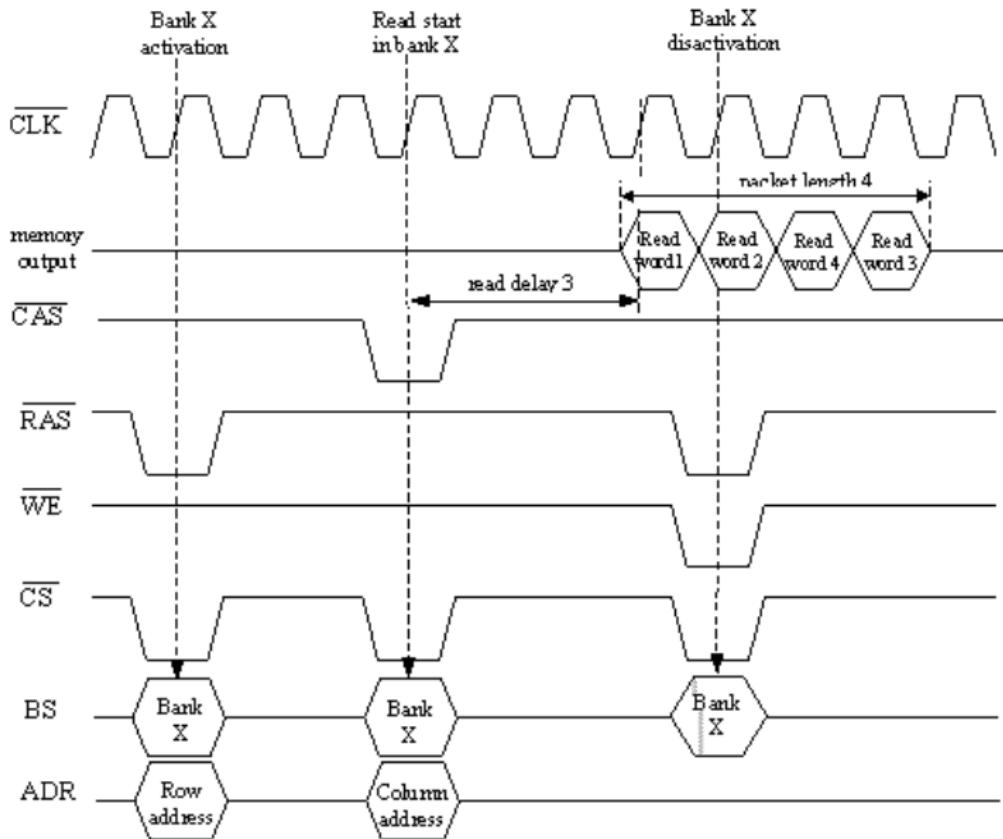
Access Pattern without Interleaving:



Access Pattern with 4-way Interleaving:

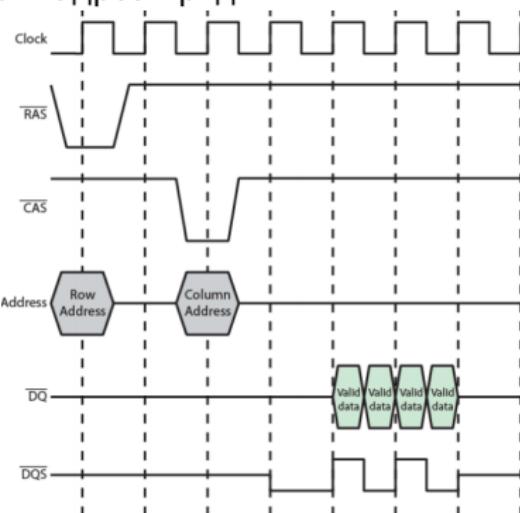


Часова діаграма SDRAM із банками



SDR SDRAM, DDR SDRAM, etc

- Першою в "сімействі" була Single Data Rate SDRAM – SDR SDRAM. Вона здатна отримувати одну команду та повернати один результат кожен такт.
- DDR SDRAM – має аналогічні команди, однак користується буфером передвибірки (prefetch buffer), вибираючи два сусідніх слова та повертуючи їх по фронту та спаду тактового сигналу.
- На відміну від старіших підходів, типу page mode DRAM, не потребує передачі адреси рядків.



DDRx SDRAM

- DDR2 (комерційно доступні з 2003 року) – 4 слова за раз.
- DDR3 (2007) – 8 слів за раз. (Іноді кажуть "буфер на 8 біт" – говорячи про одну лінію біт).
- DDR4 (2014, хоча анонсовані ще в 2008) – вищі частоти, нижчі напруги.
- DDR5 (планується до публікації в 2018 2019 2020 році).
- Зазвичай, ширина шини даних DDR модулів – 64 біта, 8 байт. В спеціалізованій апаратурі трапляються інші значення.
- Кожне із поколінь додавало й інші, більші чи менші, вдосконалення, що дозволяли підвищити частоту та загальну продуктивність, зменшити енергоспоживання (принаймні, в перерахунку на біт).
- В цілому, між версіями помітно зростала пропускна здатність (throughput), а латентність (час очікування) змінювався мало.
- Згадані вище DDRx види пам'яті є стандартами на SDRAM, які публікуються JEDEC, Solid State Technology Association. Між виробами різних виробників можуть бути розбіжності.
<https://en.wikipedia.org/wiki/JEDEC>

Посилання на стандарти та даташіти (draft)

- <http://cs.ecs.baylor.edu/~maurer/CSI5338/JEDEC79R2.pdf> – DDR SDRAM JEDEC Standard
- http://oz.nthu.edu.tw/~d947207/JEDEC/JESD79-2E_DDR2_SDRAM.pdf – DDR2 SDRAM JEDEC Standard
- <http://mermaja.act.uji.es/docencia/is37/data/DDR3.pdf> – DDR3 SDRAM JEDEC Standard
- <http://www.softnology.biz/pdf/JESD79-4B.pdf> – DDR4 SDRAM JEDEC Standard
-
- <http://wiki.laptop.org/images/4/43/A2S56D40CTP-G5.pdf> – 256Mb DDR SDRAM Specification 256Mb DDR SDRAM Specification (A2S56Dx0CTP).
- https://www.samsung.com/semiconductor/global.semi/file/resource/2017/11/ddr3_device_operation_timing_diagram_rev14-2.pdf – Samsung DDR3 SDRAM Specification: Device Operating & Timing Diagram.
- https://www.samsung.com/semiconductor/global.semi/file/resource/2017/11/ddr2_device_operation_timing_diagram_may_07-0.pdf – Samsung DDR2 SDRAM Specification: Device Operating & Timing Diagram.
-
- <https://compress.ru/article.aspx?id=16737> – "Энциклопедия современной памяти", непоганий огляд.
- <http://www.es.ele.tue.nl/premadona/files/akesson01.pdf> – "An introduction to SDRAM and memory controllers".

Маркування

- DDR1. Частота: 133, 166, 200 MHz. Позначається як DDR-266, DDR-333, DDR-400 – частота, помножена на два (двічі на такт здатна передавати дані). Відповідні DIMM-модулі (див. далі): PC-2100, PC-2700, PC-3200 (ще помножено на 8 – ширину шини даних в байтах). Ємність одного DIMM - до 1Гб (?).
- DDR2. Частота: 200, 266, 333, 400 MHz. DDR2-400, DDR2-533, DDR2-667 and DDR2-800. DIMM: від PC2-3200 до PC2-6400 – ще на 2 помножено. Ємність одного DIMM - до 8Гб (?).
- DDR3. Частота: 400, 533, 667, 800 MHz. DDR3-800, DDR3-1066, DDR3-1333, DDR3-1600. DIMM: PC3-6400, PC3-8500, PC3-10600, PC3-12800. Ємність одного DIMM - до 16Гб.
- DDR4. Частота: між 800 і 2133 MHz. DDR4-1600 – DDR4-4266. DIMM: PC4-12800 – PC4-17000.
- При тому, зовнішня частота (та, що приходить у модуль із материнської плати) – 200-400 MHz. Ємність одного DIMM - до 64Гб.
- Трапляються й інші варіанти. Увага, не всі комбінації розміру модулів та їх внутрішньої організації підтримуються всіма материнськими платами!

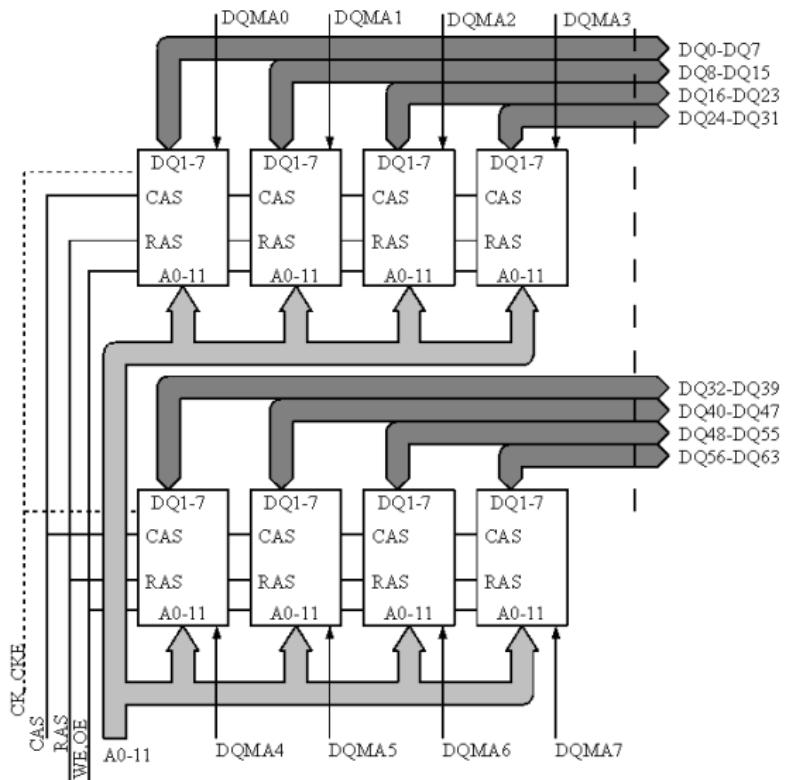
Підсумовуючи

DDR SDRAM Standard	Internal rate (MHz)	Bus clock (MHz)	Prefetch	Data rate (MT/s)	Transfer rate (GB/s)	Voltage (V)
SDRAM	100-166	100-166	1n	100-166	0.8-1.3	3.3
DDR	133-200	133-200	2n	266-400	2.1-3.2	2.5/2.6
DDR2	133-200	266-400	4n	533-800	4.2-6.4	1.8
DDR3	133-200	533-800	8n	1066-1600	8.5-14.9	1.35/1.5
DDR4	133-200	1066-1600	8n	2133-3200	17-21.3	1.2

- Взято тут:
<https://www.transcend-info.com/Support/FAQ-296>.
- Важливий нюанс: через необхідність підтримувати синхронність між модулями – для всіх їх сотень пінів, із зростанням частоти паде кількість модулів, які можуть знаходитися на шині (каналі). Наприклад, DDR3 може "жити" на каналі лише на самоті²³.

²³Можливо, в подальшому це обмеження було знято – не впевнений. О.Ф.

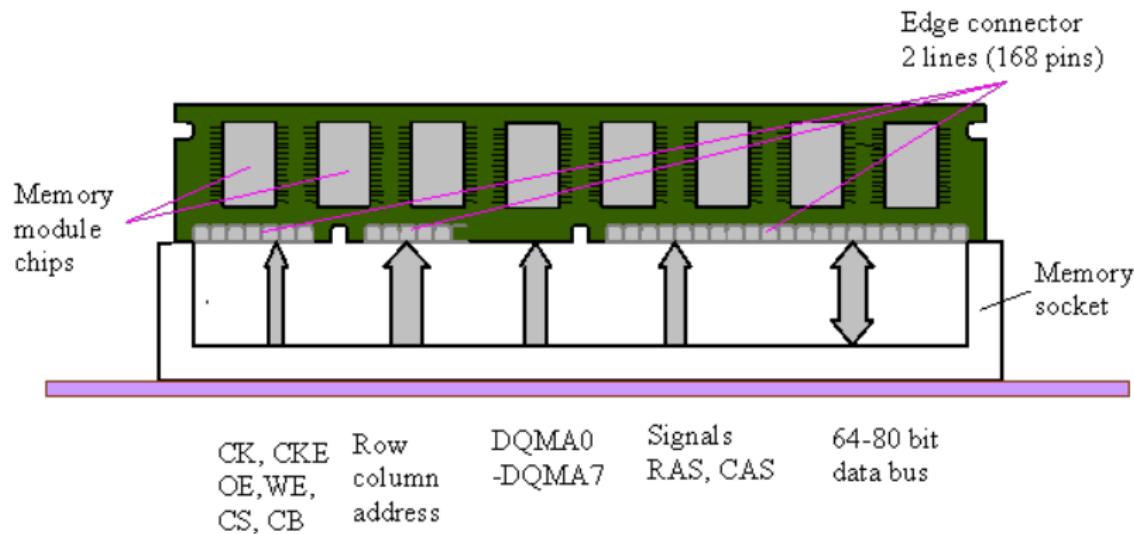
Будова модуля SDRAM



- Взято тут:

<https://edux.pjwstk.edu.pl/mat/264/lec/main93.html>

Схема фізичного виконання модуля SDRAM



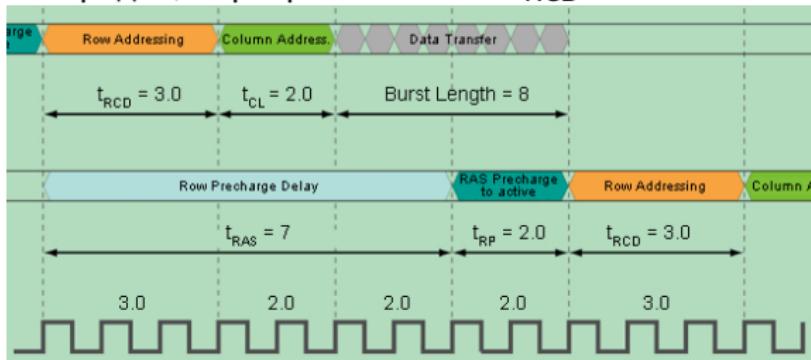
- Взято тут:
<https://edux.pjwstk.edu.pl/mat/264/lec/main93.html>.

Memory timing

- Конструкція SDRAM накладає цілком конкретні вимоги на часові інтервали, протягом яких відповідні сигнали повинні бути активними. Ключових виділяють чотири, вимірюються вони у тривалості тактів.
- Їх прийнято показувати у формі: $CL - T_{RP} - T_{RCD} - T_{RAS}$ – див. наступний слайд.

Memory timing

- CL – CAS latency, кількість тактів між отриманням адреси стовпця і появою правильних даних на виході. Це значення має дотримуватися точно, як RAM, так і контролером пам'яті.
- T_{RCD} – Row Address to Column Address Delay, мінімальна к-сть тактів між вказанням рядка і можливістю вказувати стовпець.
- T_{RP} – Row Precharge Time, мінімальний час між командою "precharge" та доступом до наступного рядка.
- Час доступу до комірки SDRAM, із зміною рядка:
 $T_{RP} + T_{RCD} + CL$.
- T_{RAS} – Row Active Time, мінімальний час між командою активації рядка і precharge. Час, потрібний для внутрішнього оновлення рядка, перекривається із T_{RCD} .



Memory timing

- Іноді наводять ще один, п'ятий параметр – Command Rate, як часто можна CS активувати.
- Параметри ці записуються у спеціальні mode registers модуля пам'яті (чим пізніша версія DDRx, тим їх більше). Сучасні модулі пам'яті містять спеціальну пам'ять конфігурації, Serial Presence Detect (SPD) EEPROM (https://en.wikipedia.org/wiki/Serial_presence_detect), що зберігає ідентифікацію модулів та їх оптимальні конфігурації, вказані виробниками. Ця пам'ять доступна ОС та драйверам по SMBUS, варіанту I2C. За потреби, на деяких MB/модулях, її можна перепрограмовувати.
- Ці параметри часто доступні у BIOS Setup, разом із напругою живлення та використовуються при overclocking – акуратна зміна може підвищити стабільність системи. Неакуратна – фізично вивести з ладу.

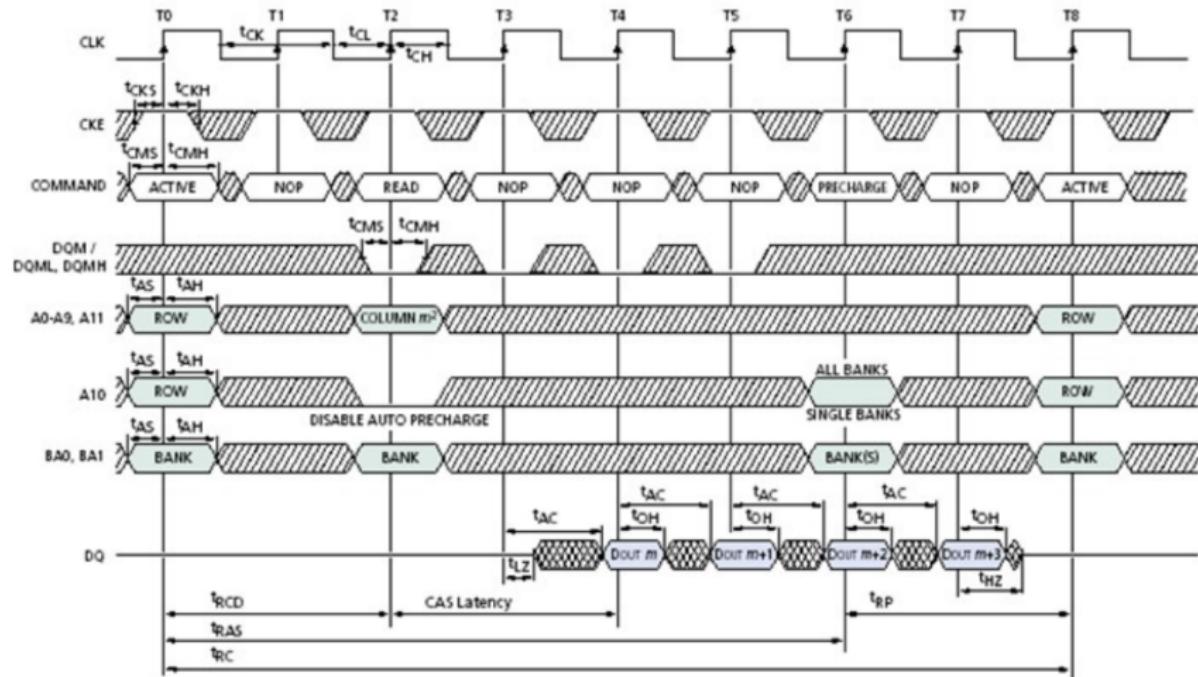
BIOS: memory timing setup

Phoenix - AwardBIOS CMOS Setup Utility
Memory Timing Setting

Parameters	Setting	Current Value	Item Help
Memory Timing Setting	[Expert]		Menu Level 
tCL (CAS Latency)	[4]	4	
tRCD	[4]	4	Select [Expert] to enter timings manually
tRP	[4]	4	
tRAS	[12]	12	
Command Per Clock (CMD)	[1T]	1T	
** Advanced Memory Settings **			
tRRD	[Auto(3)]	3	
tRC	[Auto(22)]	22	
tWR	[Auto(5)]	5	
tWTR	[Auto(9)]	8	
tREF	[Auto]	7.8uS	

Memory timing

- Фактично, параметрів більше – див. діаграму SDR SDRAM:



Memory timing – характерні значення

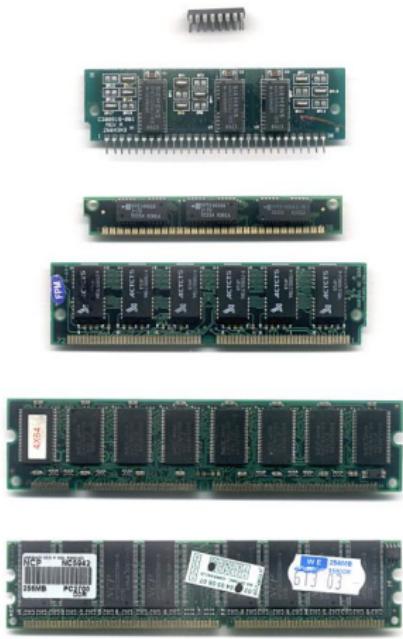
	PC-3200 (DDR-400)				PC2-6400 (DDR2-800)				PC3-12800 (DDR3-1600)			
	Typical		Fast		Typical		Fast		Typical		Fast	
	cycles	time	cycles	time	cycles	time	cycles	time	cycles	time	cycles	time
t_{CL}	3	15 ns	2	10 ns	5	12.5 ns	4	10 ns	9	11.25 ns	8	10 ns
t_{RCD}	4	20 ns	2	10 ns	5	12.5 ns	4	10 ns	9	11.25 ns	8	10 ns
t_{RP}	4	20 ns	2	10 ns	5	12.5 ns	4	10 ns	9	11.25 ns	8	10 ns
t_{RAS}	8	40 ns	5	25 ns	16	40 ns	12	30 ns	27	33.75 ns	24	30 ns

- Взято тут: https://en.wikipedia.org/wiki/Dynamic_random-access_memory.
- Детальніше про функціювання DDR3 SDRAM, команди, реєстри режиму тощо, окрім стандартів вище, див. презентацію: <https://courses.cs.washington.edu/courses/cse467/11wi/lectures/SDRAM.pdf>.

Фізичне виконання

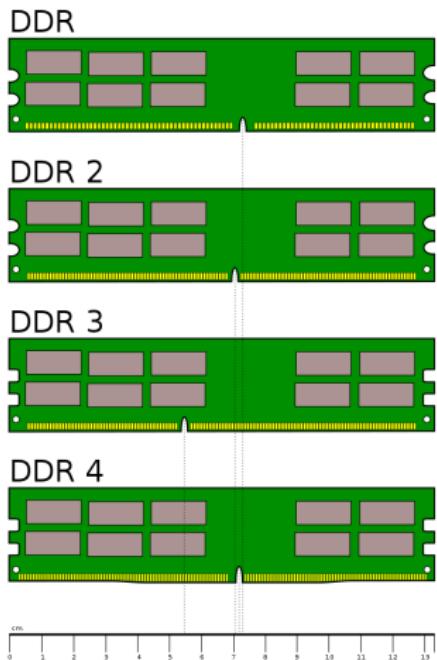
- Дуже давно: DIP, ZIP (вертикальний варіант).
- Давно: SIMM – single in-line memory module, односторонній модуль, схожий на звичні нам зараз (SIPP – варіант із пінами типу ZIP). Піни продубльовано з обох сторін. Найпоширеніші варіанти: 30-pin (8-бітні дані, використовувалися завжди в парі для CPU із 16-бітною шиною, по 4 – із 32-бітною) та 72-pin (32-бітна шина даних, на процесорах із 64-бітною шиною, типу ранніх Пентіумів, теж використовувалися по два).
- Зараз: DIMM – dual in-line memory module, контакти із різних сторін розділені. Типова реалізація DRAM із часів Pentium (середина 90-х). 64-бітна шина даних.
- Зараз: SO-DIMM, small outline DIMM – варіант останнього для ноутбуків, компактніший.
- Кількість пінів дуже різна – див. далі та списки тут:
<https://en.wikipedia.org/wiki/DIMM> і
https://en.wikipedia.org/wiki/Memory_module. Відзначу лише декілька: 100-pin – принтерна SDRAM, 204-pin SO-DIMM DDR3, 240-pin DIMM DDR2 та DDR3.

Фізичне виконання



- Зверху вниз, DIP, SIPP, SIMM (30-pin), SIMM (72-pin), DIMM (168-pin), DDR DIMM (184-pin).
- Взято тут: https://en.wikipedia.org/wiki/Memory_module.

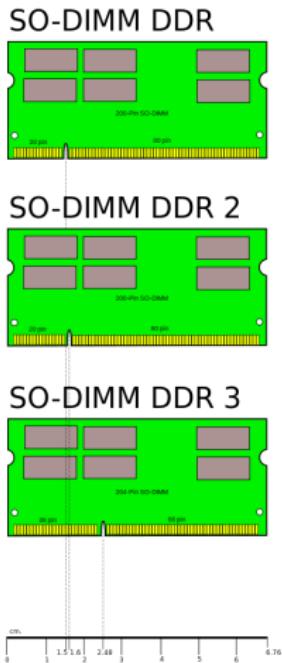
Фізичне виконання – "великі" DIMM



- Різні версії DDR не сумісні між собою електрично чи логічно, тому фізично вони виконані так, щоб їх не можна було випадково вставити у слот, що їх не підтримує.
- Зображення взято тут:

https://en.wikipedia.org/wiki/DDR4_SDRAM

Фізичне виконання – SO-DIMM



- Взято тут: <https://en.wikipedia.org/wiki/SO-DIMM>.

Багатопортова пам'ять – DPRAM

- Пам'ять, що розглядалася вище – однопортова. Вона допускає лише одне звертання за раз.
- Там, де пропускна здатність особливо критична (відеопам'ять, високопродуктивне мережеве обладнання, файл регістрів, пам'ять DSP), використовується також dual-port пам'ять – вона підтримує одночасне читання і запис. (RW)
- Виділяють також two-port²⁴ пам'ять – вона має два порти, кожен із яких може здійснювати і читання і запис²⁵. Наприклад, виконувати одночасно два читання (RR), або читання і запис (RW), або два записи (WW).
- Наскільки мені відомо, у звичайних ПК і серверах – не використовується.

²⁴Тут маємо перекладацьку трудність – це різні види пам'яті, але англо-українські словники вважають їх синонімами.

²⁵Існують й інші інтерпретації цього терміну.

QDR SRAM

- Суміжна до багатопортової пам'яті ідея: Quad Data Rate (QDR) SRAM.
- Відрізняється від DDR тим, що може виконувати читання і запис одночасно. Для цього читання і запис має окремі тактові сигнали і шини даних.
- Розпаралелення досягається, в основному, завдяки можливості дати більше часу на поширення сигналів – заряджання відповідних ліній.
- Однак, це однопортова пам'ять – всередині виконує лише одну операцію за раз.
- Максимальна пропускна здатність, коли читання і запис строго чергуються.

GDDR SDRAM

- GDDR – Graphics DDR SDRAM, варіант такої пам'яті для відеокарт.
- Станом на 2018, існують: GDDR2, GDDR3, GDDR4, GDDR5, GDDR5X, GDDR6.
- Фактично, не є dual-port пам'яттю, але емулює таку "двопортовість", дозволяючи читати і писати одночасно у різні сторінки. Поміж старіших технологій відеопам'яті, траплялися і dual-port, і, навіть triple-port варіанти.
- Ширина шини даних більше в два рази від DDR, 128 біт на канал (повна – 256-512 біт).
- Крім того, використовує всілякі хитрі трюки-оптимізації, користуючись меншим тиском зворотної сумісності.
- Номери не слід співставляти безпосередньо. Наприклад, GDDR5 базується на технології, близькій до DDR3, зокрема, підтримує буфер на 8 слів (в деяких формулуваннях – біт).

ECC Memory

- ECC memory – Error-correcting code memory, пам'ять, здатна виправляти внутрішні помилки, наприклад, викликані космічним випромінюванням.
- Таке випромінювання здатне як інвертувати біти, так і впливати на схеми керування RAM чи CPU.
- Є дані, що на поверхні Землі, під дією космічної радіації, в основному нейтронів²⁶, може псуватися до 5 біт²⁷ із 8Гб за годину²⁸, хоча дані інших досліджень можуть відрізнятися на порядки, прогнозуючи, скажімо, один збій домашнього комп'ютера в 8 місяців²⁹. Це занадто часто навіть для частини домашніх систем.
- Із зростанням висоти проблема різко зростає, потік нейтронів у 300 раз вищий на висоті 10 км – де літають авіалайнери. В космосі все ще гірше.

²⁶ Породжені космічною радіацією частинки помітно "агресивніші" для RAM від фонової радіації. А від мюонів ще й екраниватися важко. На земну поверхню потрапляє близько 10 000 мюонів на квадратний метр за хвилину. Утворюються вони у верхніх шарах атмосфери під дією космічного випромінювання. Надійне екранивання потребує сотень метрів ґрунту. Свинець ефективніший, правда.

²⁷ Звичайно, не кожен зіпсаний біт впливає на результати роботи, але, все ж.

²⁸ Дослідження Гугла, <http://www.cs.toronto.edu/~bianca/papers/sigmetrics09.pdf>.

²⁹ <https://www.microsoft.com/en-us/research/publication/cycles-cells-and-platters-an-empirical-analysis-of-hardware-failures-on-a-million-consumer-pcs/>

ECC Memory – 2

- ECC³⁰ RAM містить додаткові біти, які дозволяють, до певної межі, виправляти помилки та повідомляти про невиправні³¹.
- Типова ECC-пам'ять може виправити всі однобітові помилки та повідомити про всі двобітові в 64-бітному слові, при тому, потребуючи 72 біти на кожних 64 збережених. Зазвичай, використовуються коди Хеммінга (хоча випробовується і багато інших підходів), в контрольних схемах також буває triple modular redundancy – голосування трьох ідентичних ланок.
- Зверніть увагу – одиничне пошкодження в колах вибірки бітів може породжувати багато однобітових і більших помилок.
- ECC модулі можуть логувати як лише невиправні помилки, так і ті, що їм вдалося виправити – це важливо для моніторингу.
- Сучасні процесори, зазвичай, підтримують ECC-пам'ять, однак, її підтримка відсутня у дешевших материнських платах, особливо – користувачьких, на противагу серверним.
- Також, сучасні процесори використовують ECC у своїх кешах.

³⁰Error-correcting codes.

³¹RAM parity – варіант "для бідних", пам'ять, здатна повідомити про поодинокі бітові інверсії. Зараз мало використовується. На старих материнських платах, із DIP-пам'яттю, іноді можна було помітити, що на байт іде 9 мікросхем із 1-бітовим словом – +1 для біта парності.

ECC Memory – 3

- Іноді трапляються модулі пам'яті, які можуть працювати у режимі ECC і без корекції, із трішкою різним розміром³².
- Перезавантаження комп'ютера із ECC може бути повільним, через необхідність занулити всю RAM та порахувати ECC, однак, сповільнення в роботі не повинно бути у процесорів із вбудованим контролером пам'яті (на старіших CPU могло складати 2-3%).
- На деяких системах використовується техніка Memory scrubbing – регулярного читання всієї пам'яті та запису її назад, для виправлення однобітових, щоб ліквідувати їх, мінімізуючи шанс виникнення двобітових помилок.

³²Автор стикався із такими, але не маю детальнішої інформації.

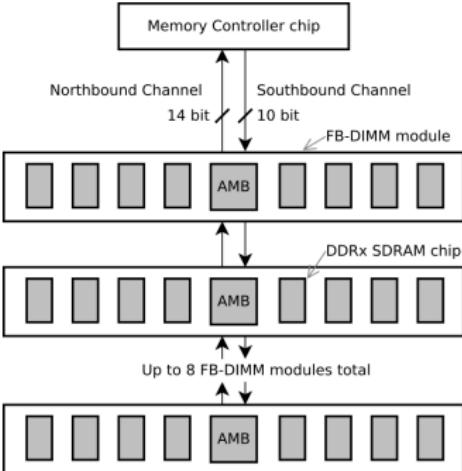


Registered (buffered) memory – RDIMM

- Спеціальна пам'ять, що містить реєстр між модулем пам'яті і її контролером.
- Такі реєстри зменшують електричне навантаження на контролер і дозволяють встановити більше модулів на один канал.
- Однак, така пам'ять дорожча та трішки повільніша – потребує один зайвий такт на кожен обмін (лише один "на всіх" в пакетному, burst, режимі).
- З іншого боку, завдяки перемежуванню (interleaving) кількох модулів, сумарна продуктивність системи може й зрости.
- Зазвичай, реєстрова пам'ять має і ECC – просто тому, що використовується у системах, де розміри пам'яті більші, а надійність важливіша, ніж в ПК.
- Буферизують лише контрольні сигнали та адреси звертання.

FB-DIMM

- Існує також FB-DIMM – Fully Buffered DIMM, яка буферизує і лінії даних. Вона базується на advanced memory buffer (AMB), який послуговується послідовним інтерфейсом³³ між AMB і контролером пам'яті. Також він займається ECC.
- Як Registered memory, так і FB-DIMM потребують спеціалізованих материнських плат – вони ні механічно (зазвичай і на щастя) ні електрично, несумісні як із звичайними DDR, так і між собою.



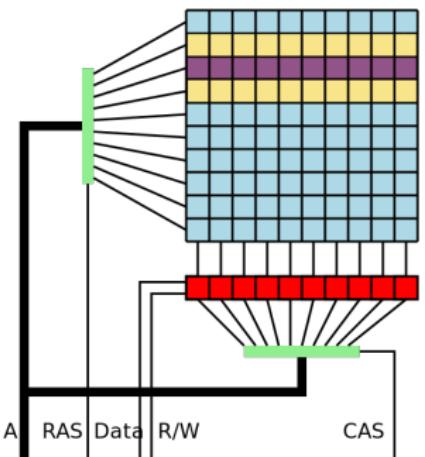
³³14 ліній даних до RAM, 10 ліній назад, диференційнимиарами, частота в 12 раз вища за частоту пам'яті. Див. https://en.wikipedia.org/wiki/Fully_Buffered_DIMM.

Row hammer

- Комірки в сучасних DRAM розташовані дуже тісно, для зарядки відповідних конденсаторів потрібно лічені десятки тисяч електронів³⁴.
- Завдяки цьому, взаємодіючи із комірками пам'яті, можна впливати на сусідні комірки. Мова, звичайно, про фізично близькі комірки.
- Це можна використати для того, щоб маніпулюючи доступною зловмиснику пам'яттю, впливати на недоступну йому. Така атака називається Row hammer attack.

³⁴Щоб зрозуміти, наскільки це мало – число Авогадро ще хтось пам'ятає?

- Часта активування якогось рядка пам'яті спричиняє флюктуації на лініях сусідніх рядків.
- Такі флюктуації прискорюють розрядку конденсаторів пам'яті.
- Особливо це ефективно, якщо "смикати" сусідні рядки з обох боків:



- Дослідження показали, що для цього достатньо порядку 140 000 послідовних звертань (із скиданням кешу – щоб змусити систему виконувати фізичне звертання до пам'яті).
- Що цікаво, результат слабо залежить від температури, зате сильно – від паттерну бітів, що в пам'яті зберігаються.

- Один із способів боротися – використання ECC, інший – збільшення частоти оновлення. Існують також більш вишукані методи – лічильники активації рядків, "розумне" оновлення рядків, які можуть бути під атакою тощо.
- MemTest86, коли комп'ютер завантажено в UEFI режимі, вміє тестиувати на вразливість до цієї атаки.
- DDR і DDR2 практично не вразливі. DDR3 і DDR4 – вразливі.
- Існують демонстрації (Exploits) виконання цієї атаки, в тому числі такі³⁵: що надають доступ до всієї пам'яті непривілейованому Linux-процесу; не прив'язані до x86, включають смартфони; можуть бути реалізовані на JavaScript. На щастя, зробити все це одночасно – важко або й неможливо.
- Відправною точкою для детальнішого ознайомлення можуть служити посилання тут:
https://en.wikipedia.org/wiki/Row_hammer, зокрема
<https://github.com/google/rowhammer-test>.

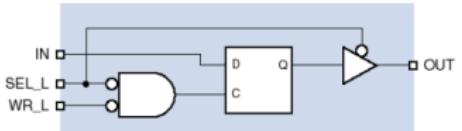
³⁵ Важливим для практичного застовування є техніка memory spraying.

Нотатки щодо SRAM

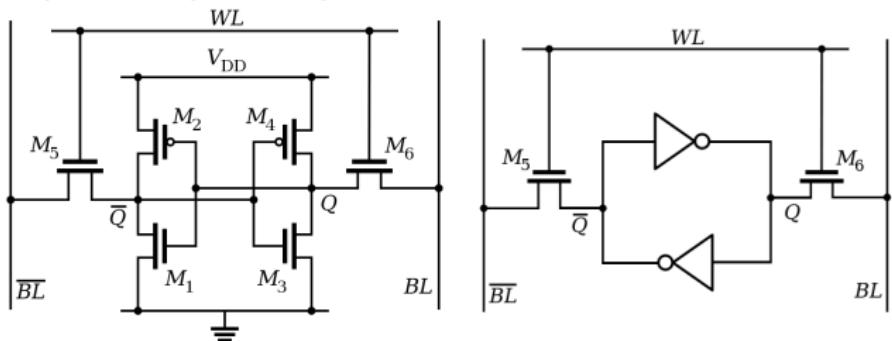
- Статична пам'ять не потребує регулярного оновлення. Тому простіша у роботі і не містить додаткових кіл.
- На практиці використовуються менші її об'єми. Тому адреса, зазвичай, передається зразу, а не в два етапи.
- Існують як синхронні, так і асинхронні варіанти.
- Існують підходи типу DDR для підвищення продуктивності, там, де вона особливо критична. Див, наприклад, <https://www.cypress.com/documentation/application-notes/an4065-qdr-ii-qdr-ii-ddr-ii-and-ddr-ii-design-guide>
- Крім кешів, часто використовується в мікроконтролерах (для таких малих об'ємів ціна не критична, а SRAM простіша в роботі), існують дискретні мікросхеми, див. серії 6264 та 62C256.

Комірка SRAM

- SRAM можна побудувати на базі розглянутих раніше тригерів³⁶:



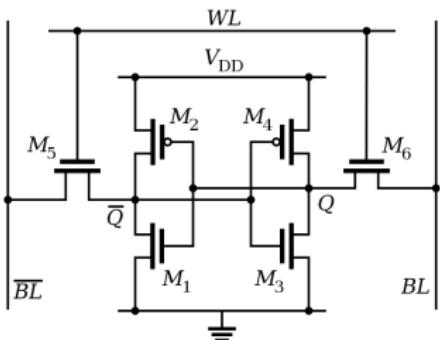
- Однак, зазвичай, використовують більш економні спеціалізовані 6-транзисторні комірки:



- Існують також 4-транзисторні комірки, із більшим енергоспоживанням – у них постійно протікає помітний струм, та комірки із більшою за 6 кількістю транзисторів.

³⁶ Для простоти, їх роблять асинхронними, за потреби, синхронність забезпечується схемами керування.

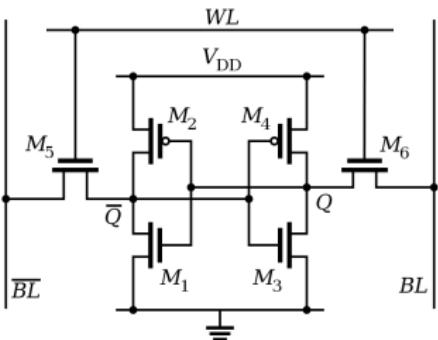
Комірка SRAM – читання



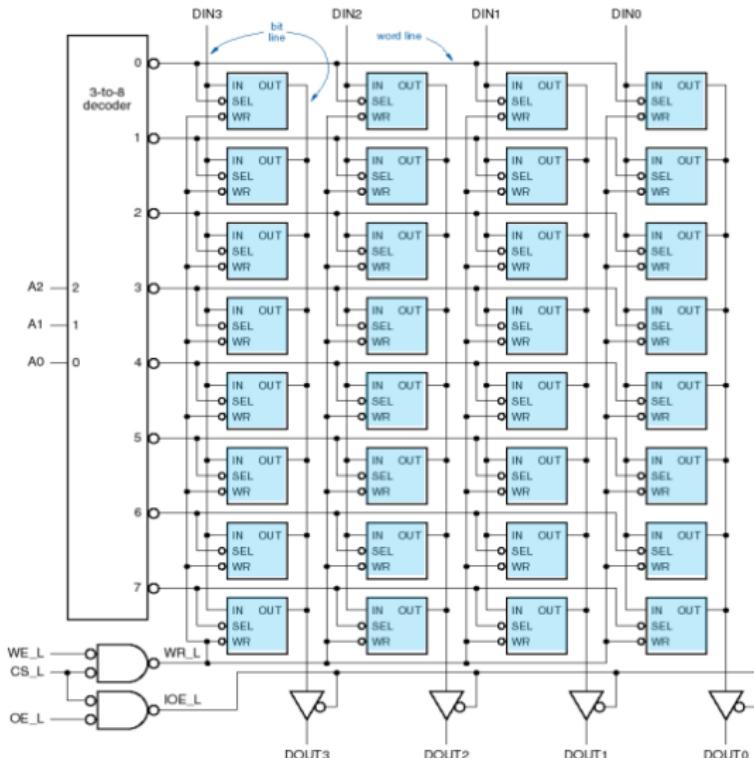
- Standby: лінія WL – Word Line пасивна. Тоді транзистори M1-M4 від'єднані від Bit Line (BL) і просто зберігають своє значення.
- Reading: спочатку на BL і \overline{BL} подається високий рівень, (precharging), тоді активується WL. Тоді, на лінії, де має бути 1, напруга буде трохи меншою. Підсилювач, sense amplifier³⁷, визначає, де – на BL чи на \overline{BL} , є зараз 1 і повертає відповідне значення. Чим швидше підсилювач здатен відчути цю різницю, тим швидше працюватиме пам'ять.

³⁷ https://en.wikipedia.org/wiki/Sense_amplifier

Комірка SRAM – запис

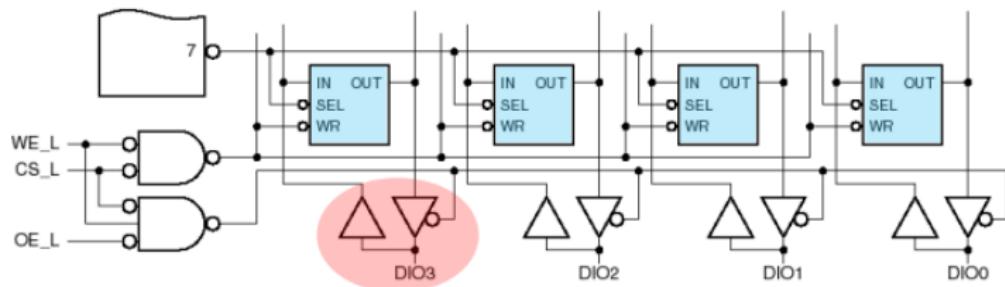


- Writing: на BL та \overline{BL} виставляються потрібні значення, активується WL . Так як лініями BL керують драйвери, потужніші за $M1-M4$, вони перетягають комірку у потрібний стан.
(Зрозуміло, що $M5$ та $M6$ також мають бути потужнішими за $M1-M4$ – пропускати більші струми).
- Для кращого розуміння – побавтеся у симулаторі (<http://www.falstad.com/circuit/index.html>, Circuits-Sequential logic-Static RAM).
- Також, див. <https://www.slideshare.net/soveran/sram>.

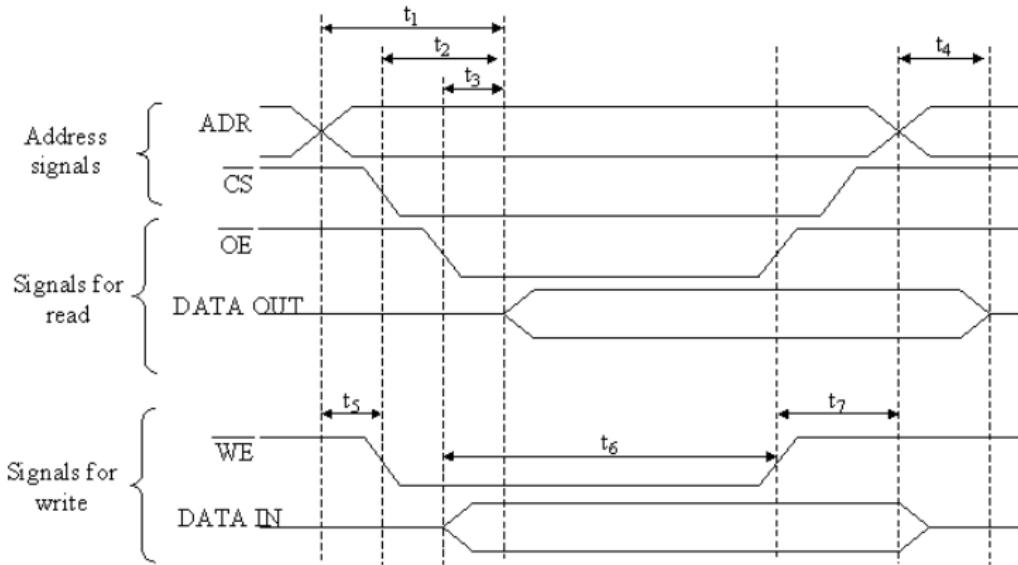


- У виконанні із тригерів, модуль SDRAM виглядатиме якось так – зображення вам відоме!
- Взято тут:
<https://www.slideshare.net/abhilash128/lec-21-16642228>





- Виводи часто – двонаправлені.



- Зображення взято тут:
<https://edux.pjwstk.edu.pl/mat/264/lec/main93.html>.
- Див. також інтерфейс SRAM для ARM: <https://developer.arm.com/docs/ddi0380/g/functional-overview/functional-operation/sram-interface-timing-diagrams> та інтерфейс FSMC (flexible static memory controller) для STM32.

