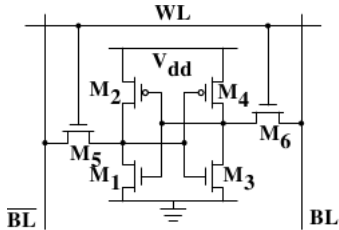


# Pamięć DRAM

Filip Pacanowski

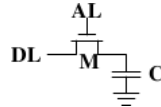
3.12.2014

# Statyczny i dynamiczny RAM



## SRAM

- 6 tranzystorów
- niemal natychmiastowy dostęp do stanu

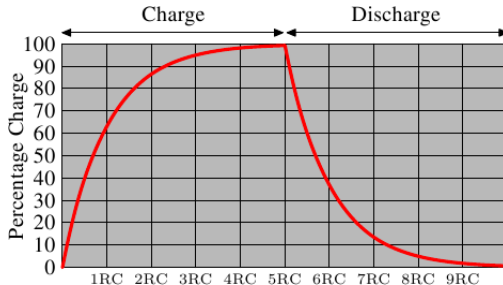


## DRAM

- tranzystor i kondensator
- wymaga cyklicznego odświeżania
- tania w produkcji i łatwa do gęstego upakowania

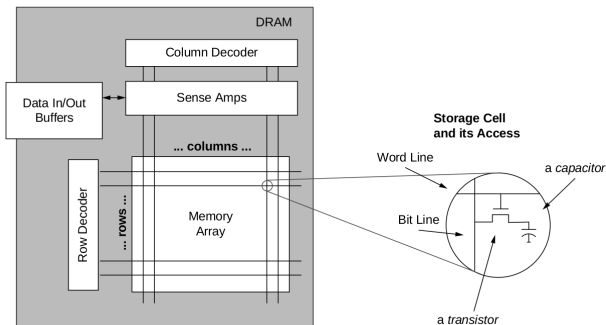
## Problemy z pamięcią opartą na kondensatorach

- wyciek elektronów
- konieczność użycia wzmacniaczy(ang. *sense amplifiers*)
- opóźnienie związane z rozładowywaniem kondensatora



Rysunek: Ładowanie/rozładowanie kondensatora

# Macierz DRAM

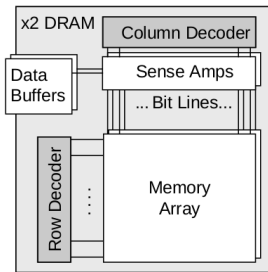


Jak odczytać pojedynczy bit?

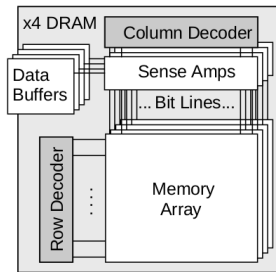
- trzeba odczytać całą wiersz
- precharging bitlines
- ustawienie stanu wysokiego na odpowiednim *wordline*

# Bank

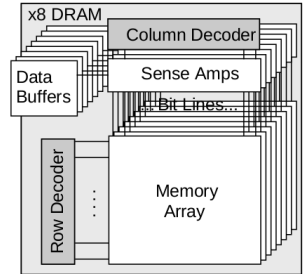
- macierze grupowane są w większe jednostki zwane bankami
- każda macierz w banku dostaje identyczne wejście
- pozwala to na równoległy dostęp do kilku bitów



x2 DRAM

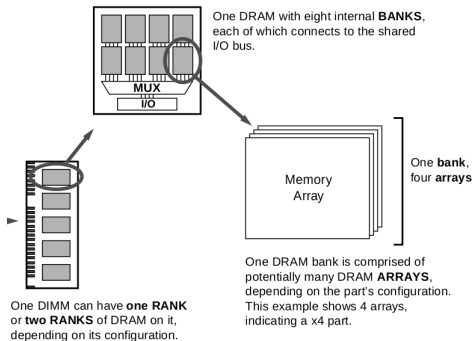


x4 DRAM



x8 DRAM

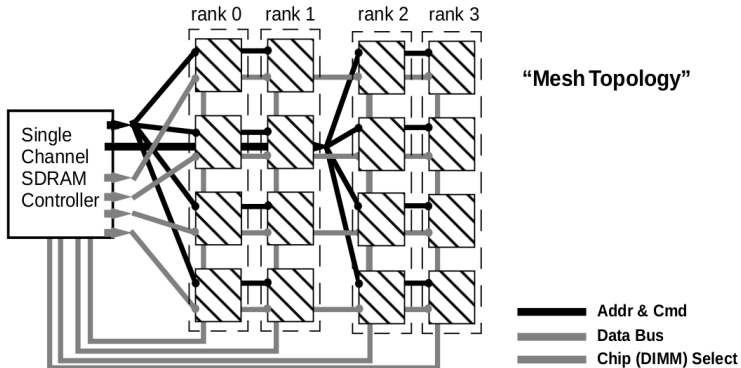
# Kość



- pojedyncze urządzenie DRAM składa się z wielu banków
- banki mogą działać niezależnie
- *rank* to zbiór współpracujących urządzeń
- jedna kość ma na ogół jeden lub dwa ranki

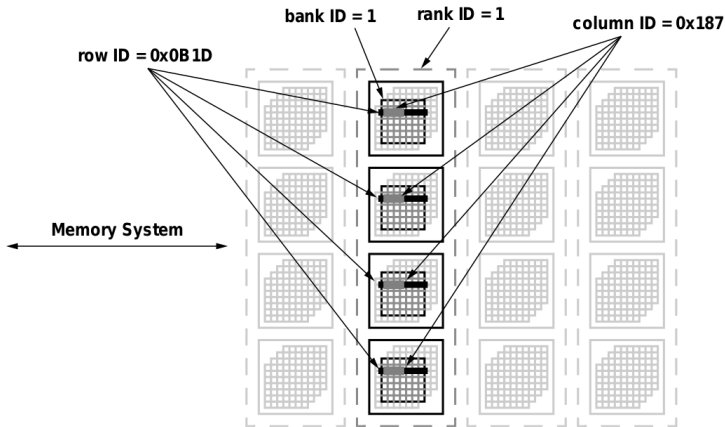
# Topologia pamięci SDRAM

- Jedno żądanie obsługuje tylko jeden rank aktywowany przez sygnał Chip Select
- Szyna danych (na ogół 64 bity) jest podzielona między urządzenia DRAM



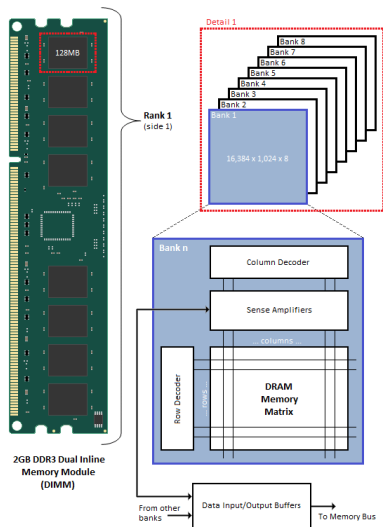
## Miejsce danych w pamięci

Pojedyncze słowo znajduje się w jednym ranku, ale różnych bankach



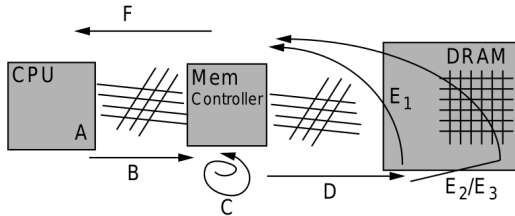


# Przykład (prawie) współczesny



- 16 urządzeń w 2 rankach
- w każdym urządzeniu 8 banków
- w każdym banku 16,384 wierszy i 1,024 kolumny
- bank przechowuje 8 bitów na każdym przecięciu wiersza z kolumną (x8 DRAM)
- łącznie  $16 * 8 * 16,384 * 1,024 * 8 = 2\text{GB}$

## Opóźnienia przy odczycie z pamięci



A: Transaction request may be delayed in Queue  
B: Transaction request sent to Memory Controller  
C: Transaction converted to Command Sequences  
(may be queued)

D: Command/s Sent to DRAM

E<sub>1</sub>: Requires only a **CAS** or

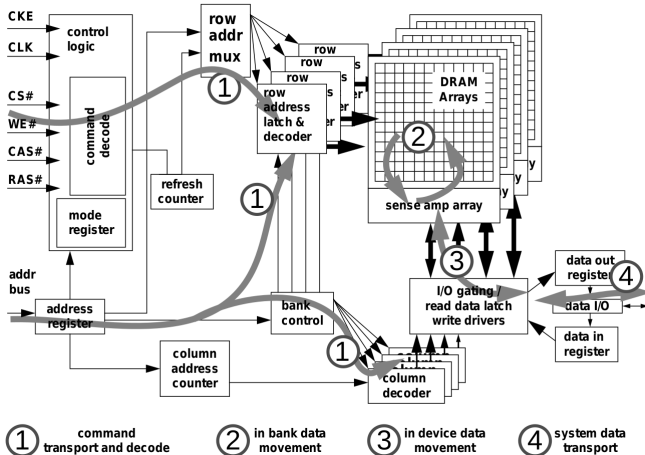
E<sub>2</sub>: Requires **RAS + CAS** or

E<sub>3</sub>: Requires **PRE + RAS + CAS**

F: Transaction sent back to CPU

DRAM Latency = A + B + C + D + E + F

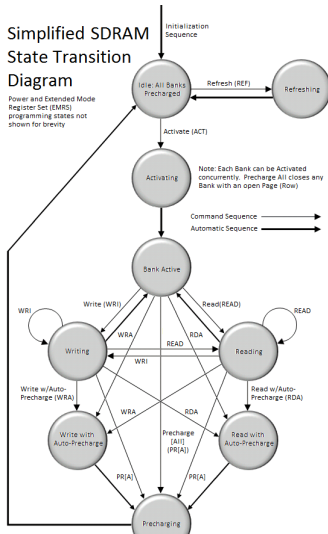
# Blizszy rzut oka na urządzenie DRAM



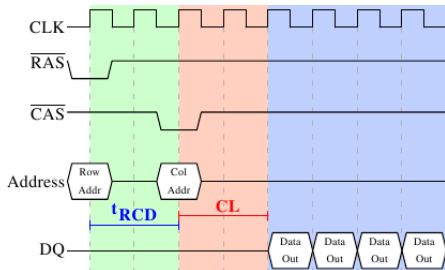
# Maszyna stanów DRAM

## Simplified SDRAM State Transition Diagram

Power and Extended Mode  
Register Set (EMRS)  
programming states not  
shown for brevity

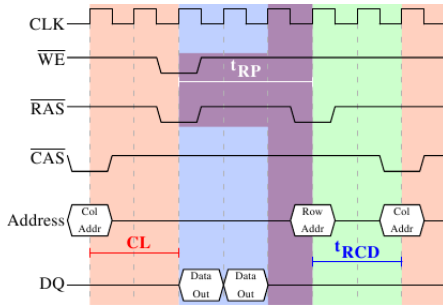


# Odczyt



- przekazanie adresu odbywa się w 2 częściach: najpierw adres wiersza, potem kolumny
- urządzenie musi przygotować wiersz do odczytu, trwa to RCD cykli
- kontroler decyduje czy zamknąć wiersz

# Precharge



- zanim kontroler wyśle nowy adres wiersza musi wstępnie naładować wzmacniacze
- ładowanie trwa  $t_{RP}$  cykli, ale może odbywać się jednocześnie z transferem danych z poprzedniego zapytania

## Opis opóźnień

Producenci typowo podają opóźnienia w postaci 4 liczb w-x-y-z.  
Na przykład 6-6-6-18 (kość DDR3)

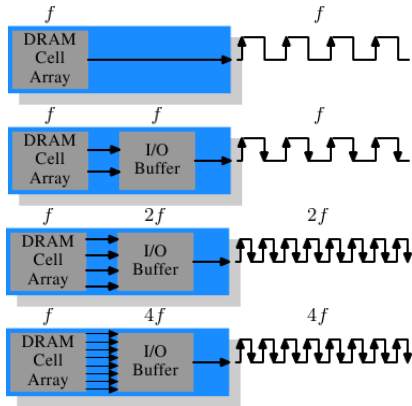
- w = CAS Latency (CL)
- x = RAS-to-CAS delay (tRCD)
- y = RAS Precharge (tRP)
- z = Active to Precharge delay (tRAS)

## Opóźnienia w 3 przypadkach

- bank nieaktywny(idle) =  $t_{RCD} + CL$
- bank aktywny, właściwy wiersz otwarty =  $CL$
- bank aktywny, inny wiersz otwarty =  $t_{RP} + t_{RCD} + CL$



# DDR<sub>x</sub>



- zwiększanie częstotliwości urządzeń DRAM jest kłopotliwe, ze względu na pobór mocy i stabilność układu
- DDR<sub>x</sub> wprowadza rozwiązania pozwalające na zwiększenie przepustowości bez zwiększania częstotliwości pamięci

## Bibliografia (i źródła obrazków)

- Bruce Jacob; Spencer Ng; David Wang (2010). Memory Systems: Cache, DRAM, Disk.
- Ulrich Drepper. What Every Programmer Should Know About Memory <http://lwn.net/Articles/250967/>
- Rajinder Gill. Everything You Always Wanted to Know About SDRAM (Memory) But Were Afraid to Ask <http://www.anandtech.com/show/3851/everything-you-always-wanted-to-know-about-sdram-memory-but-were-afraid-to-ask>

# Koniec

Dziękuję za uwagę. Pytania?