DRAM CONTROLLER

Mahdi Nazm Bojnordi

Assistant Professor

School of Computing

University of Utah

Overview

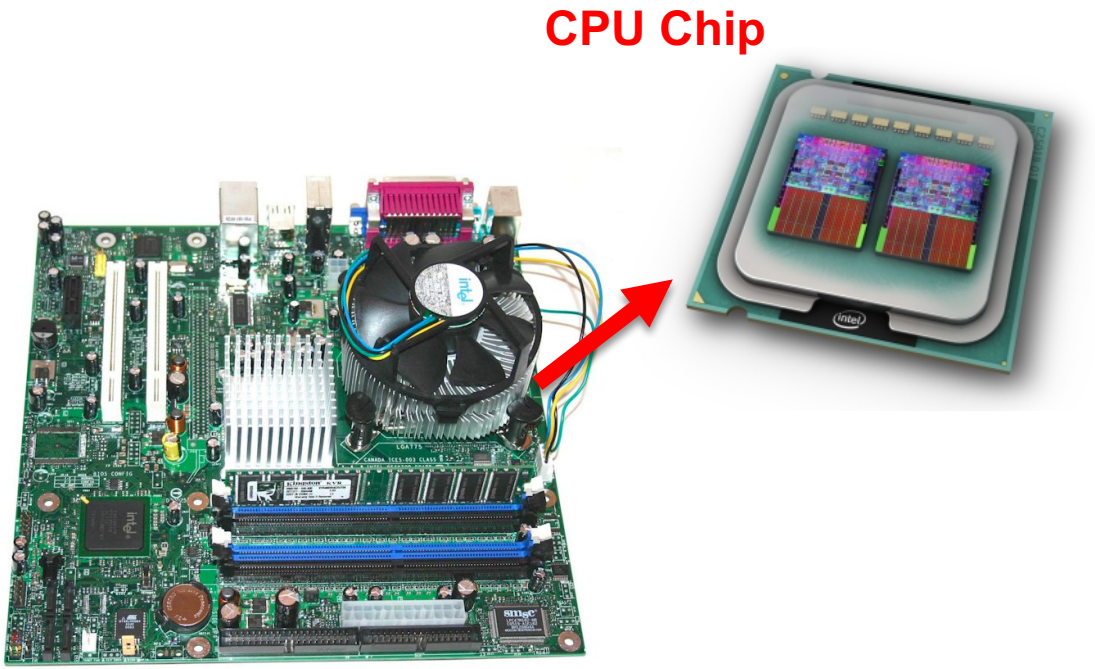
- Announcement
  - Homework 4 will be released on Nov. 28th
- This lecture
  - DRAM control
  - DRAM timing
  - DRAM hierarchy
    - Channel, bank

Recall: DRAM System

- DRAM chips can perform basic operations

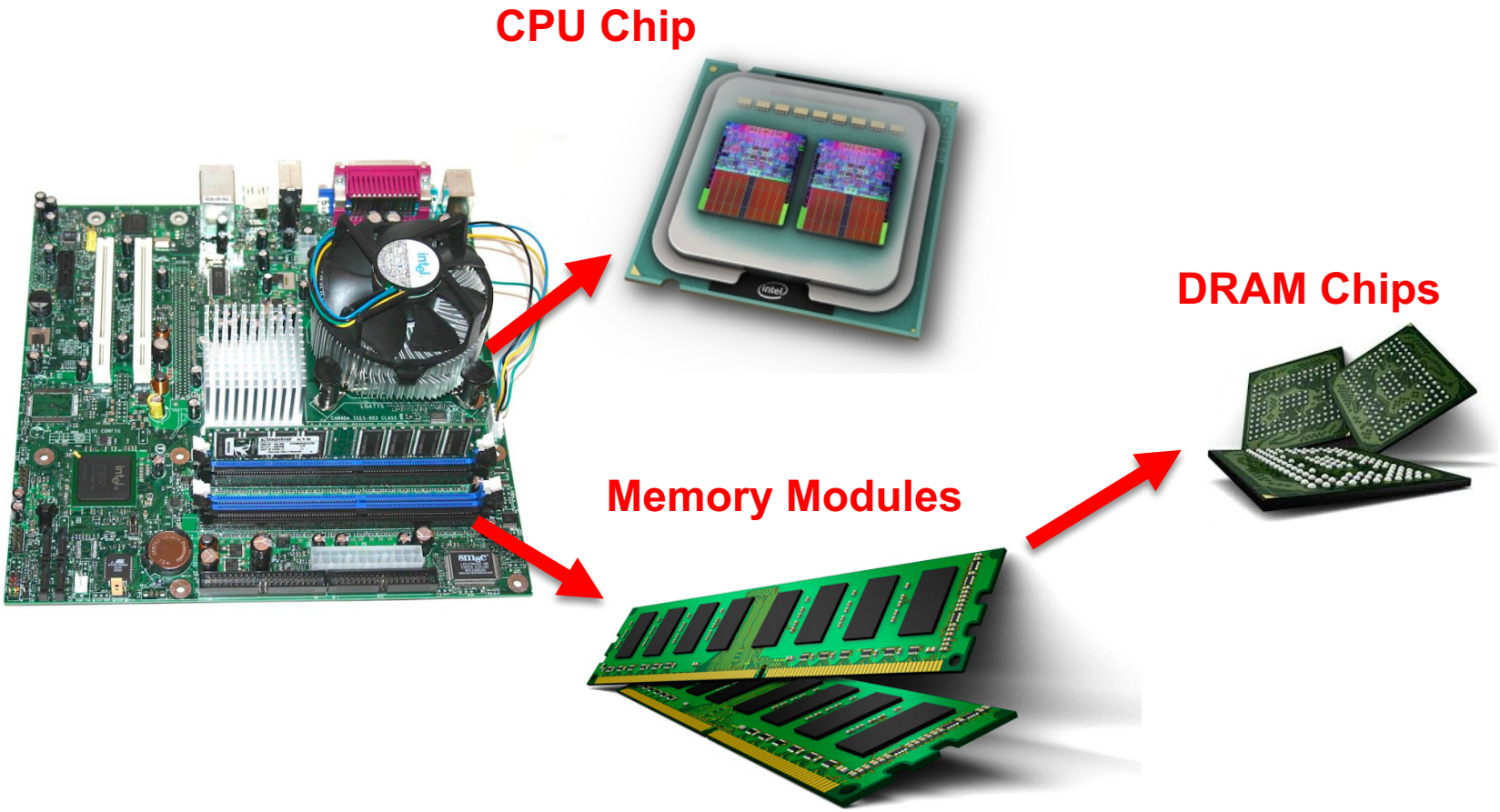
Recall: DRAM System

- DRAM chips can perform basic operations



Recall: DRAM System

- DRAM chips can perform basic operations

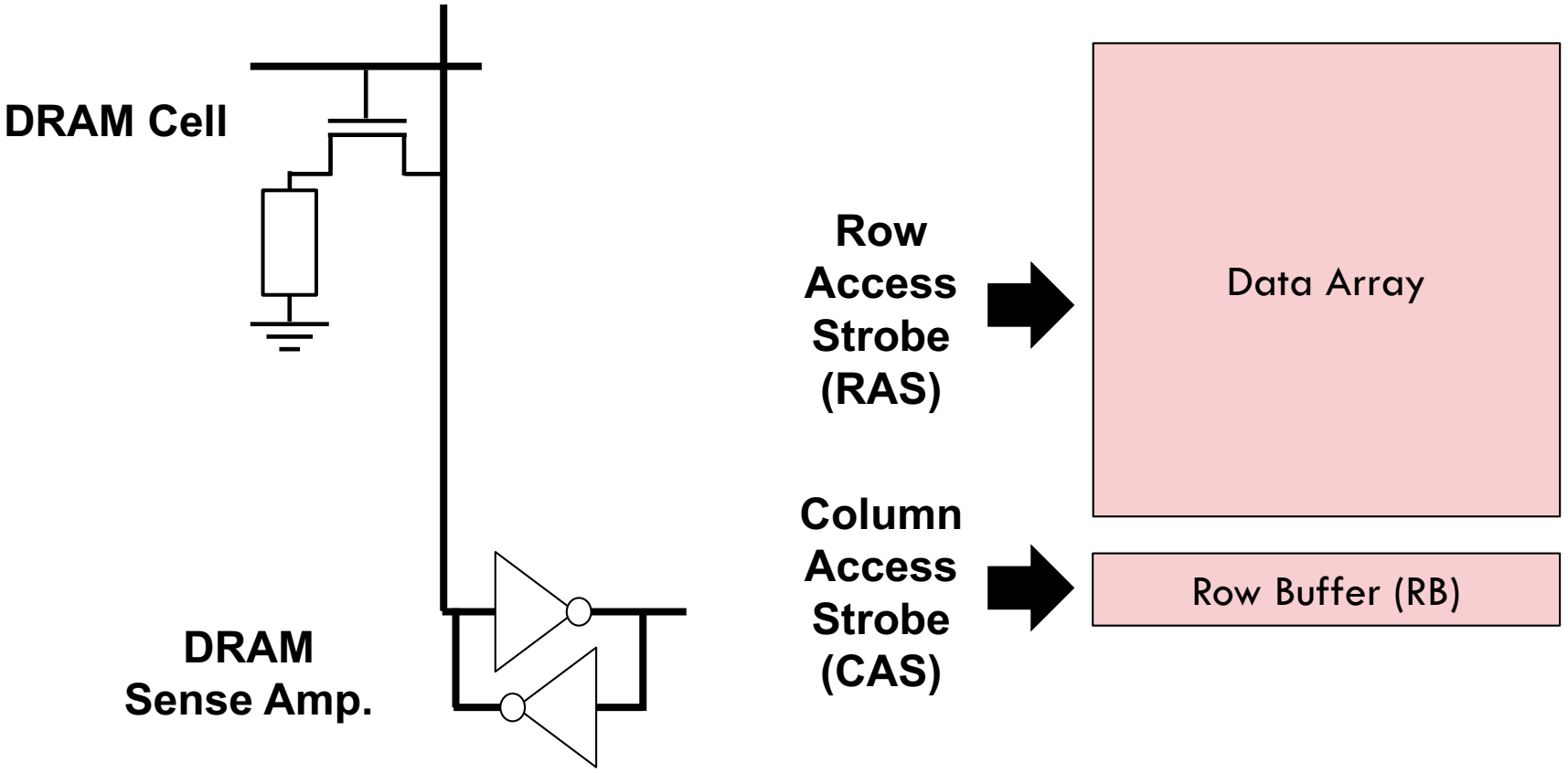


Recall: DRAM Operations

- Main DRAM operations are
  - Precharge bitlines to prepare subarray for activating a wordline
  - Activate a row by connecting DRAM cells to the bitlines and start sensing
  - Read the contents of a data block from the row buffer
  - Write new contents for data block into the row buffer
  - Refresh DRAM cells
    - can be done through a precharge followed by an activate

DRAM Row Buffer

- All reads and writes are performed through RB

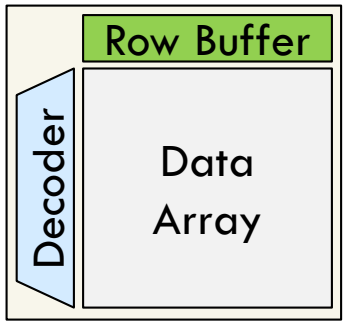


DRAM Row Buffer

- Row buffer holds a single row of the array
  - A typical DRAM row (page) size is 8KB
- The entire row is moved to row buffer; but only a block is accessed each time
- Row buffer access possibilities
  - Row buffer hit: no need for a precharge or activate
    - ~20ns only for moving data between pins and RB
  - Row buffer miss: activate (and precharge) are needed
    - ~40ns for an empty row
    - ~60ns for on a row conflict

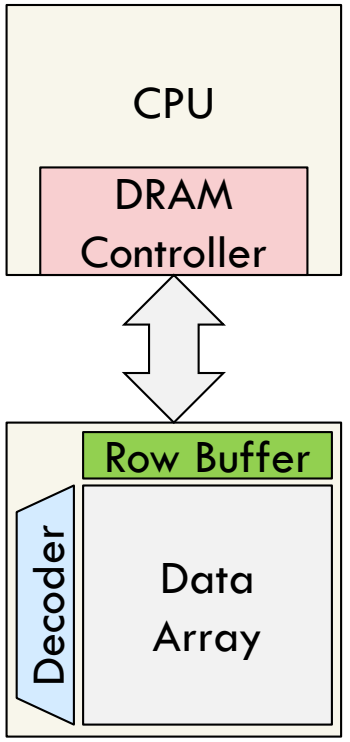
DRAM Control

- DRAM chips have no intelligence
  - An external controller dictates operations
  - Modern controllers are integrated on CPU
- Basic DRAM timings are
  - t_{CAS}: column access strobe (RD→DATA)
  - t_{RAS}: row active strobe (ACT→PRE)
  - t_{RP}: row precharge (PRE→ACT)
  - t_{RC}: row cycle (ACT→PRE→ACT)
  - t_{RCD}: row to column delay (ACT→RD/WT)



DRAM Control

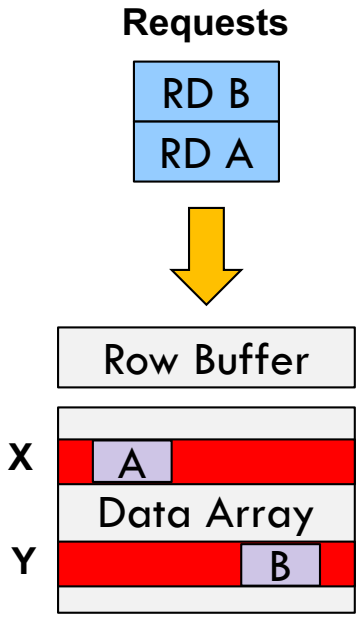
- DRAM chips have no intelligence
  - An external controller dictates operations
  - Modern controllers are integrated on CPU
- Basic DRAM timings are
  - t_{CAS}: column access strobe (RD→DATA)
  - t_{RAS}: row active strobe (ACT→PRE)
  - t_{RP}: row precharge (PRE→ACT)
  - t_{RC}: row cycle (ACT→PRE→ACT)
  - t_{RCD}: row to column delay (ACT→RD/WT)



Enforcing Timing

DRAM Timing Example

- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$



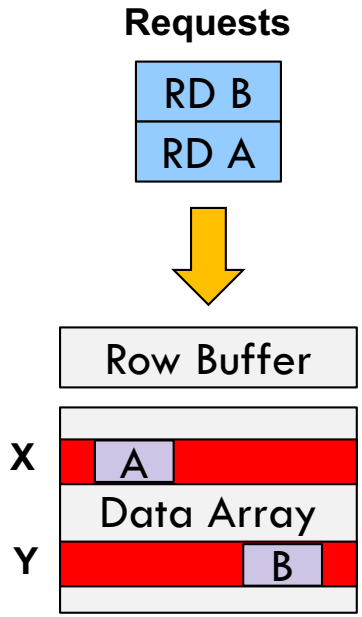
Cmd ----------

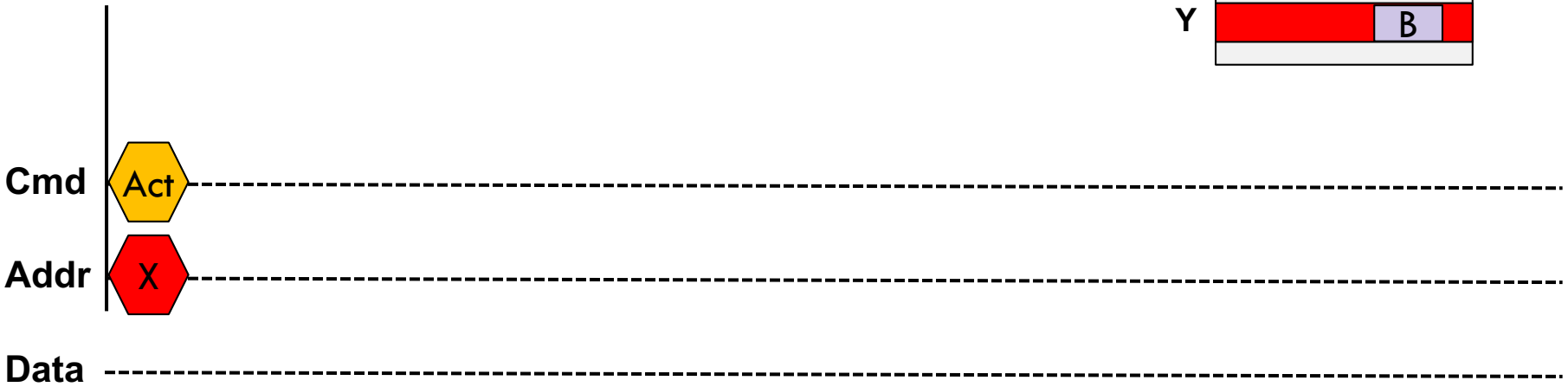
Addr ----------

Data ----------

DRAM Timing Example

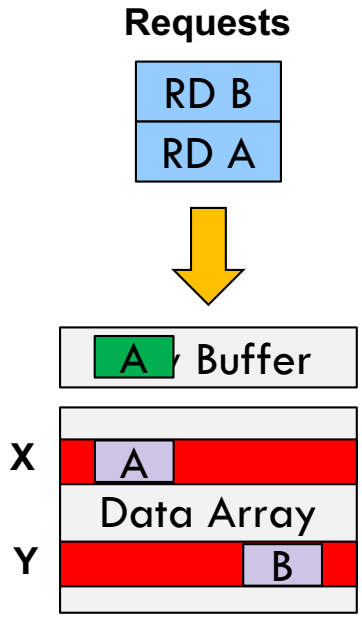
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

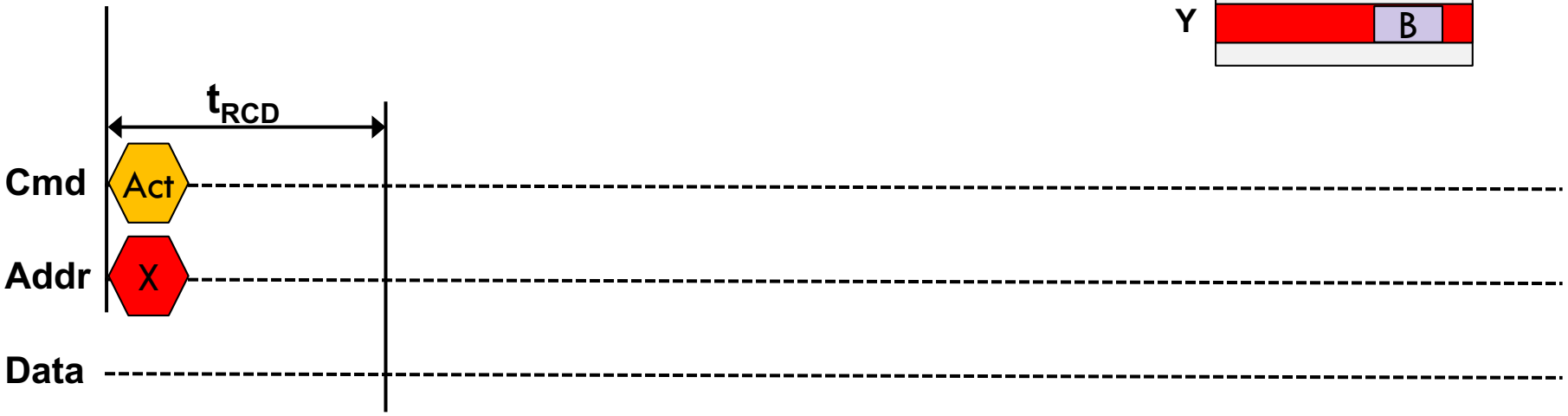




DRAM Timing Example

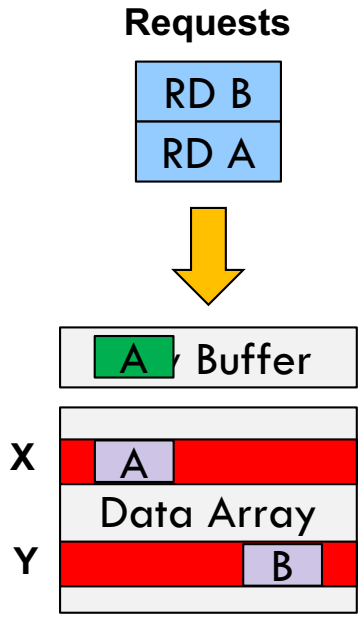
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

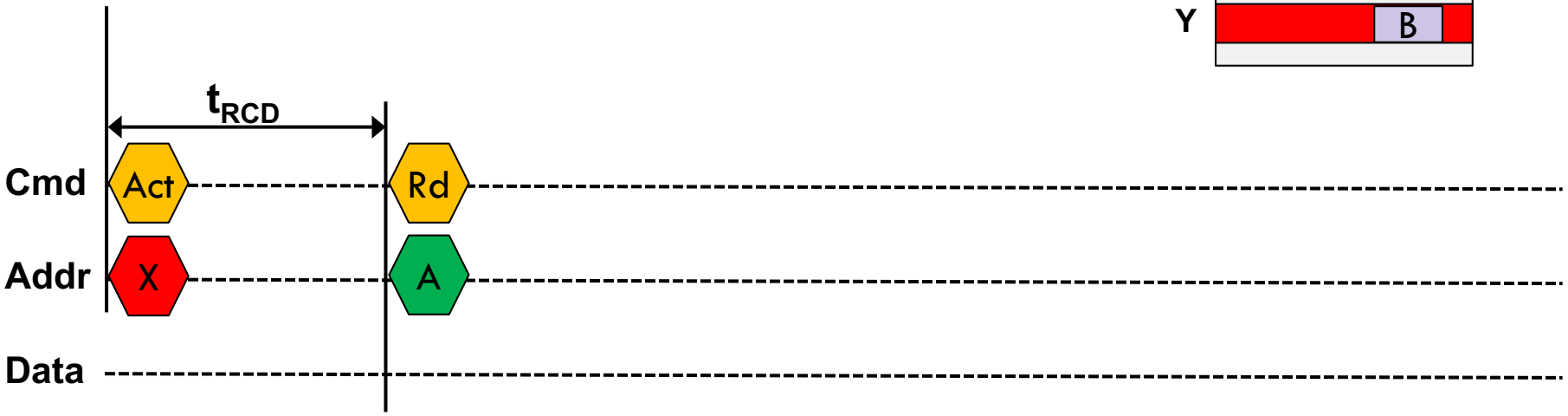




DRAM Timing Example

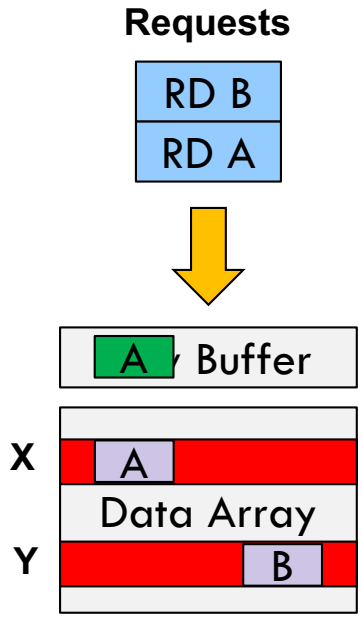
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

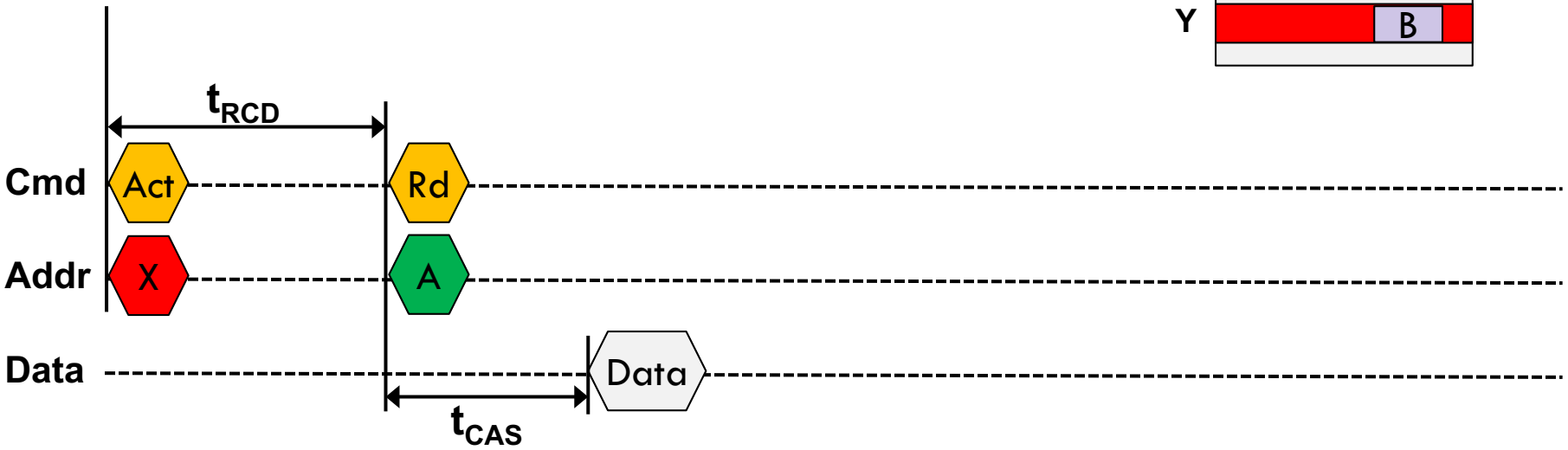




DRAM Timing Example

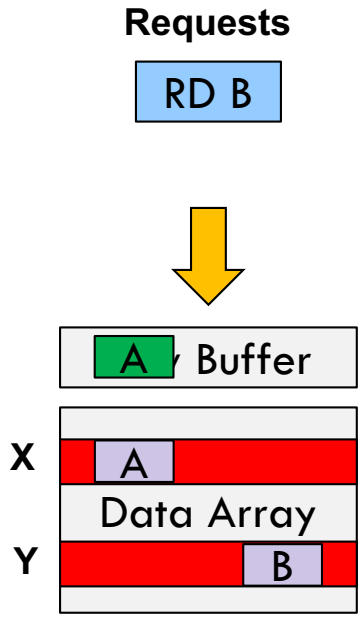
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

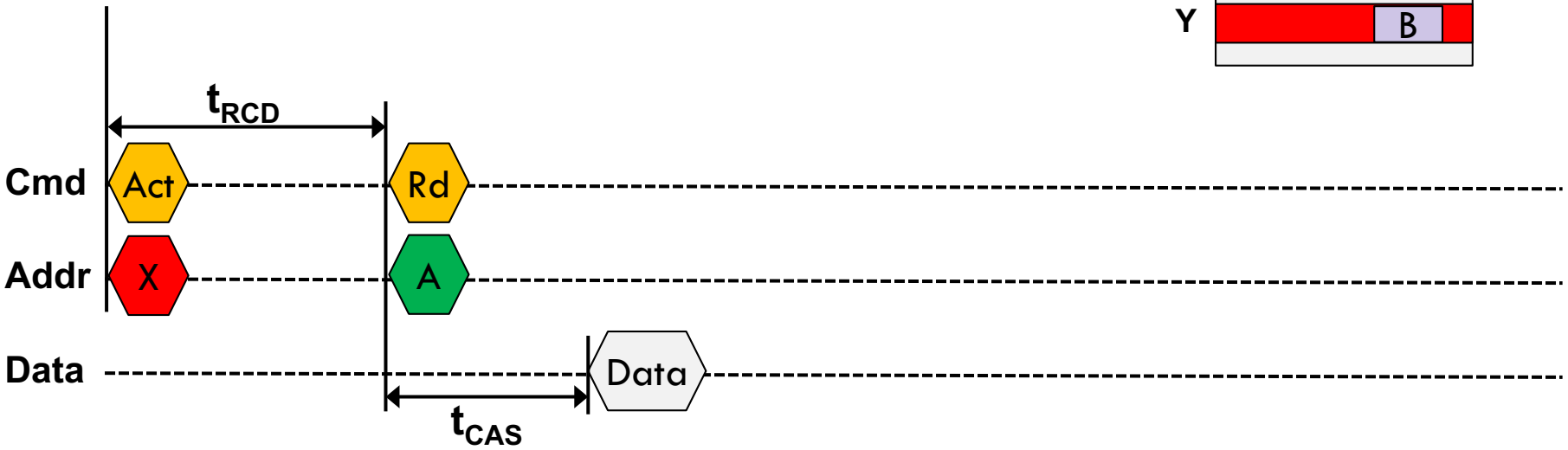




DRAM Timing Example

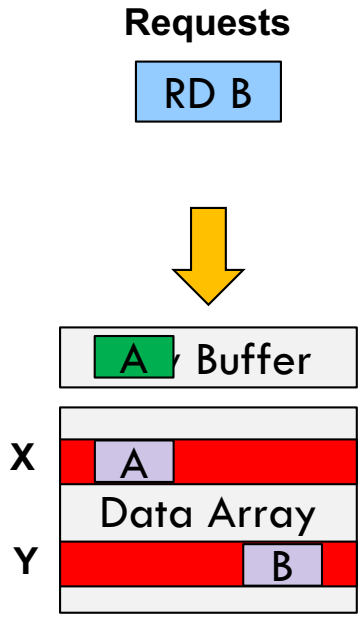
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

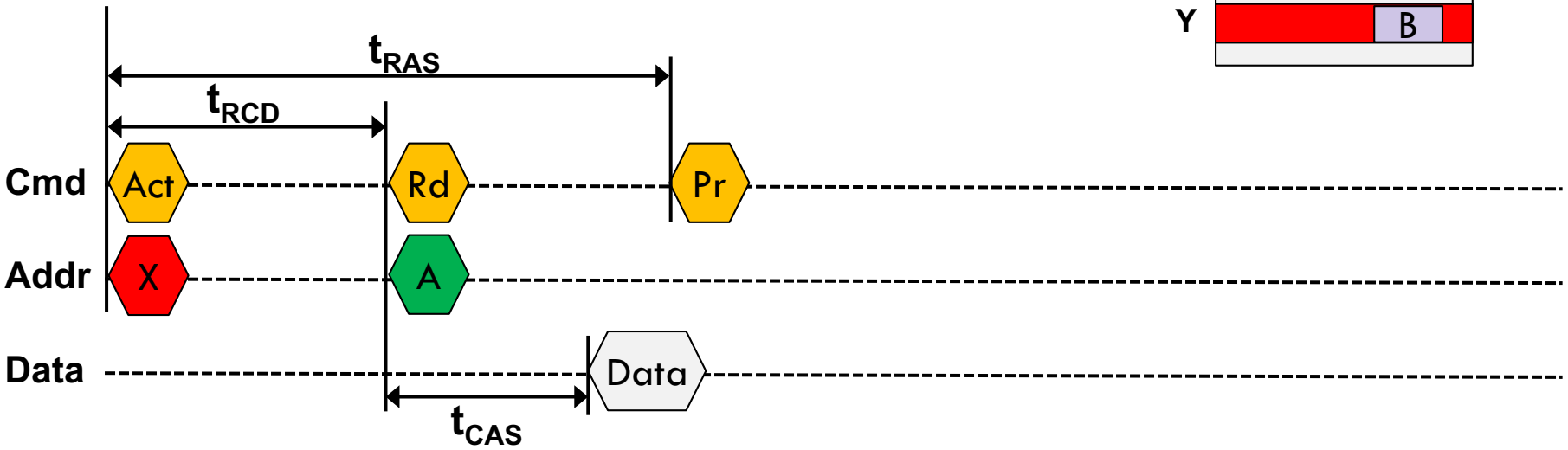




DRAM Timing Example

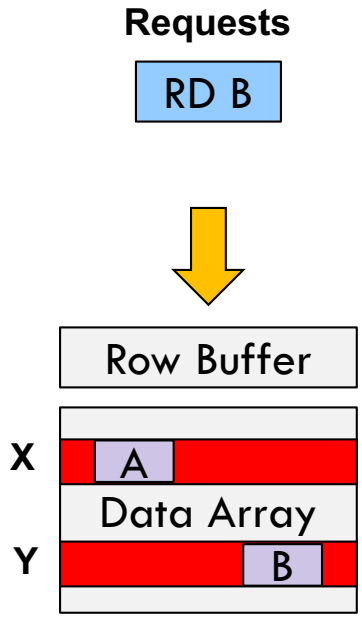
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

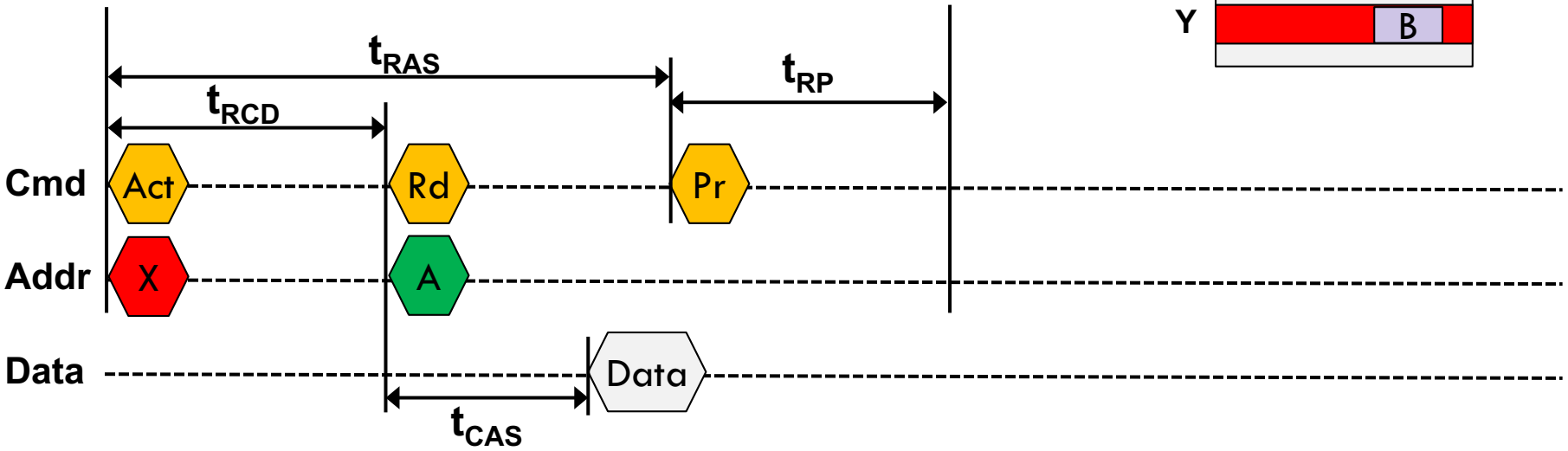




DRAM Timing Example

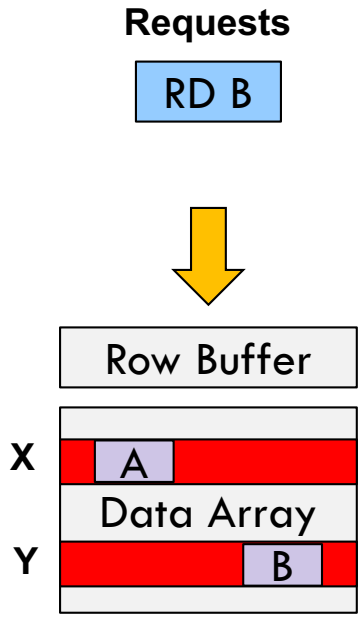
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

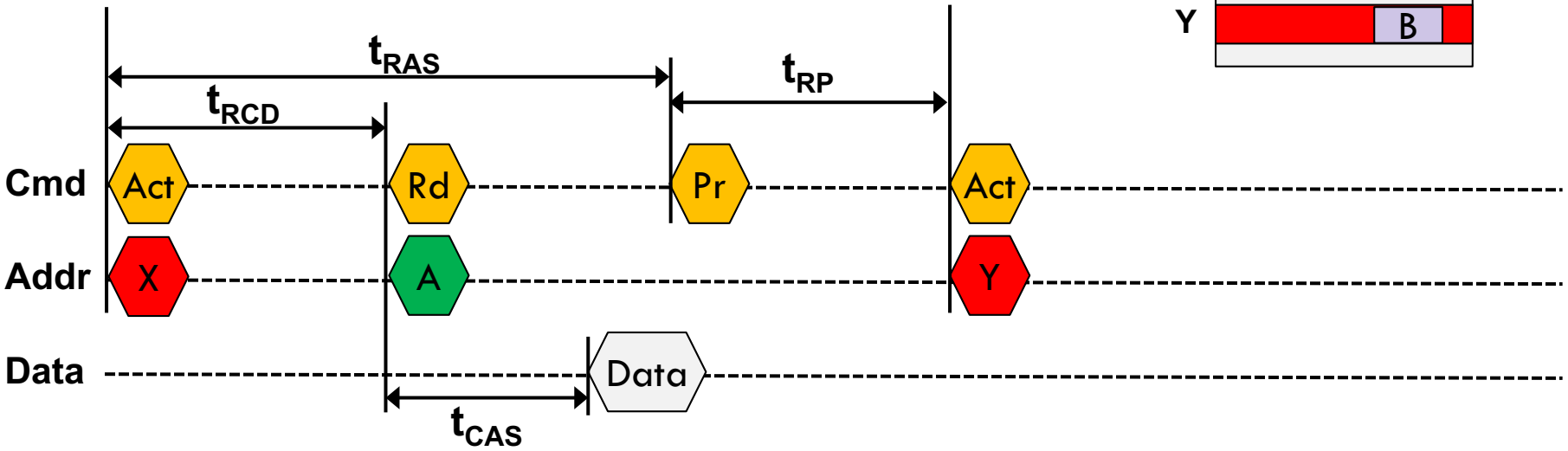




DRAM Timing Example

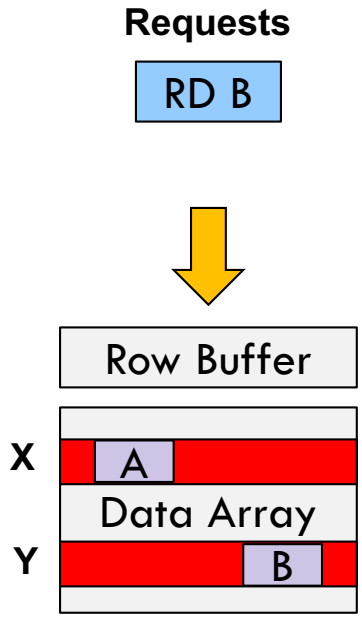
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

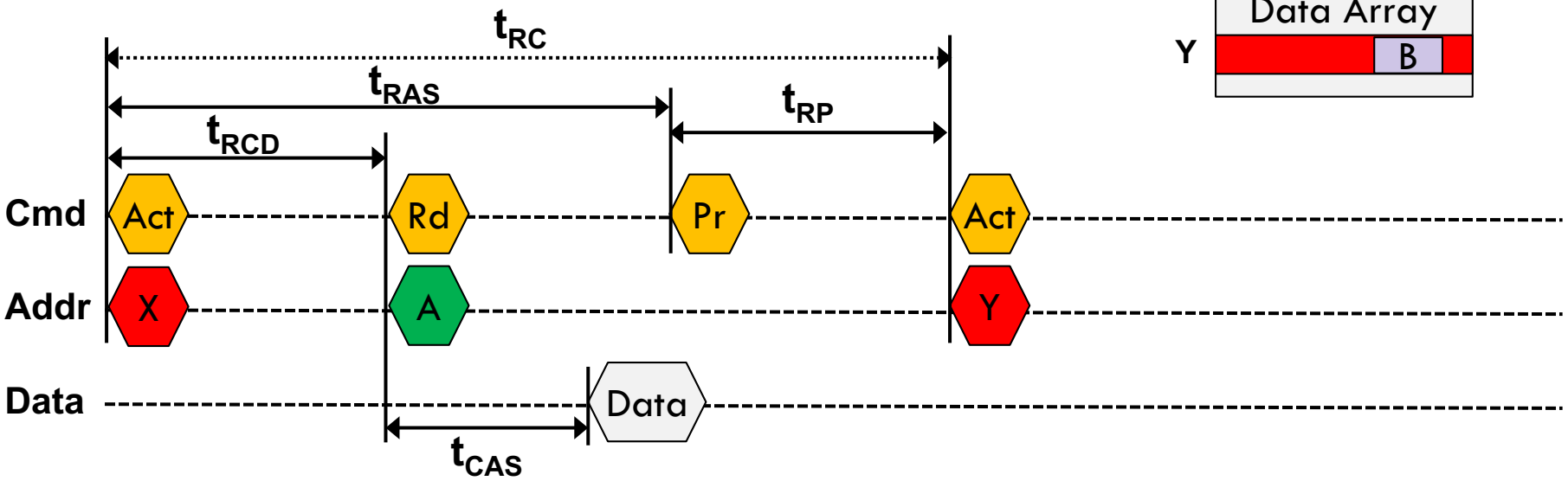




DRAM Timing Example

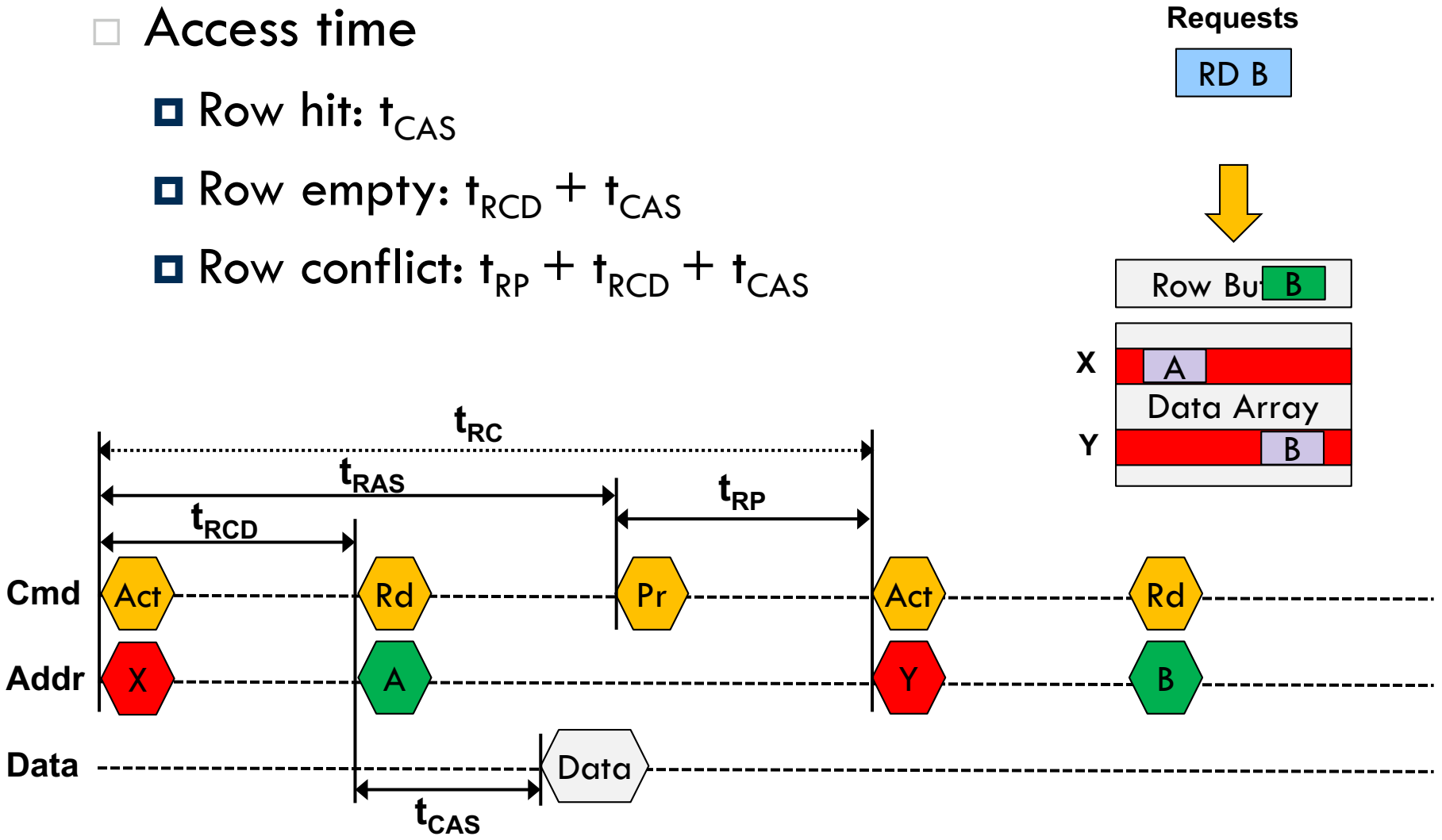
- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$





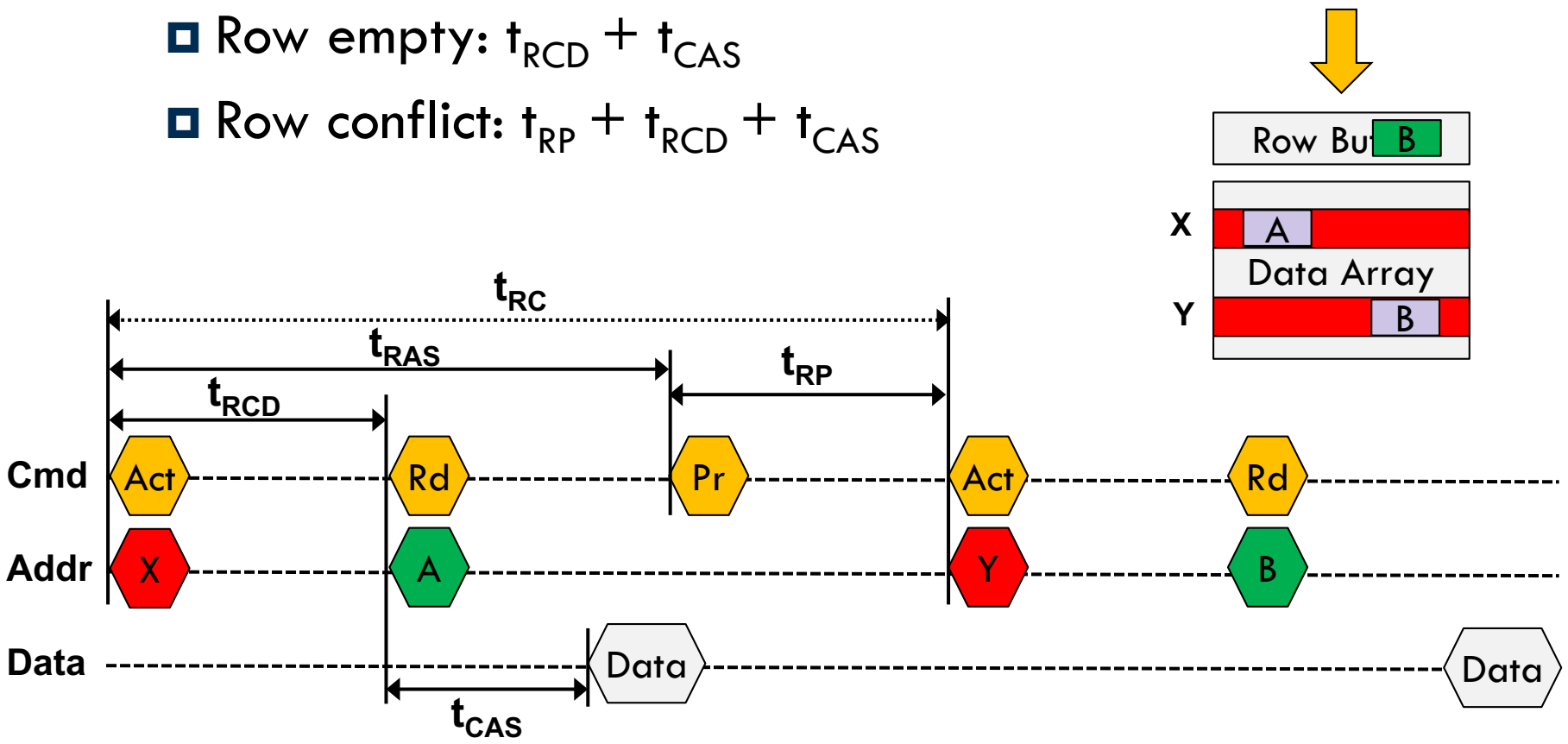
DRAM Timing Example

- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$



DRAM Timing Example

- Access time
  - Row hit: t_{CAS}
  - Row empty: $t_{RCD} + t_{CAS}$
  - Row conflict: $t_{RP} + t_{RCD} + t_{CAS}$

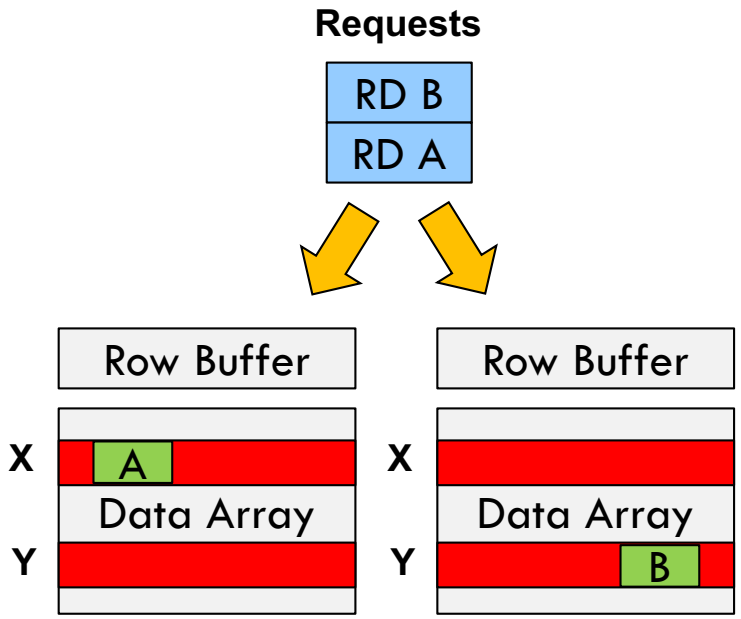


Improving Performance

DRAM Channels

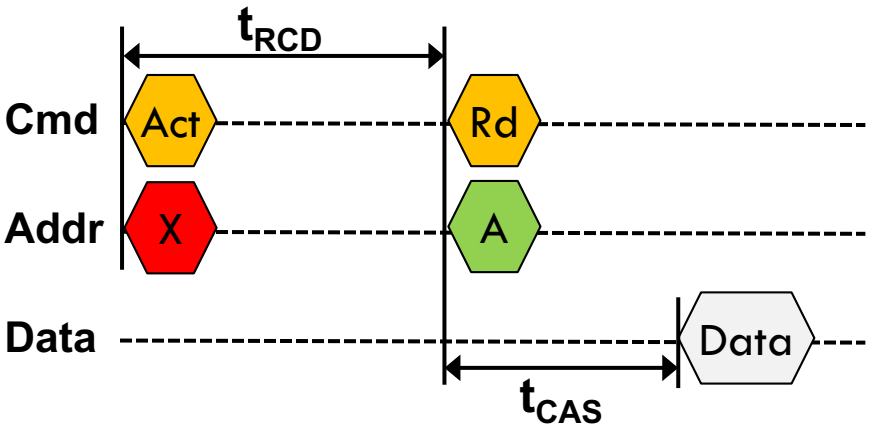
Memory Channels

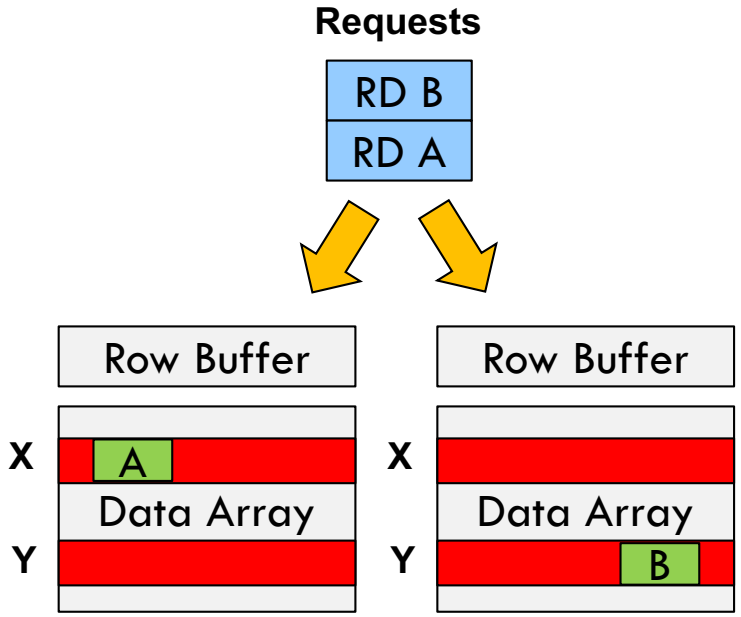
- Memory channels provide fully parallel accesses
  - Separate data, control, and address buses



Memory Channels

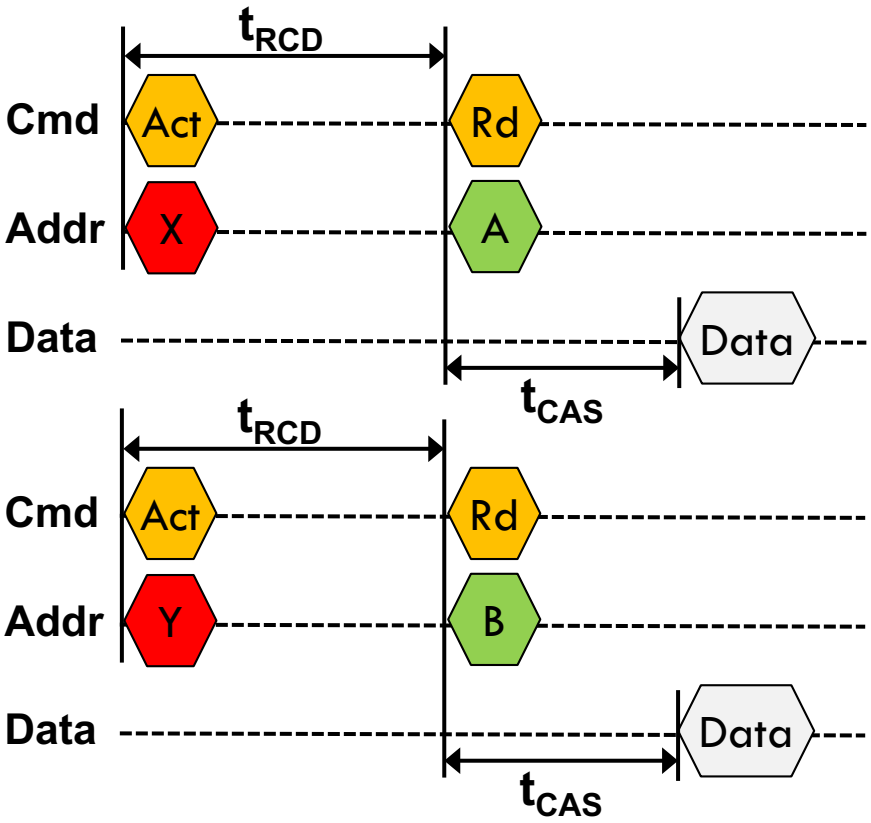
- Memory channels provide fully parallel accesses
  - Separate data, control, and address buses

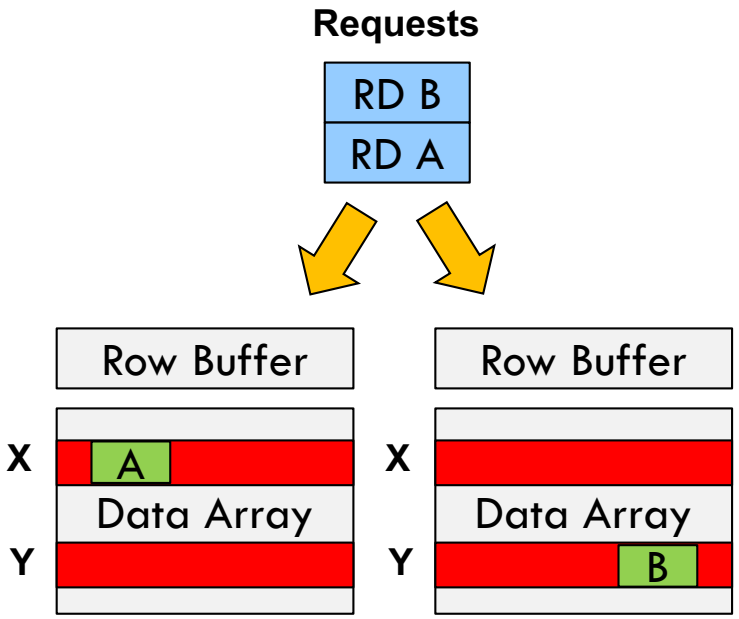




Memory Channels

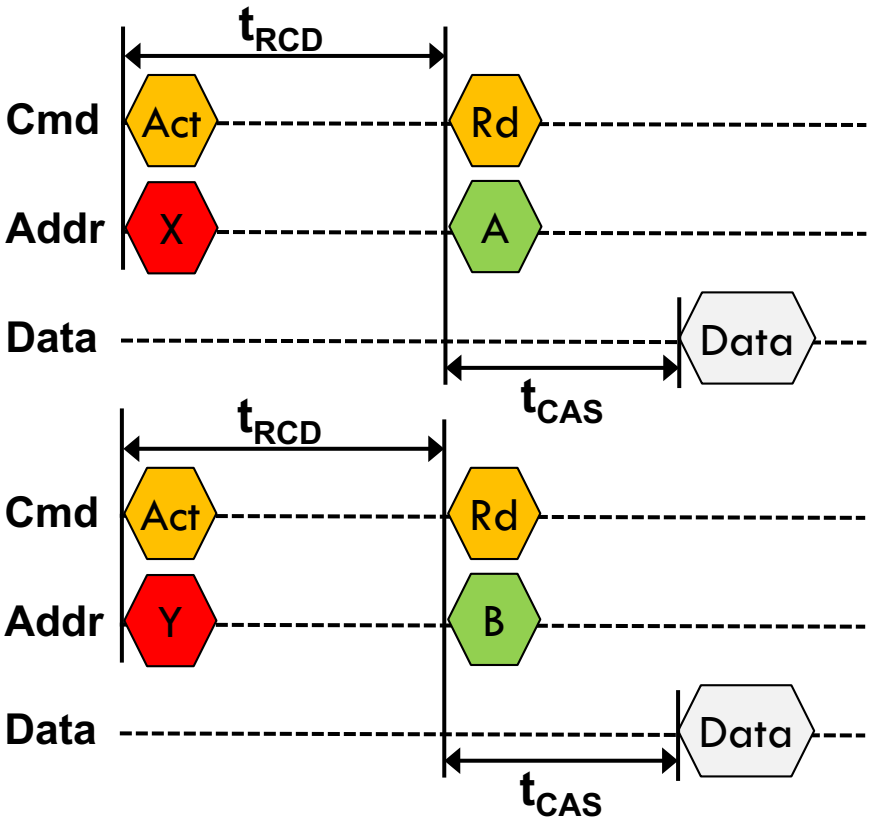
- Memory channels provide fully parallel accesses
  - Separate data, control, and address buses



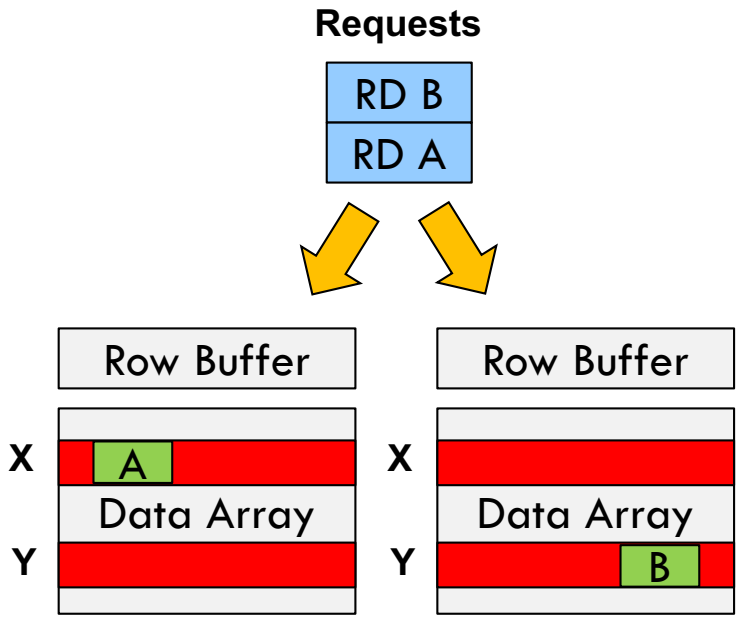


Memory Channels

- Memory channels provide fully parallel accesses
  - Separate data, control, and address buses



Not scalable due to pin overhead

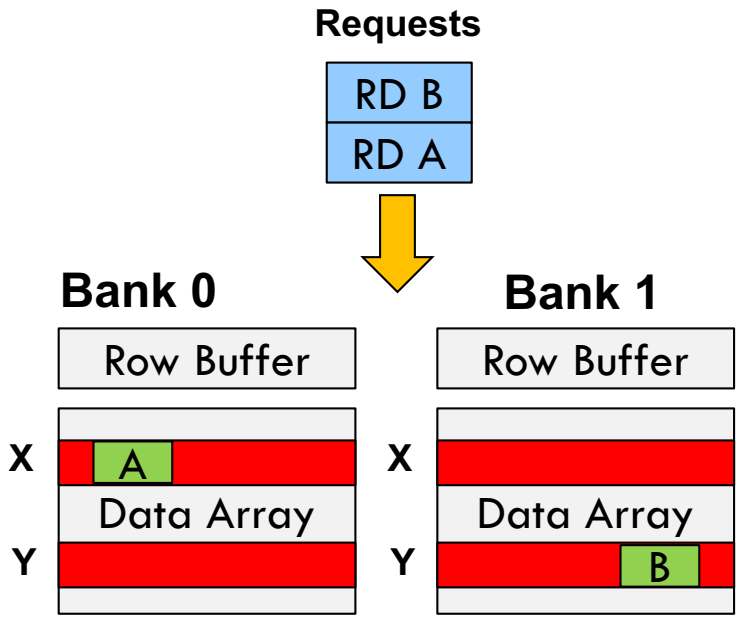


Improving Performance

DRAM Ranks

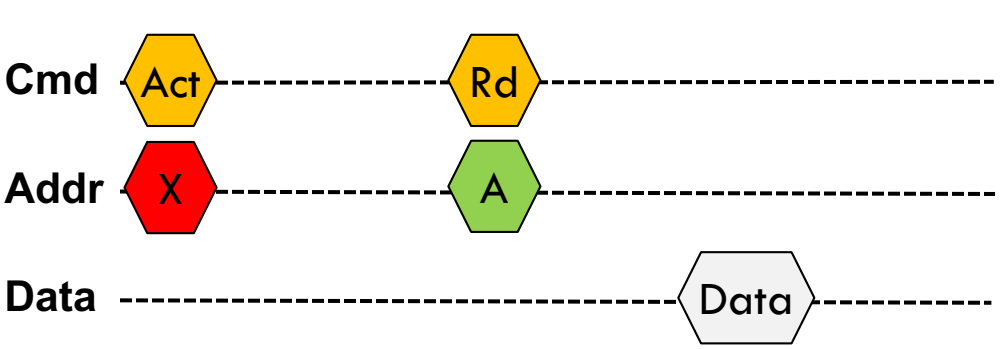
Memory Banks

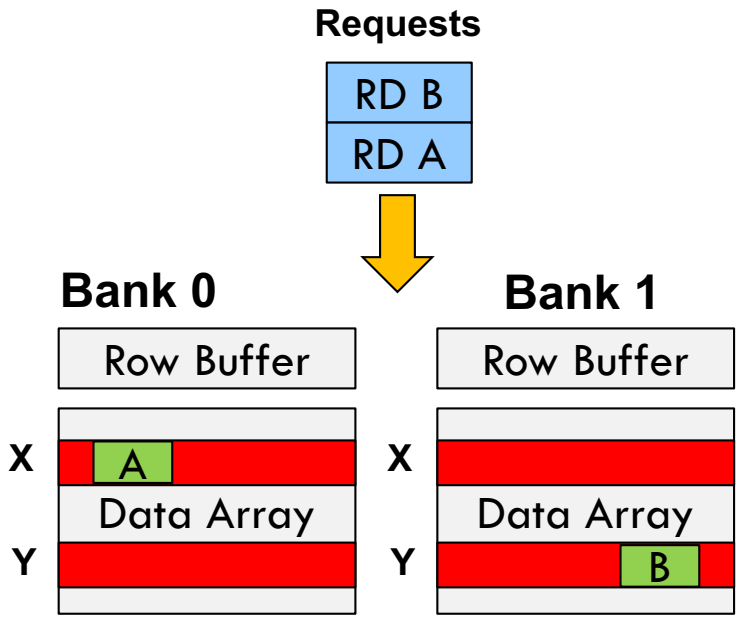
- Memory banks provide parallel operations
  - Shared data, control, and address buses
- The goal is to keep the data bus fully utilized



Memory Banks

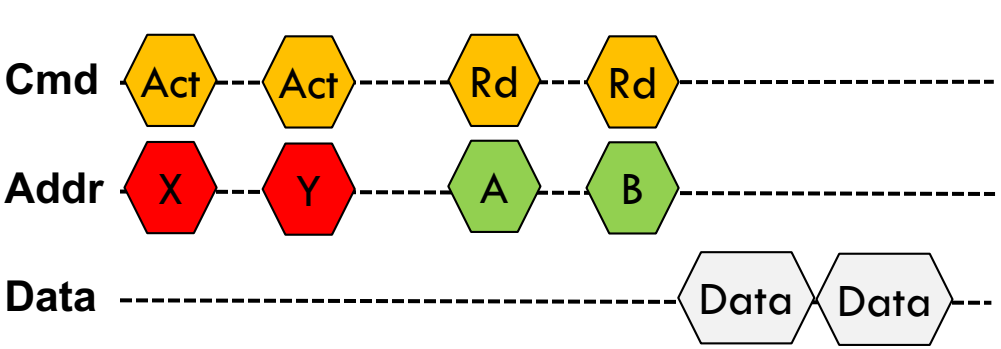
- Memory banks provide parallel operations
  - Shared data, control, and address buses
- The goal is to keep the data bus fully utilized

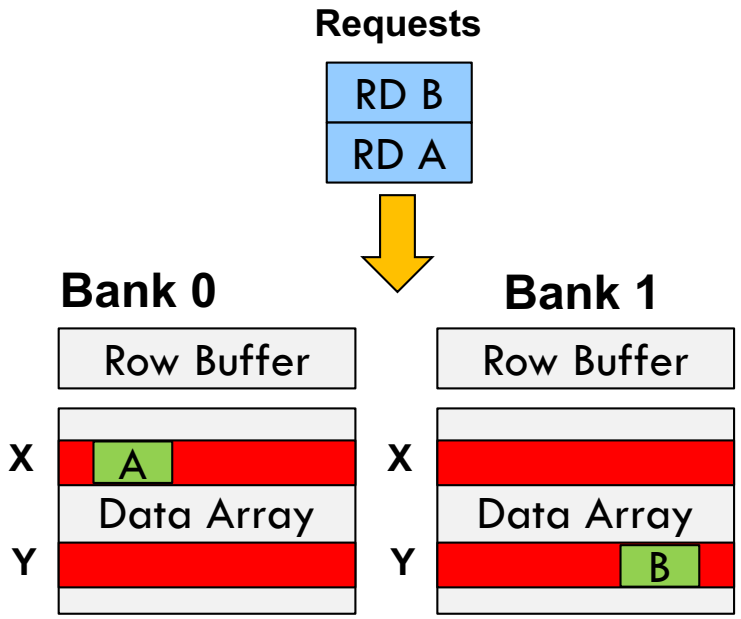




Memory Banks

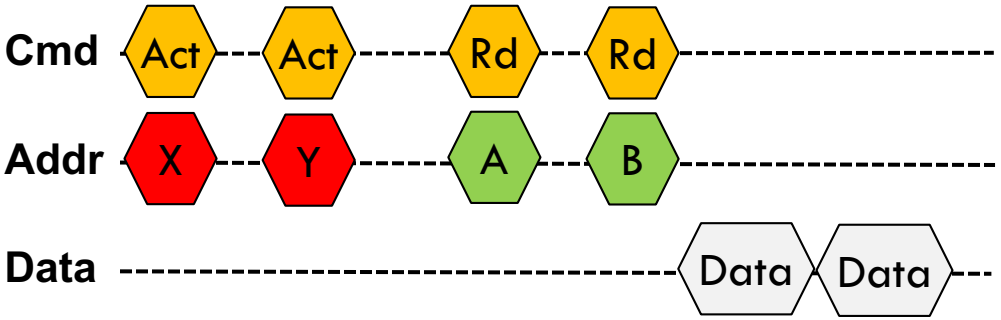
- Memory banks provide parallel operations
  - Shared data, control, and address buses
- The goal is to keep the data bus fully utilized

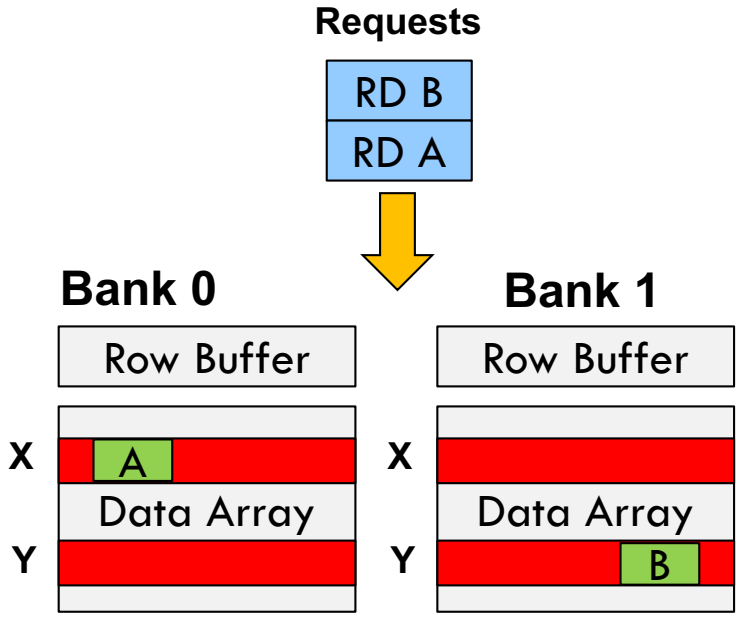




Memory Banks

- Memory banks provide parallel operations
  - Shared data, control, and address buses
- The goal is to keep the data bus fully utilized





Shorter data transfer time to reduce bus conflicts
Double data rate vs. single rate

DRAM Organization

- DRAM channels are independently accessed through dedicated data, address, and command buses
  - Physically broken down into DIMMs (dual in-line memory modules)
  - Logically divided into ranks, which are a collection of DRAM chips responding to the same memory request