Table of contents

1. Memory hierarchy
2. Cache memory
3. Main memory
4. Virtual memory
   - TLB
   - Page fault
   - IA-32 paging

Introduction

Second level of the memory hierarchy based on DRAM technology

- intermediate speed
- intermediate cost per bit $\Rightarrow$ intermediate capacity

The communication unit with other levels is the block

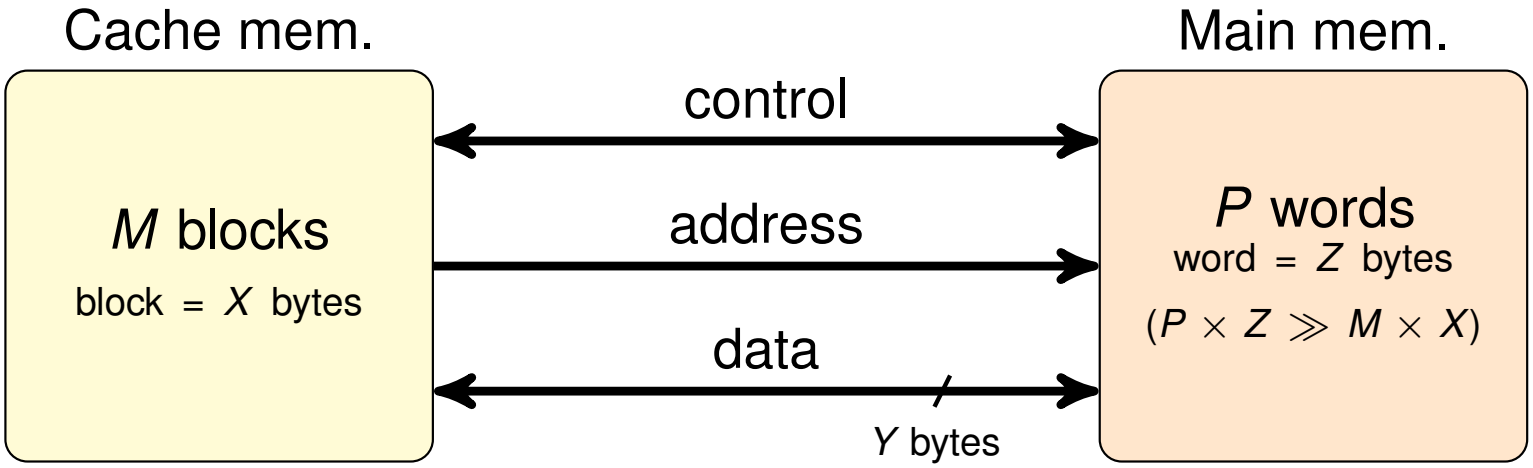
- consecutive words in memory
- different sizes depending on the level

Interaction with the cache level

- after a cache miss
- when updating a memory block

Communication with the cache memory

Cache miss or update operation $\Rightarrow$ block transferring

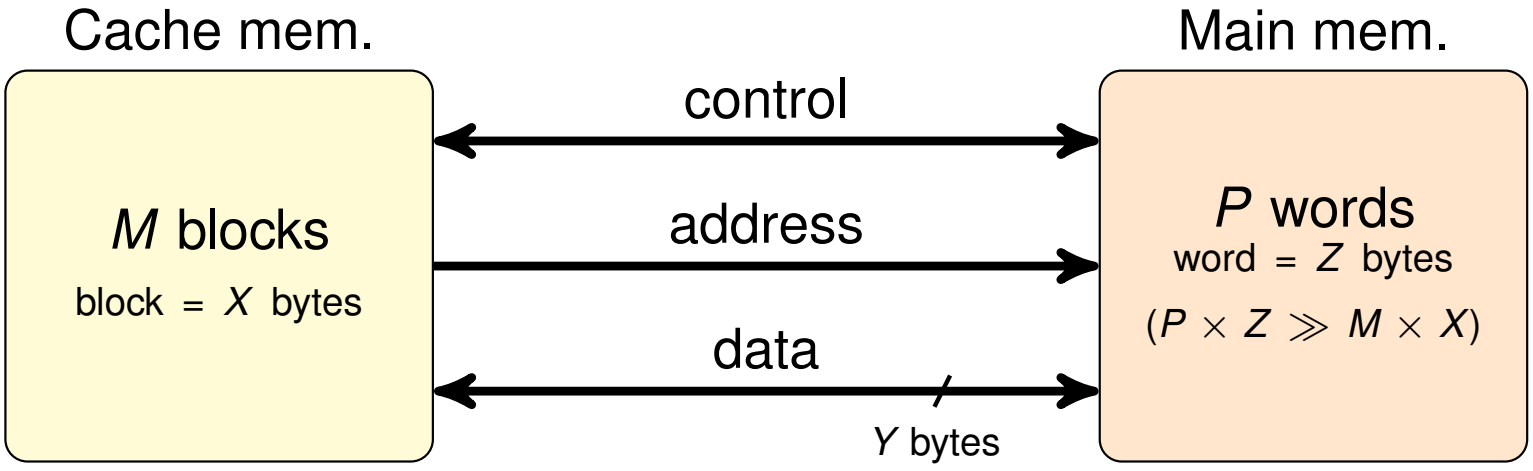


The number of memory accesses depends on

- Block size (X bytes)
- Memory word width (Z bytes)
- Data bus width (Y bytes)

Communication with the cache memory

Cache miss or update operation $\Rightarrow$ block transferring



The number of memory accesses depends on

- Block size (X bytes)
- Memory word width (Z bytes)
- Data bus width (Y bytes)

$$\left\} \frac{X}{\min(Y, Z)} \right.$$

Cost of a cache miss

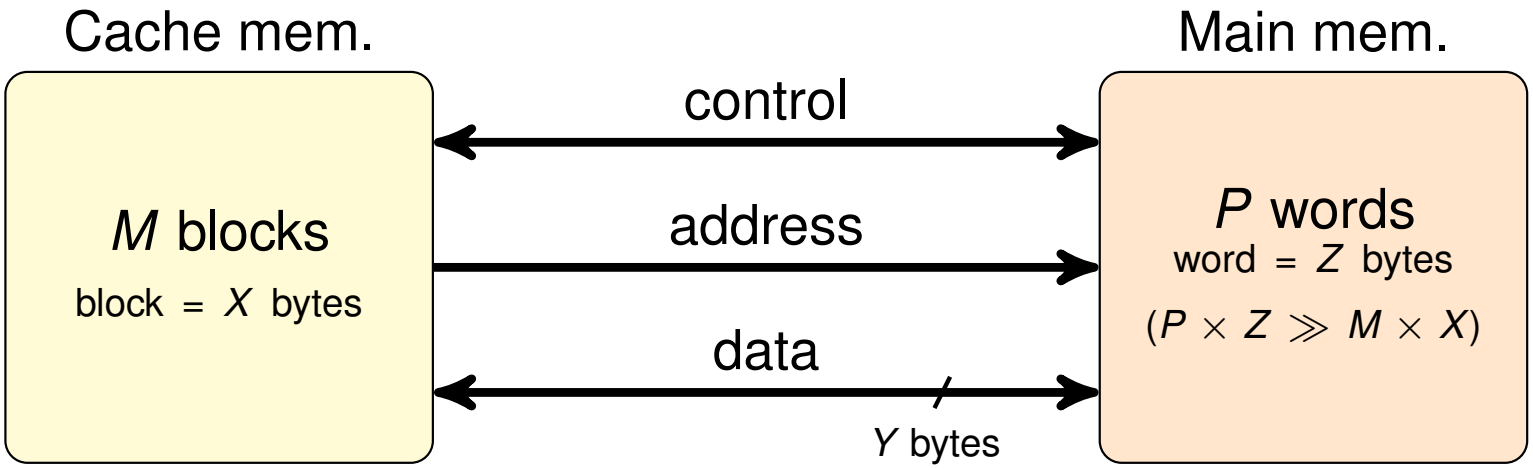
Transference: cache $\Leftarrow$ main memory

One or several bus cycles for reading

OP1 write the address

OP2 lookup the data item in memory

OP3 transfer Y bytes over the data bus

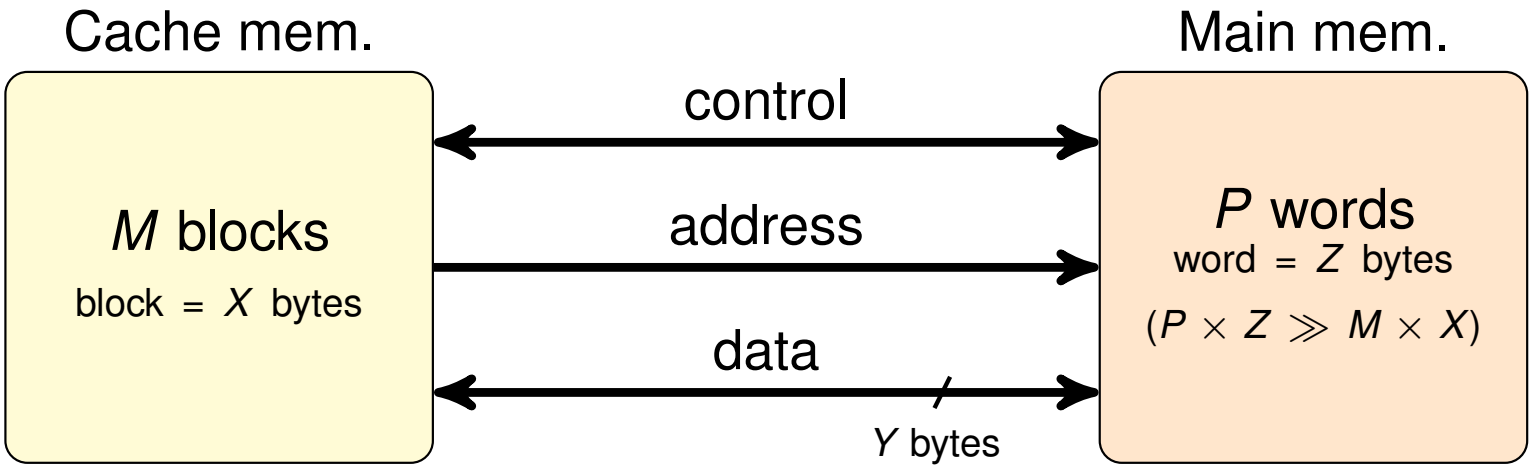


Cost of a cache miss

Transference: cache $\Leftarrow$ main memory

One or several bus cycles for reading

- OP1 write the address
- OP2 lookup the data item in memory
- OP3 transfer Y bytes over the data bus



Each operation requires several clock cycles

Cost of a cache miss

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes
- data bus width: 8 bytes
- memory word: 8 bytes

- OP1: 1 clock cycle
- OP2: 3 clock cycle
- OP3: 1 clock cycle

How many clock cycles are required to transfer a block?

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of a cache miss

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 3 clock cycle
- OP3: 1 clock cycle

How many clock cycles are required to transfer a block?

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of a cache miss

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 3 clock cycle
- OP3: 1 clock cycle

How many clock cycles are required to transfer a block?

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of a cache miss

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 3 clock cycle
- OP3: 1 clock cycle

How many clock cycles are required to transfer a block?

$$\underbrace{\frac{X}{\min(Y,Z)}}_{\text{\# of accesses}} \times \underbrace{(\text{OP1} + \text{OP2} + \text{OP3})}_{\text{clock cycles per access}} = \frac{32}{8} \times (1 + 3 + 1) = 20 \text{ clock cycles}$$

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of a cache miss

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 3 clock cycle
- OP3: 1 clock cycle

How many clock cycles are required to transfer a block?

$$\underbrace{\frac{X}{\min(Y,Z)}}_{\text{\# of accesses}} \times \underbrace{(\text{OP1} + \text{OP2} + \text{OP3})}_{\text{clock cycles per access}} = \frac{32}{8} \times (1 + 3 + 1) = 20 \text{ clock cycles}$$

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of a cache miss

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 3 clock cycle
- OP3: 1 clock cycle

How many clock cycles are required to transfer a block?

$$\underbrace{\frac{X}{\min(Y,Z)}}_{\text{\# of accesses}} \times \underbrace{(\text{OP1} + \text{OP2} + \text{OP3})}_{\text{clock cycles per access}} = \frac{32}{8} \times (1 + 3 + 1) = 20 \text{ clock cycles}$$

Assuming a clock frequency of 200 MHz, how long does it take?

$$\underbrace{20}_{\text{cycles}} \times \underbrace{\frac{1}{2 \cdot 10^8}}_{\text{period}} = 10^{-7} = 100\,\text{ns}$$

Cost of a cache miss

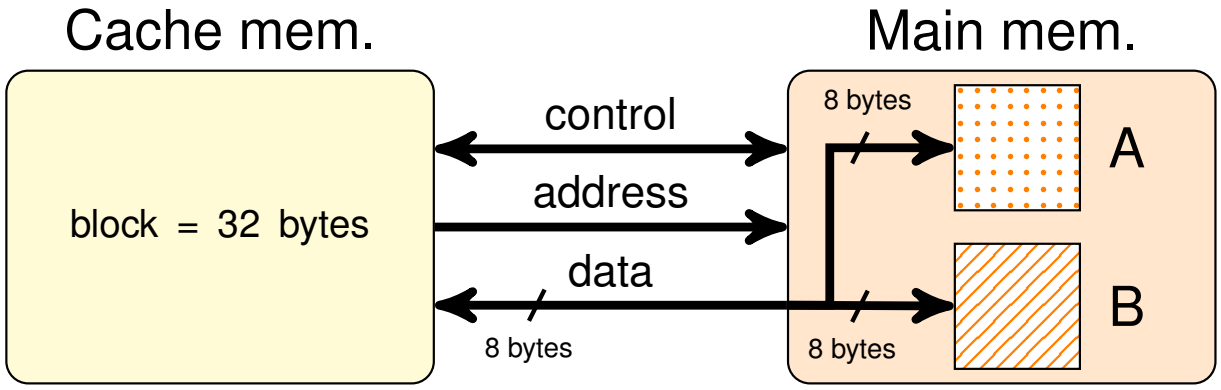
Example

- block size: 32 bytes
- data bus width: 8 bytes
- memory word: 8 bytes
- two memory banks: A y B

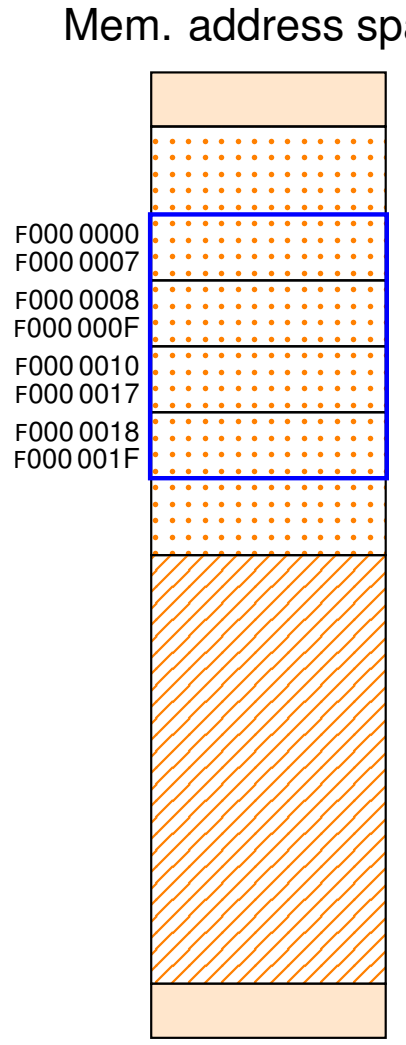
A cache miss occurs

- Transfer `F000 0000h-F000 001Fh` (32 bytes)

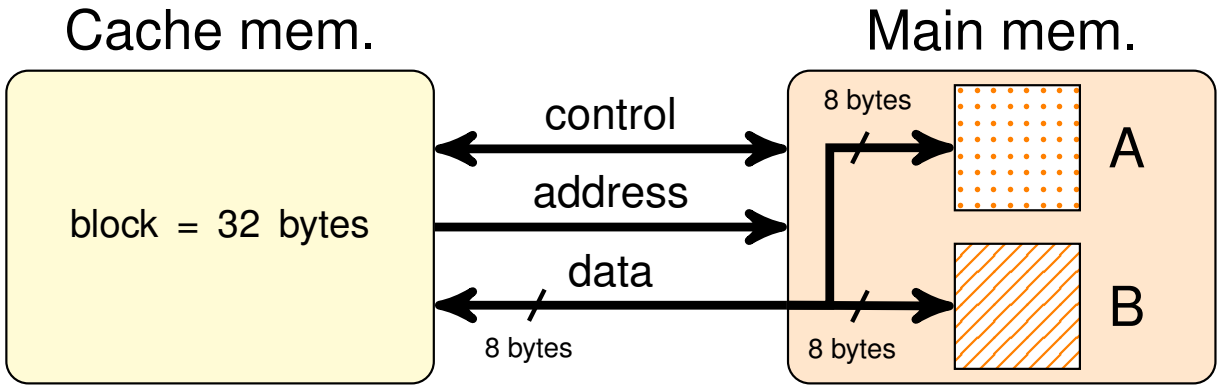
Cost of a cache miss



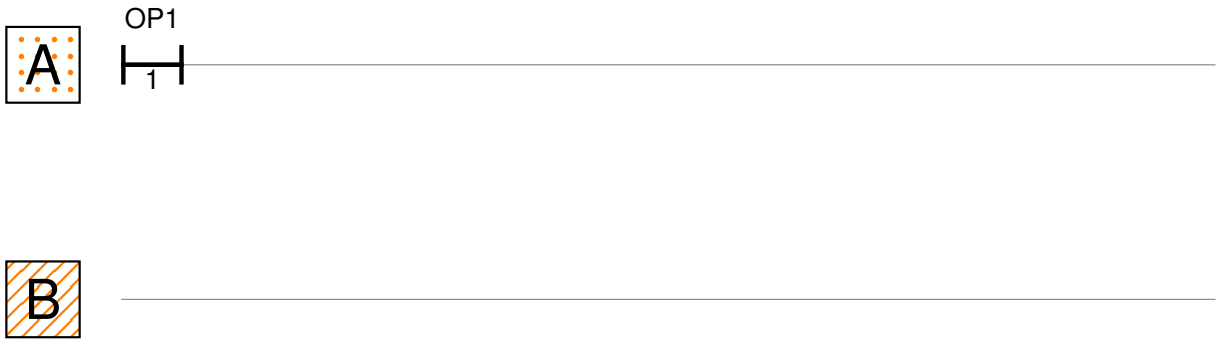
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

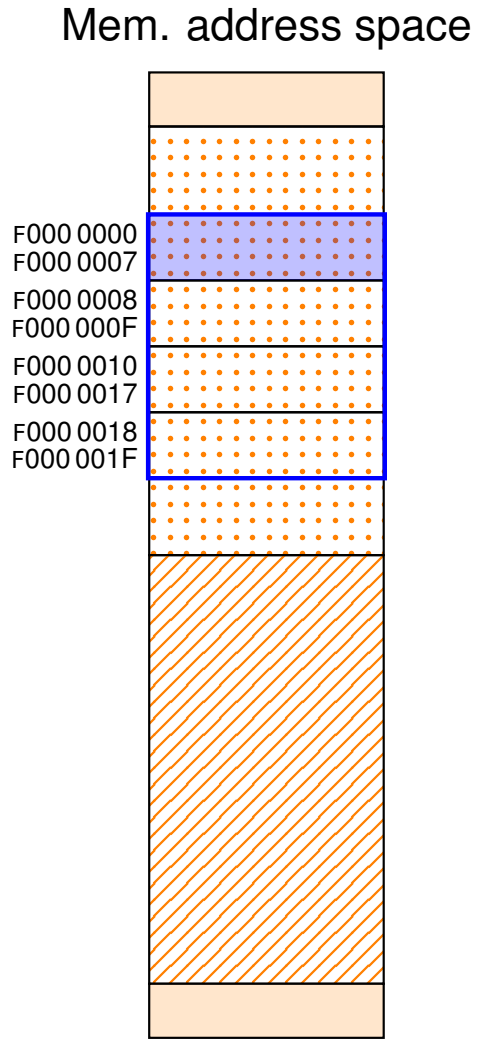


Cost of a cache miss

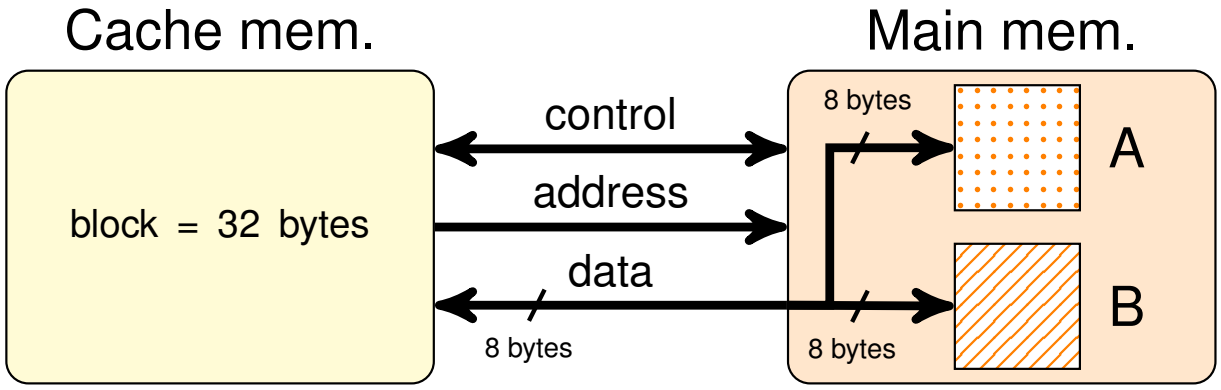


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

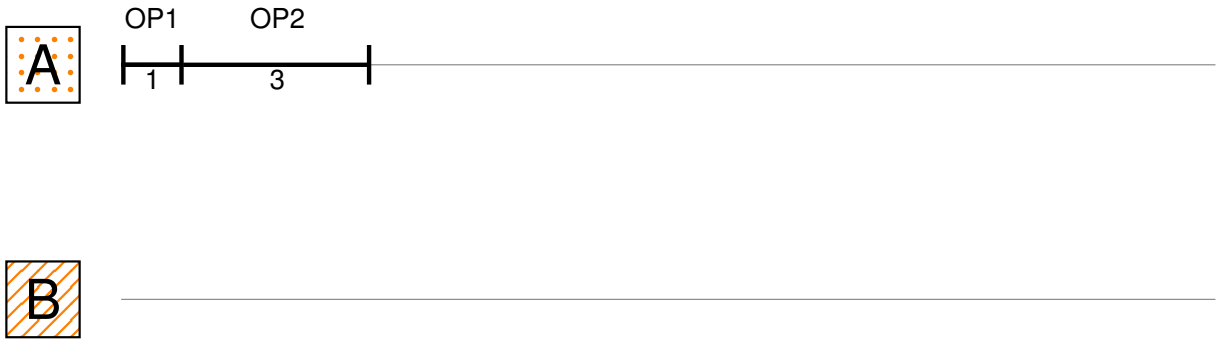


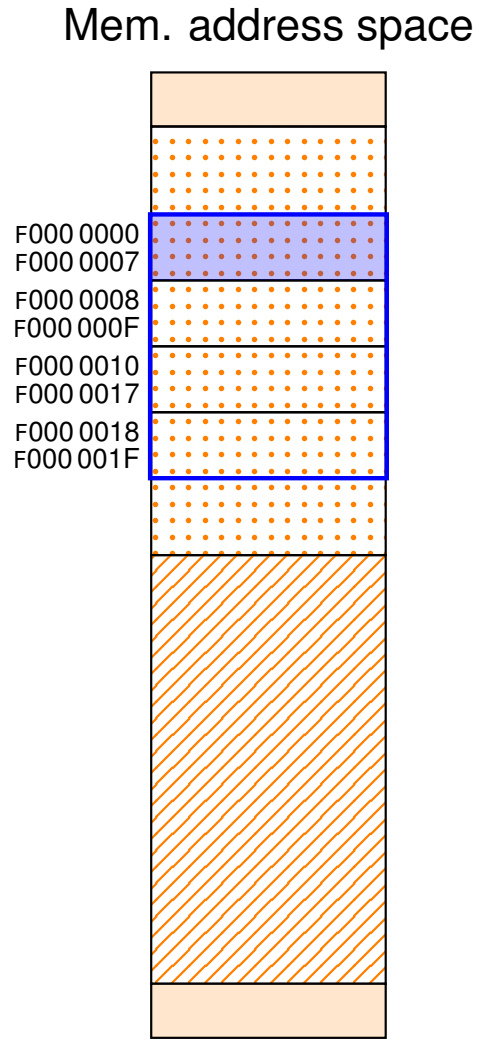


Cost of a cache miss

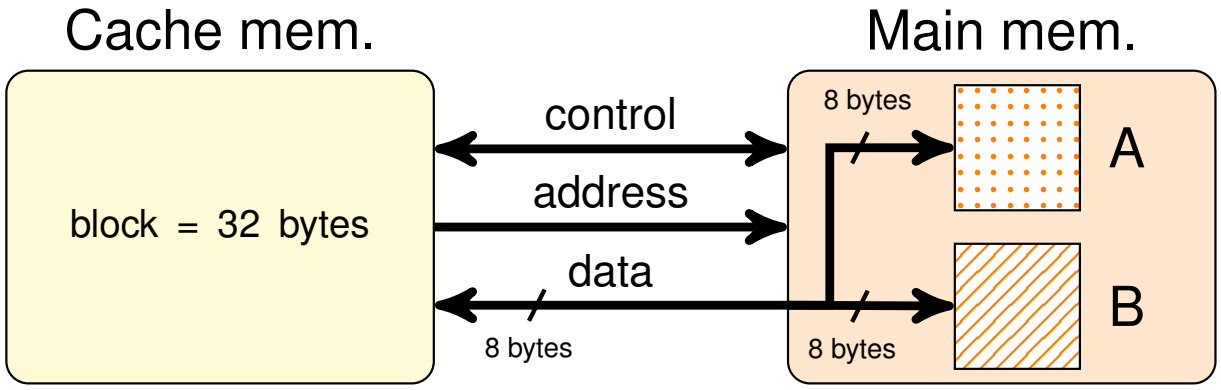


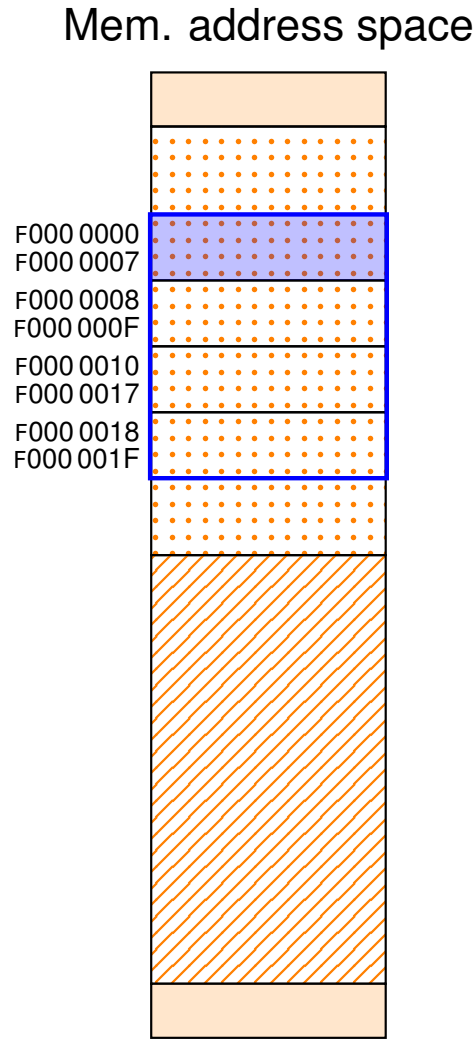
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle





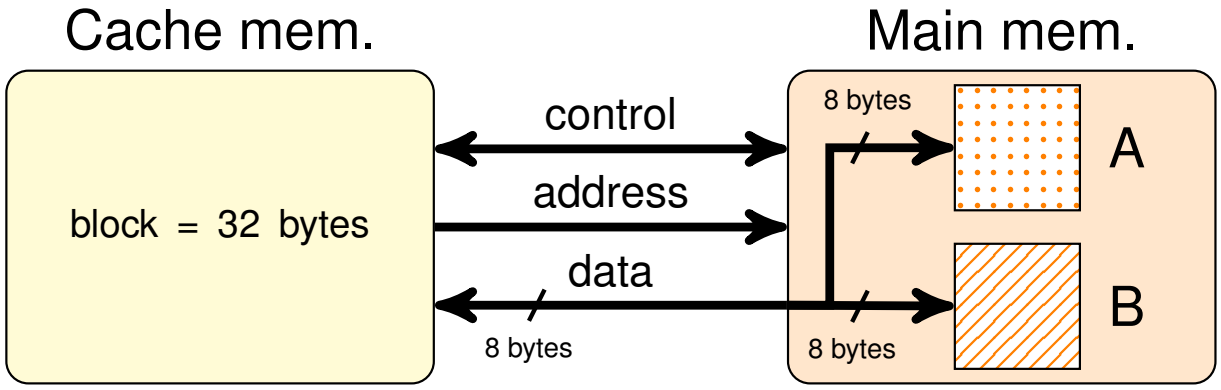
Cost of a cache miss

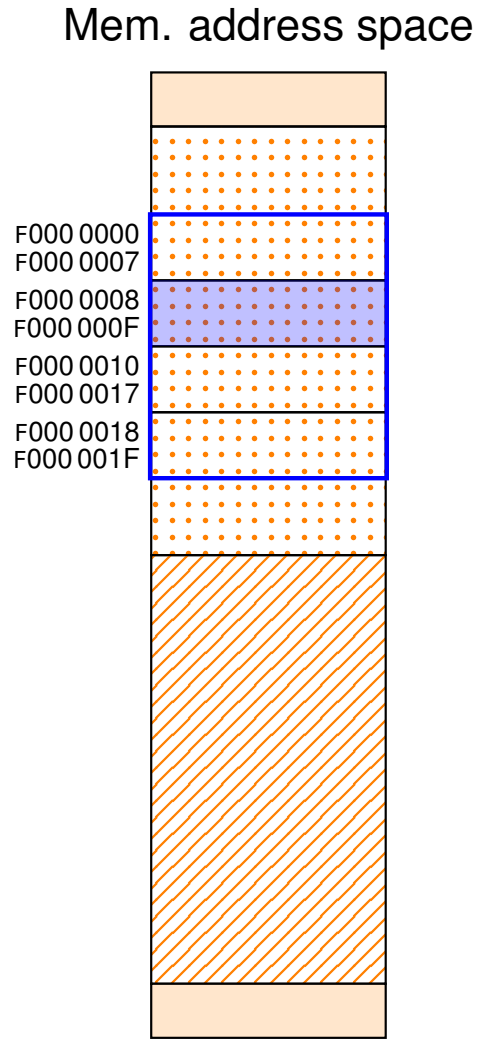




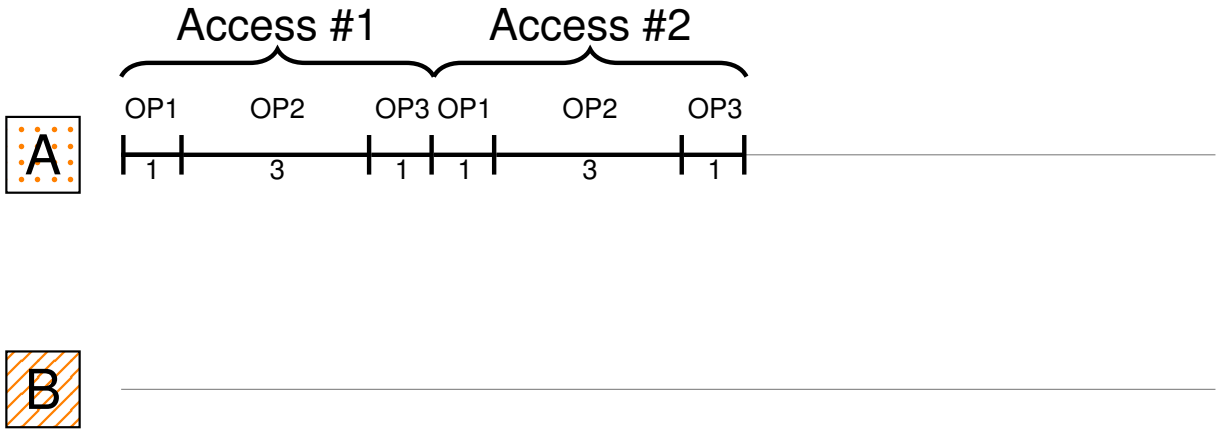
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

Cost of a cache miss

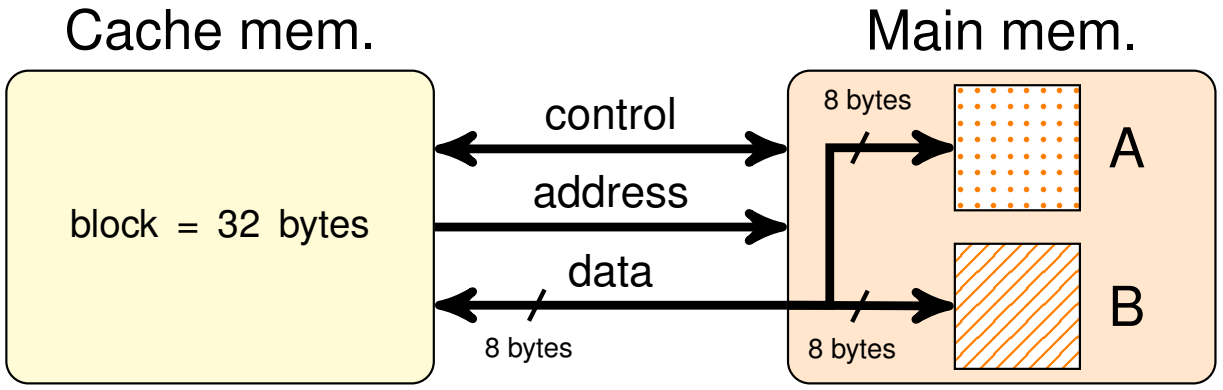


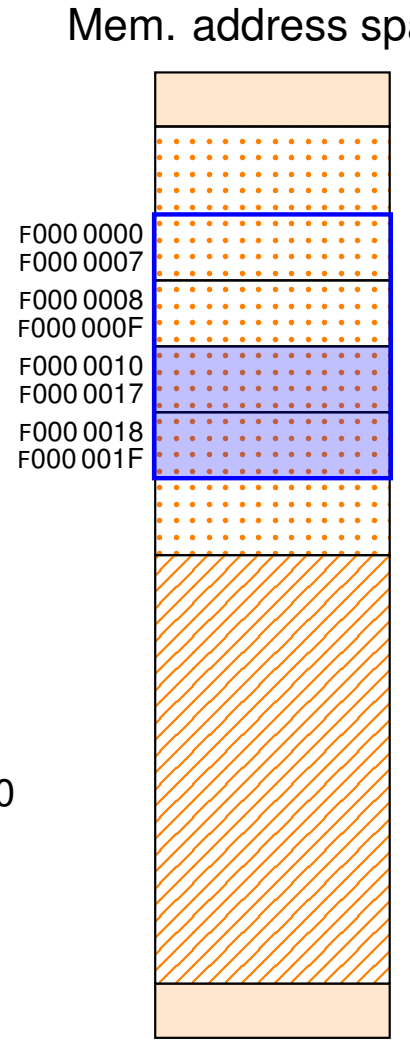


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

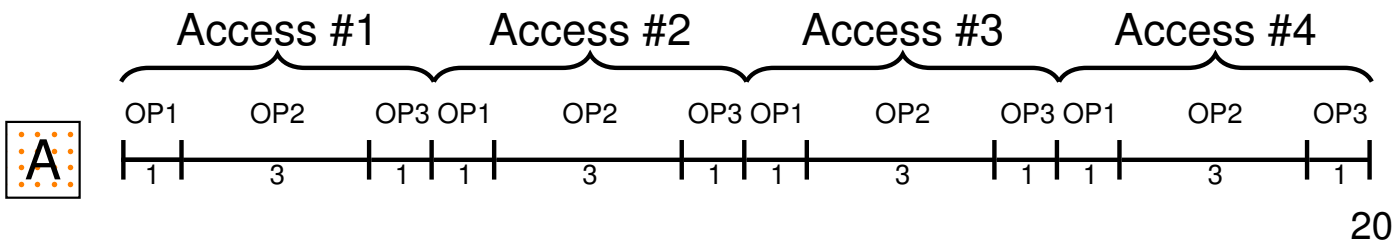


Cost of a cache miss

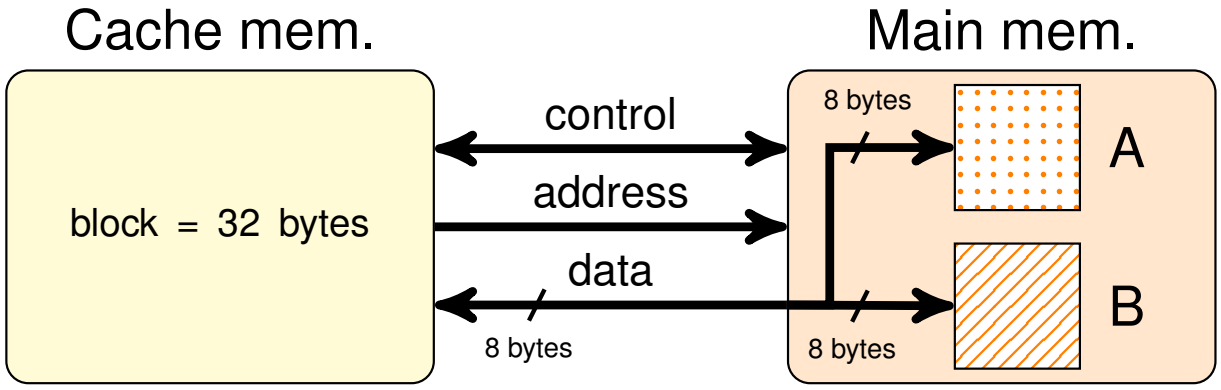


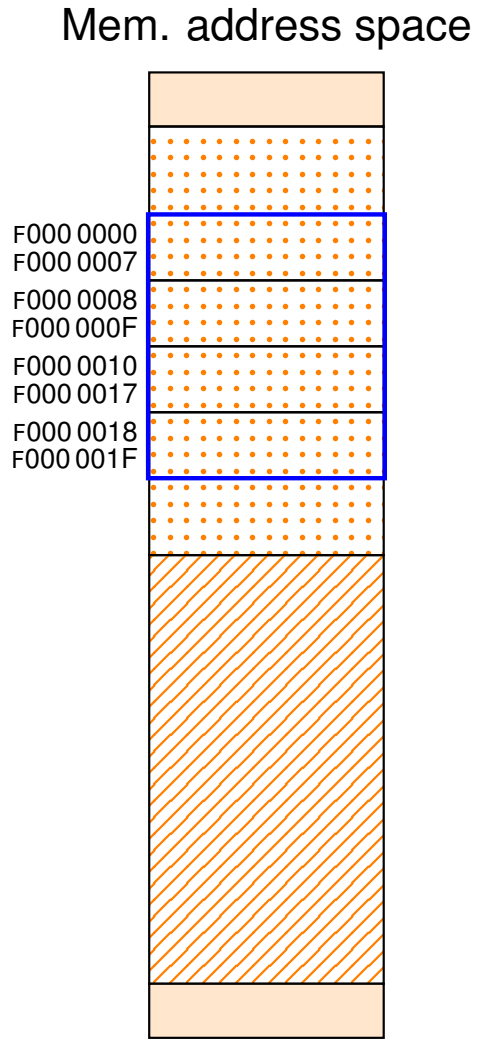


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle



Cost of a cache miss





OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

$$\text{Miss penalty} = \underbrace{(32/8)}_{\text{accesses}} \times \underbrace{(1+3+1)}_{\text{time/access}} = 20 \text{ clock cycles}$$

Cost of updating a block

Transference: cache $\Rightarrow$ main memory

- *write-through*: cache block is updated
- *write-back*
  - block is replaced
  - avoid incoherences

One or several bus cycles for writing

OP1 write the address

OP2 transfer Y bytes over the data bus

OP3 write the data item in memory

Cost of updating a block

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 1 clock cycle
- OP3: 4 clock cycle

How many clock cycles are required to update a block?

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of updating a block

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 1 clock cycle
- OP3: 4 clock cycle

How many clock cycles are required to update a block?

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of updating a block

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 1 clock cycle
- OP3: 4 clock cycle

How many clock cycles are required to update a block?

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of updating a block

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 1 clock cycle
- OP3: 4 clock cycle

How many clock cycles are required to update a block?

$$\underbrace{\frac{X}{\min(Y,Z)}}_{\text{\# of accesses}} \times \underbrace{(\text{OP1} + \text{OP2} + \text{OP3})}_{\text{cycles per access}} = \frac{32}{8} \times (1 + 1 + 4) = 24 \text{ clock cycles}$$

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of updating a block

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 1 clock cycle
- OP3: 4 clock cycle

How many clock cycles are required to update a block?

$$\underbrace{\frac{X}{\min(Y,Z)}}_{\text{\# of accesses}} \times \underbrace{(\text{OP1} + \text{OP2} + \text{OP3})}_{\text{cycles per access}} = \frac{32}{8} \times (1 + 1 + 4) = 24 \text{ clock cycles}$$

Assuming a clock frequency of 200 MHz, how long does it take?

Cost of updating a block

Example

Memory hierarchy with cache and main memory

- block size: 32 bytes (X)
- data bus width: 8 bytes (Y)
- memory word: 8 bytes (Z)

- OP1: 1 clock cycle
- OP2: 1 clock cycle
- OP3: 4 clock cycle

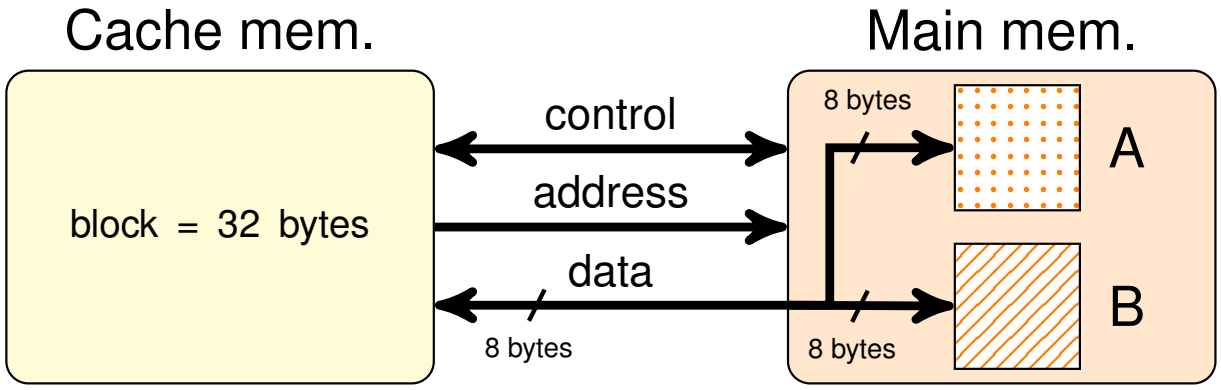
How many clock cycles are required to update a block?

$$\underbrace{\frac{X}{\min(Y,Z)}}_{\text{\# of accesses}} \times \underbrace{(\text{OP1} + \text{OP2} + \text{OP3})}_{\text{cycles per access}} = \frac{32}{8} \times (1 + 1 + 4) = 24 \text{ clock cycles}$$

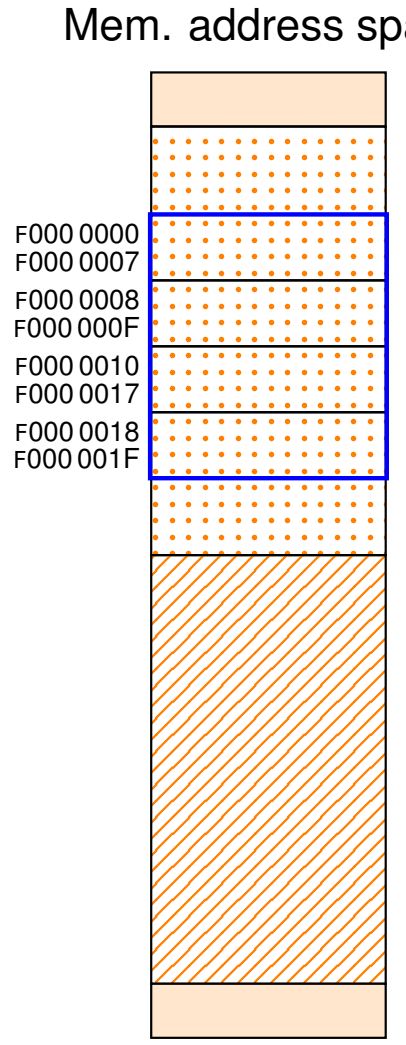
Assuming a clock frequency of 200 MHz, how long does it take?

$$\underbrace{24}_{\text{clock cycles}} \times \underbrace{\frac{1}{2 \cdot 10^8}}_{\text{period}} = 1.2 \cdot 10^{-7} = 120\,\text{ns}$$

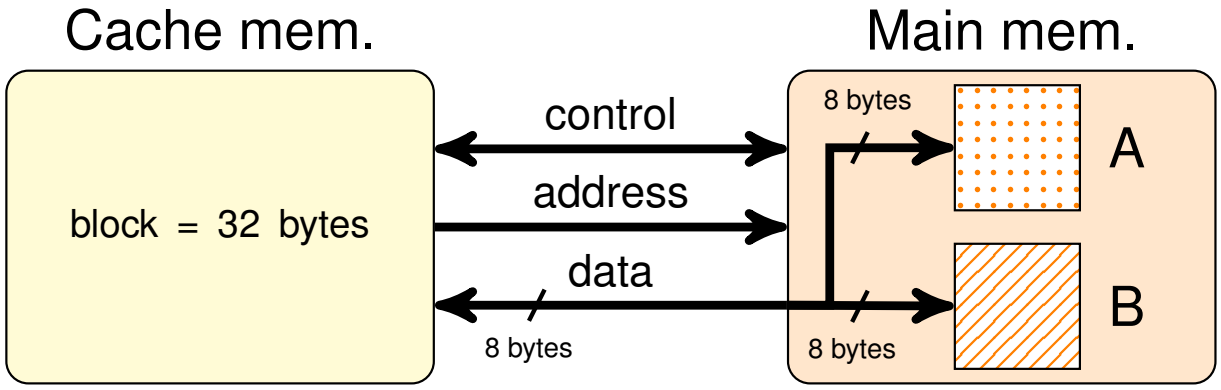
Cost of updating a block



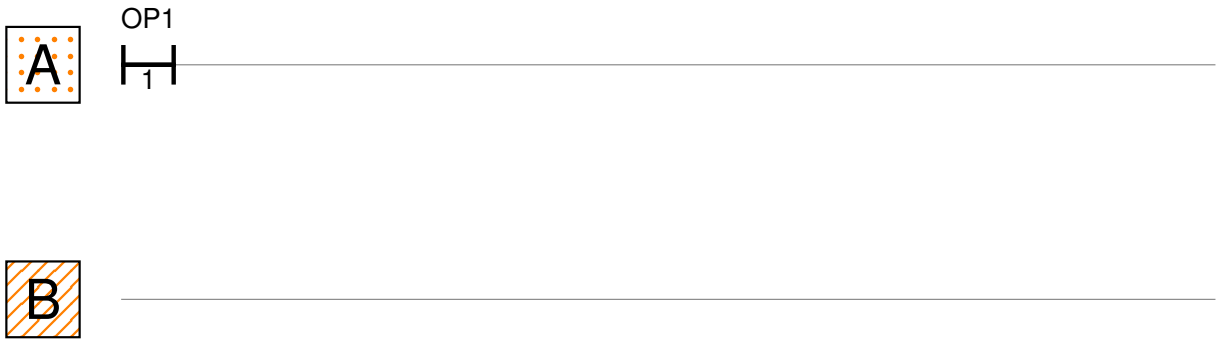
OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles

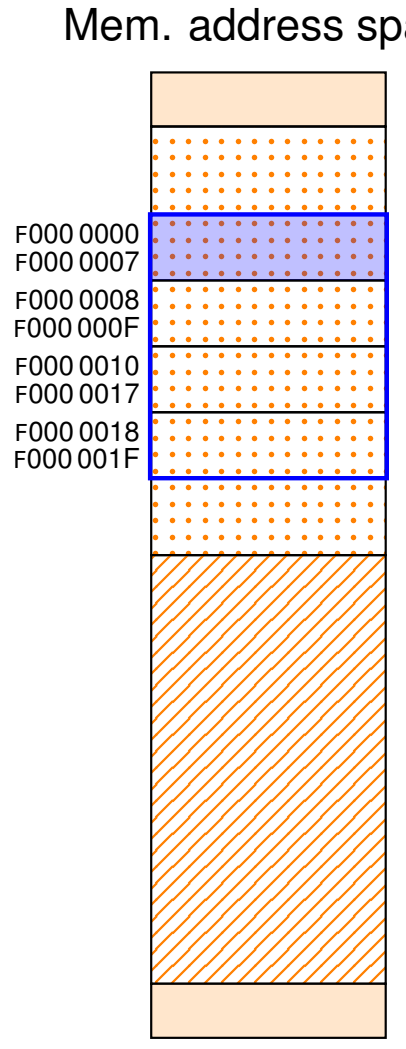


Cost of updating a block

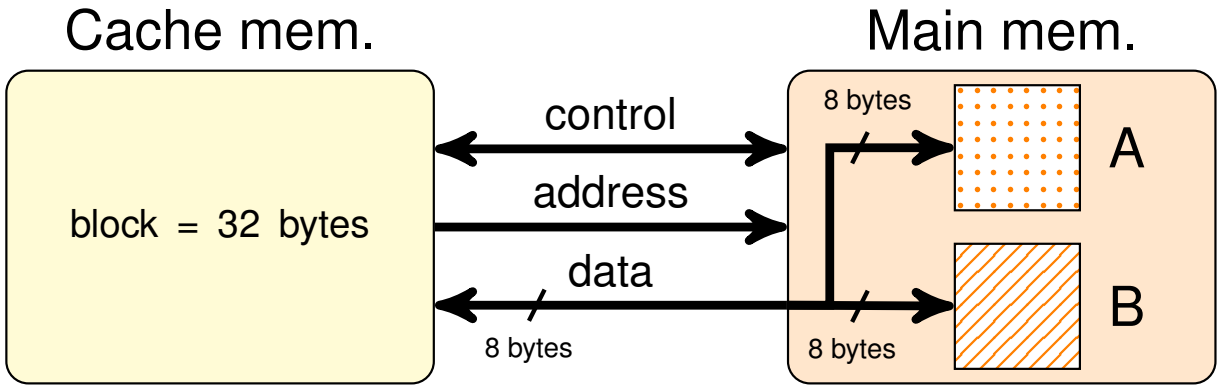


OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles

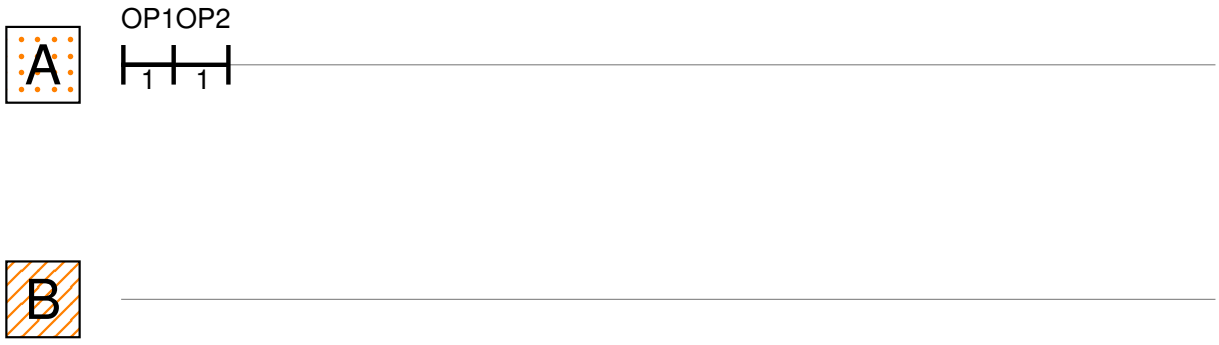


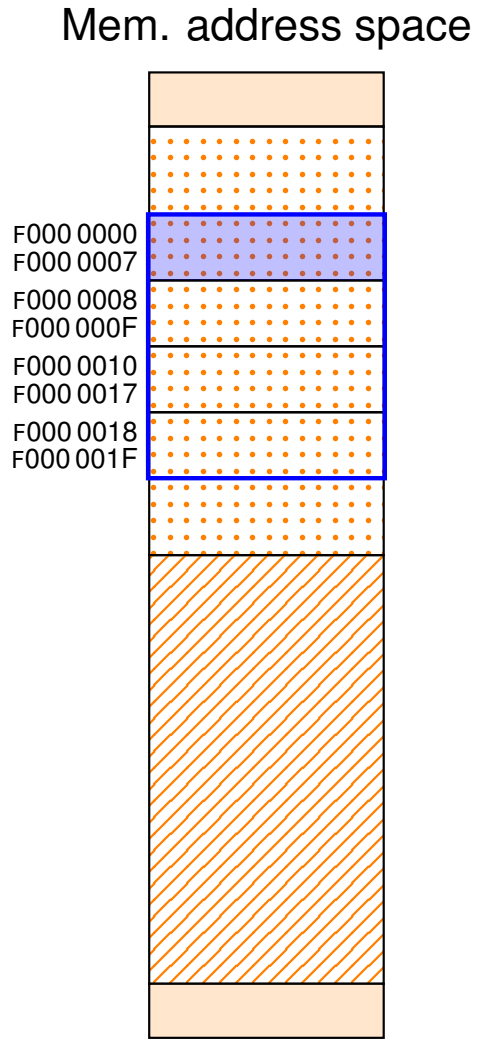


Cost of updating a block

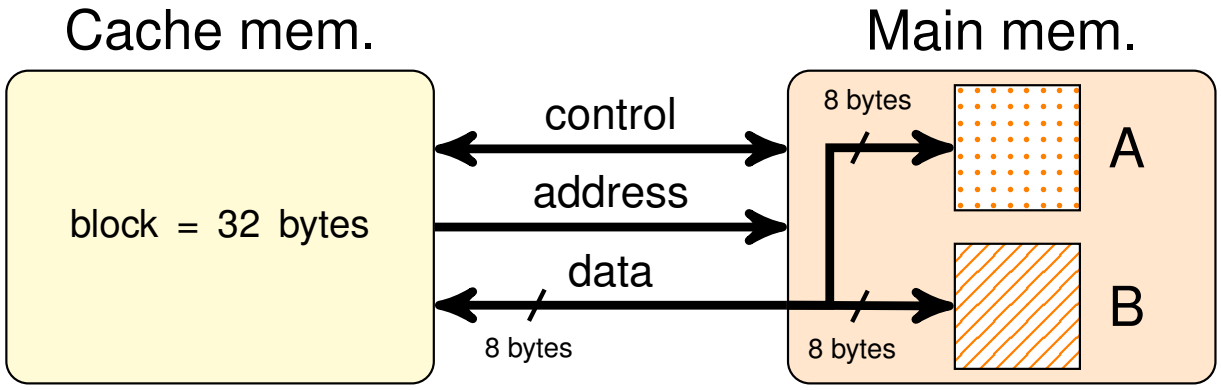


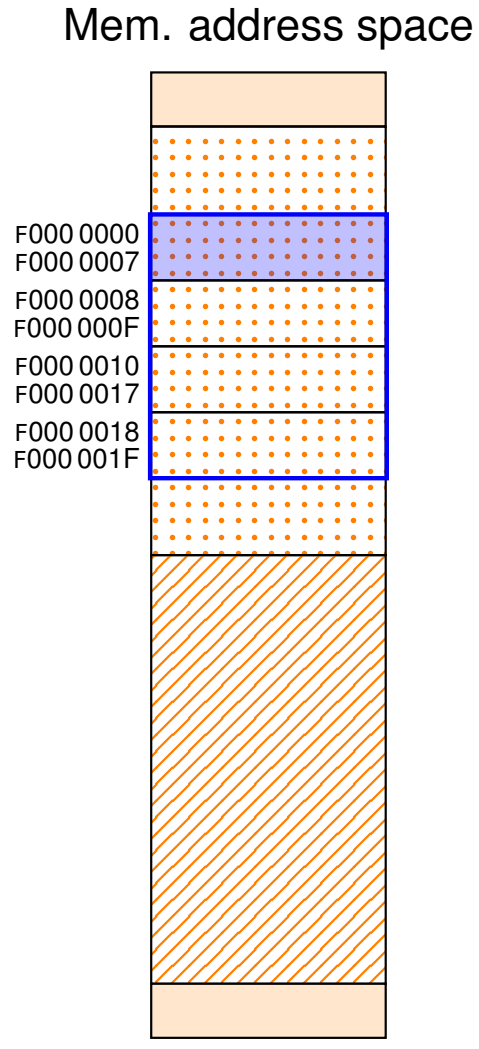
OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles



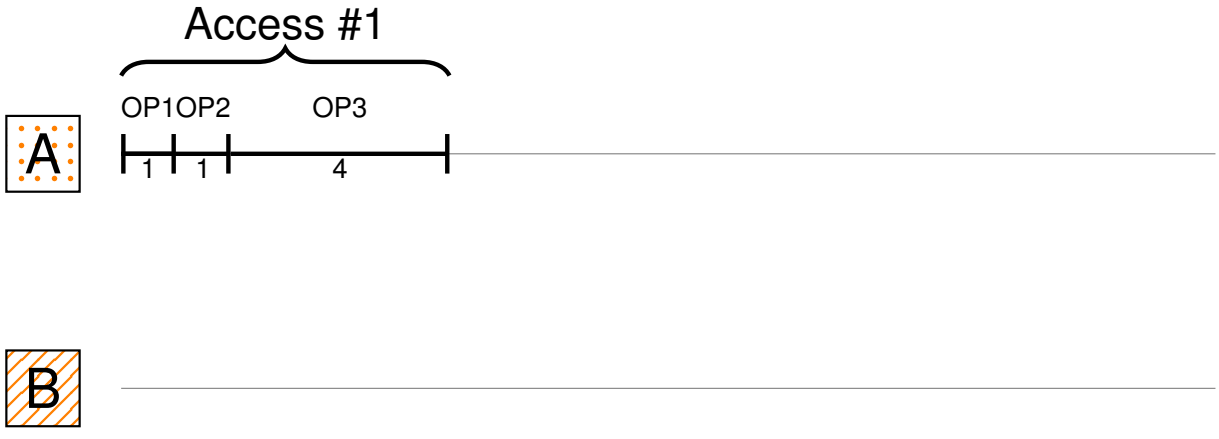


Cost of updating a block

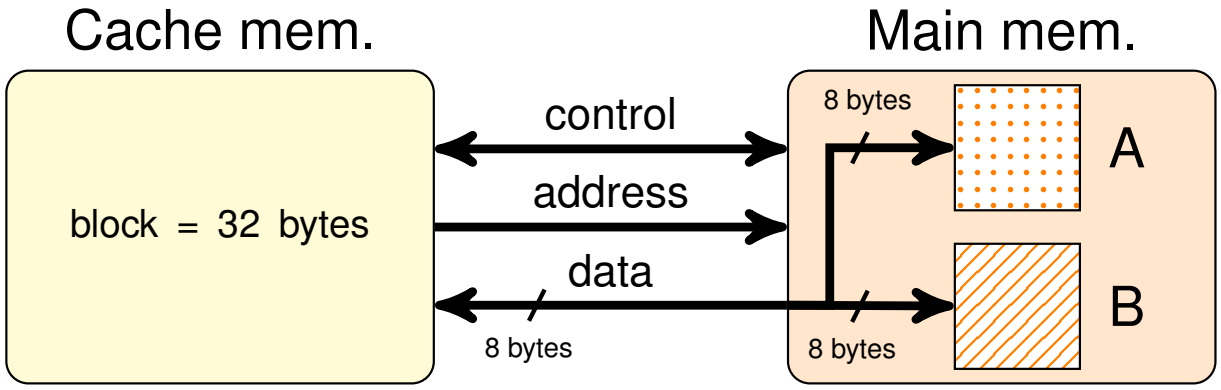


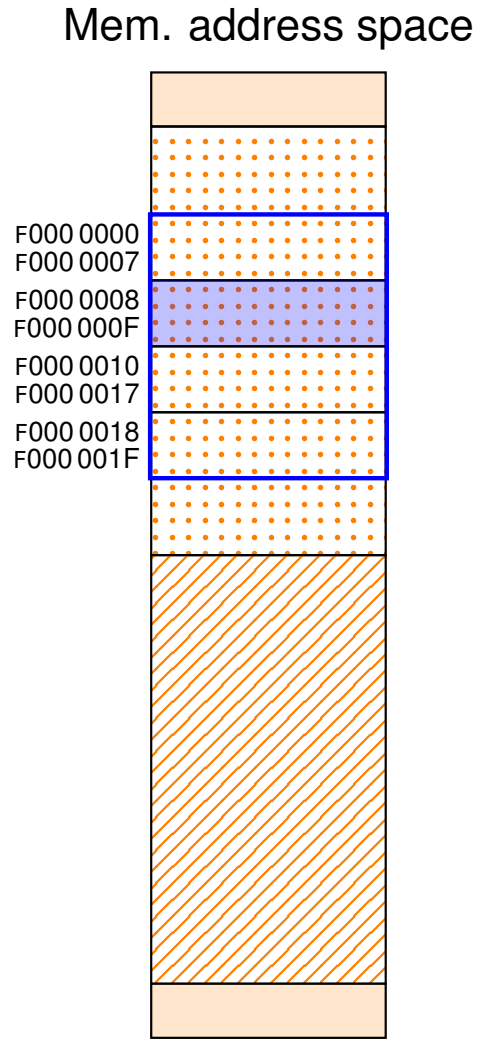


OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles

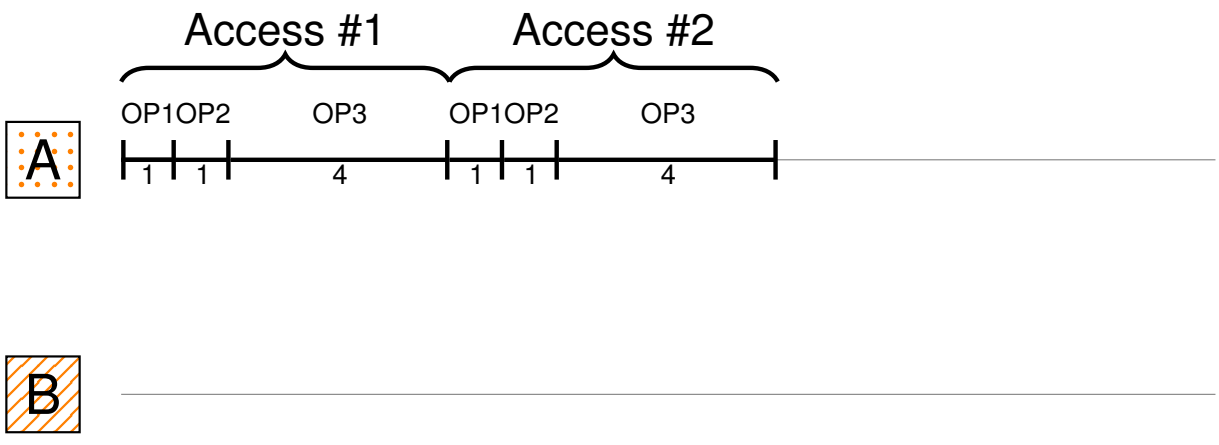


Cost of updating a block

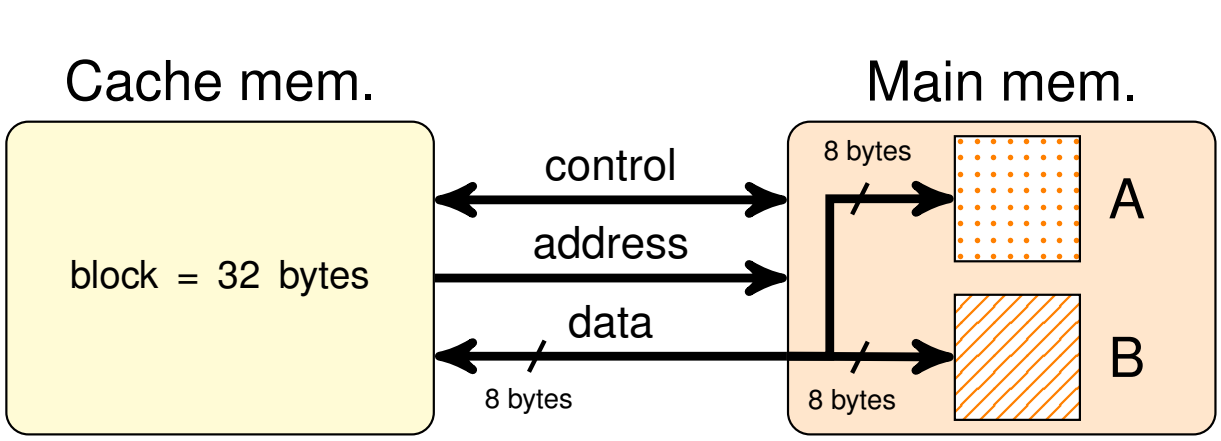


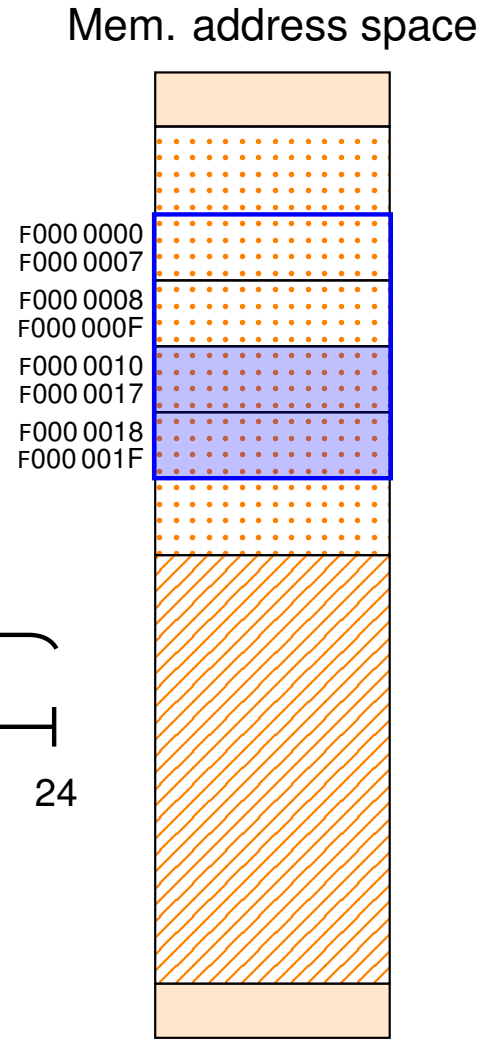


OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles

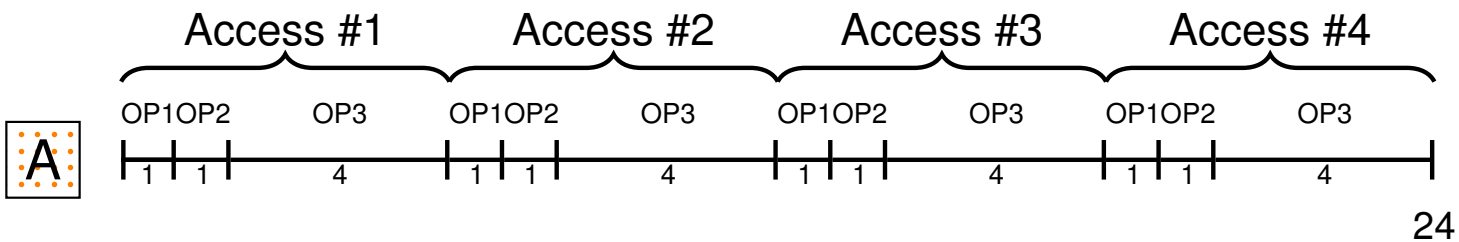


Cost of updating a block



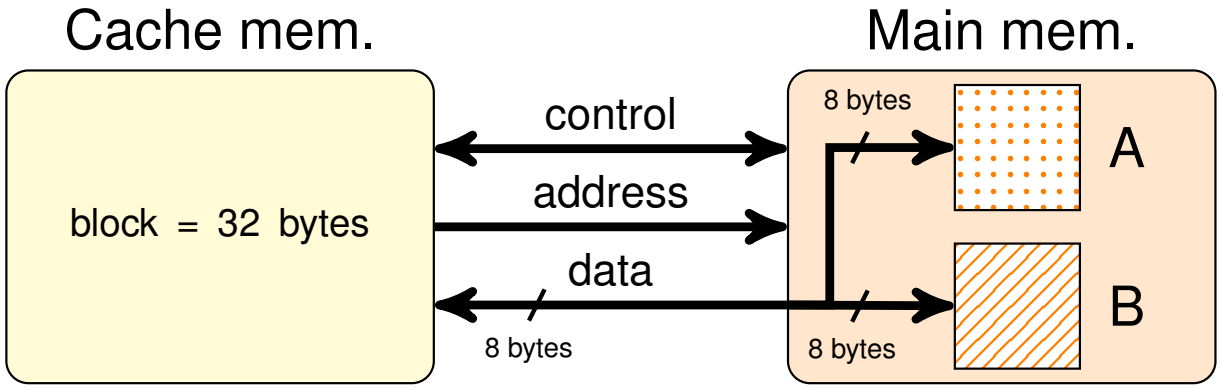


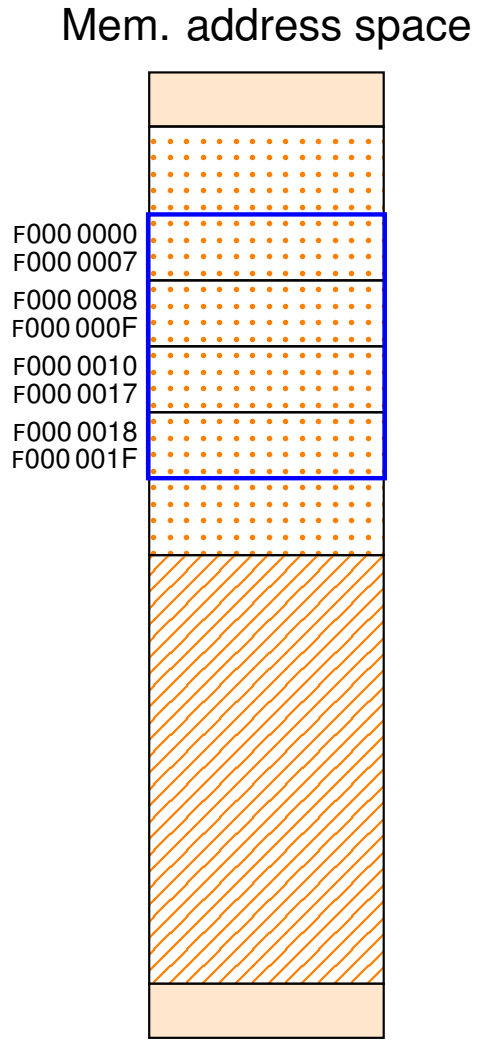
OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles



B

Cost of updating a block





OP1 = 1 clock cycle; OP2 = 1 clock cycle; OP3 = 4 clock cycles

$$\text{Miss penalty} = \underbrace{(32/8)}_{\text{accesses}} \times \underbrace{(1+1+4)}_{\text{time/access}} = 24 \text{ clock cycles}$$

Performance improvements

Accessing to the memory is a critical operation

The CPU waits several clock cycles

Improvements (combinable)

- Manufacturing technology
- Organization
  - Concurrent accesses $\Rightarrow$ interleaved memory
  - Transmission $\Rightarrow$ data bus expansion
  - Locality $\Rightarrow$ burst mode

Interleaved memory

Non-interleaved memory access only uses a memory bank

Key

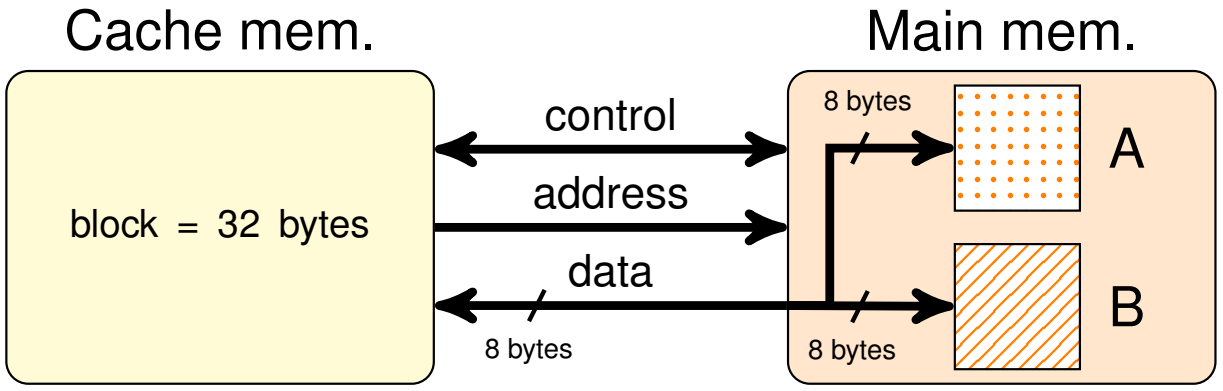
Several banks operate concurrently in each memory access

- Words are mapped alternatively in banks

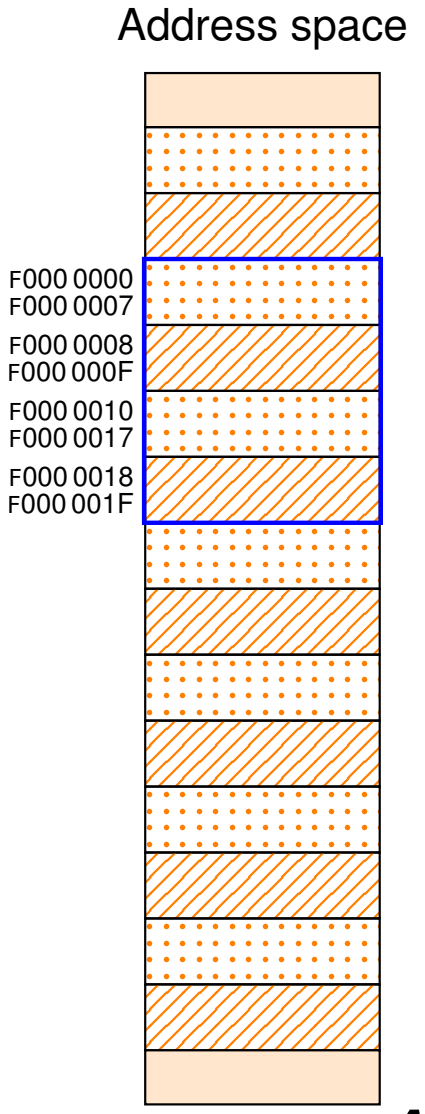
Advantages

- The access time decreases in a factor similar to the number of banks
- No additional cost

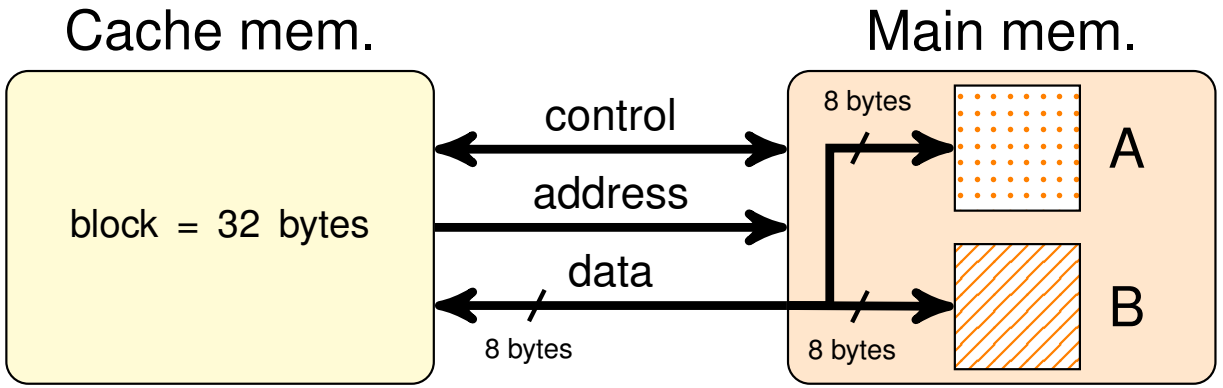
Interleaved memory



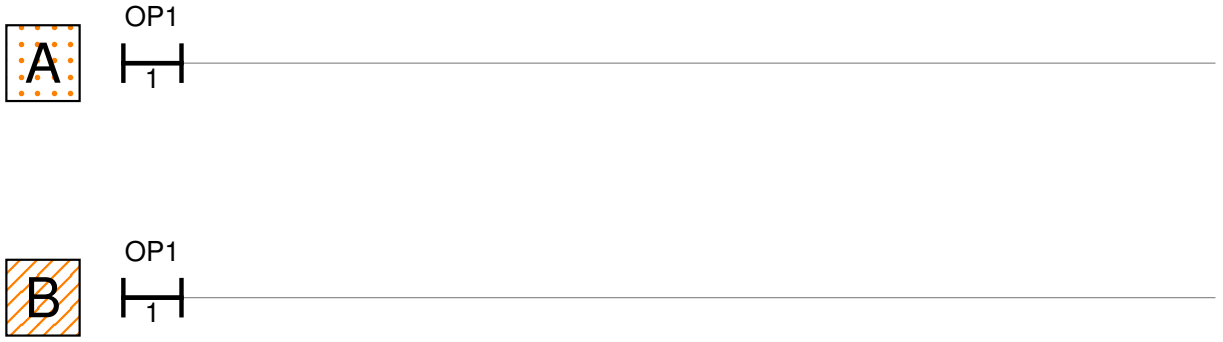
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

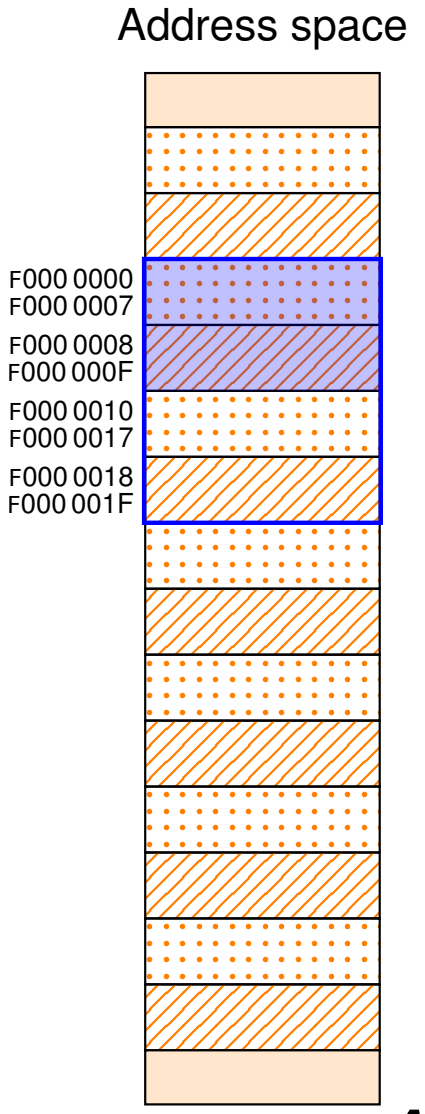


Interleaved memory

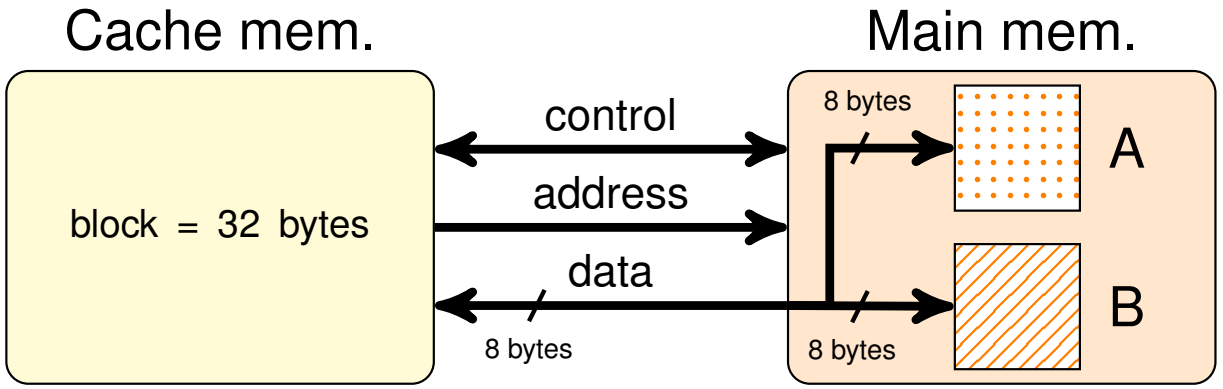


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

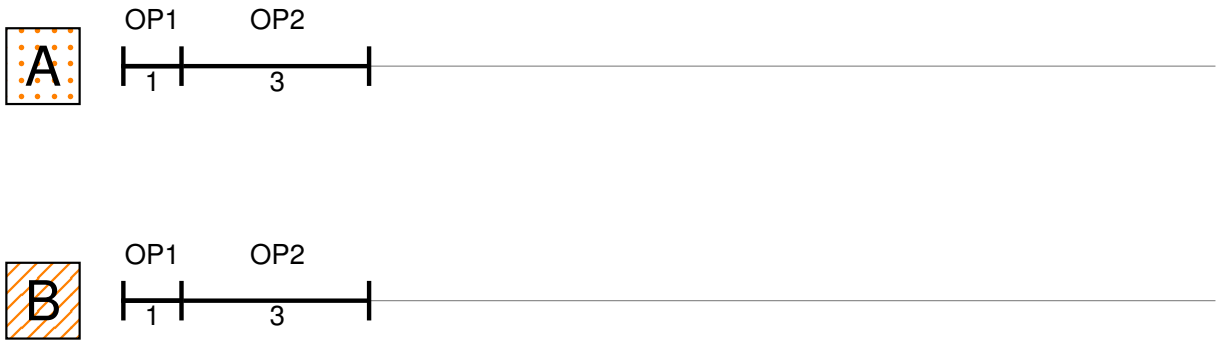


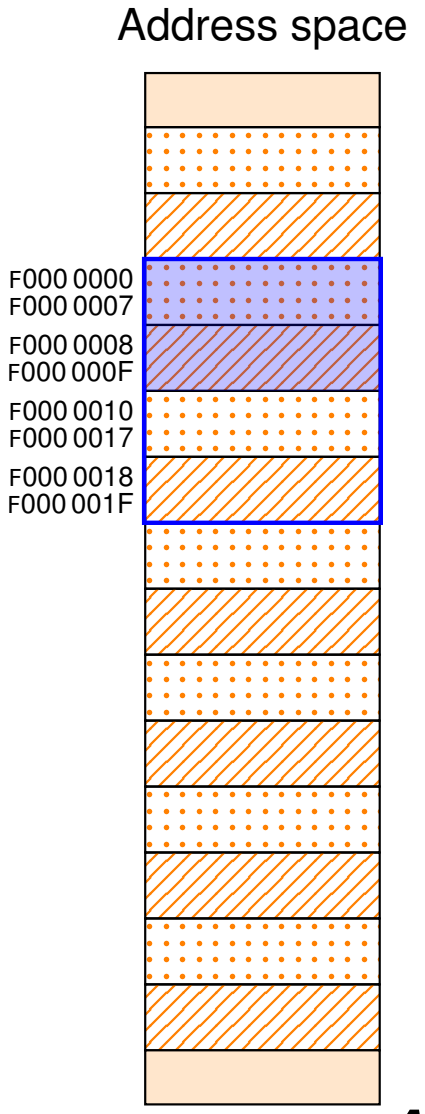


Interleaved memory

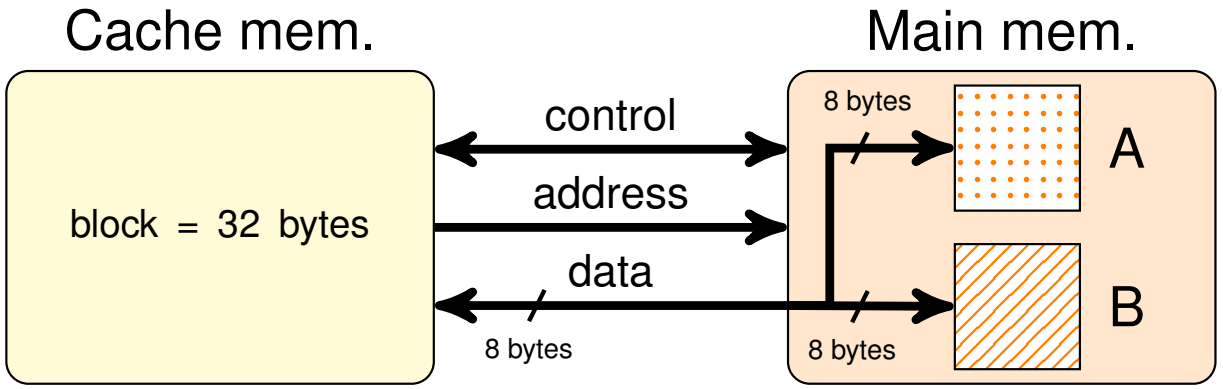


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

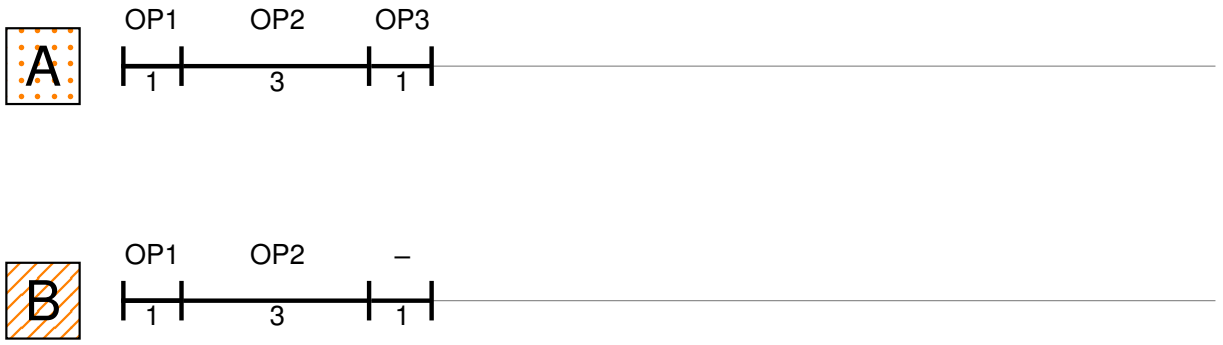


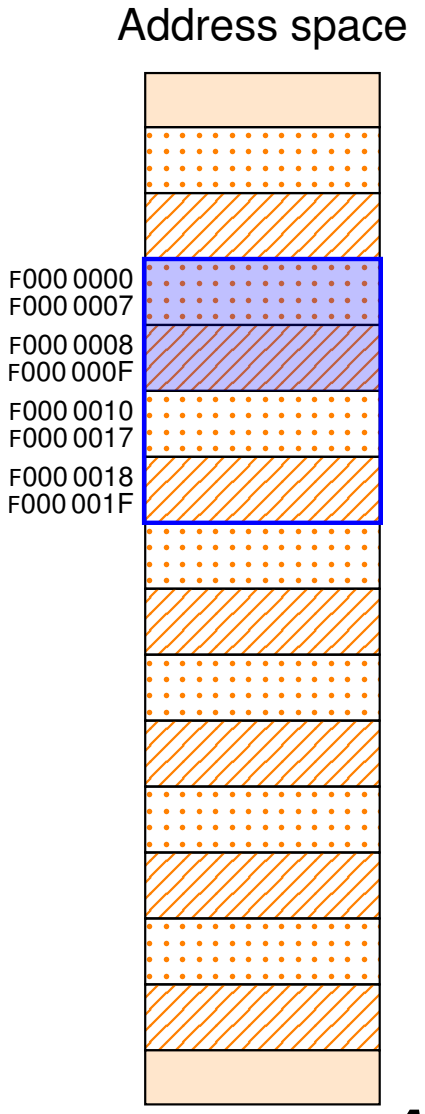


Interleaved memory

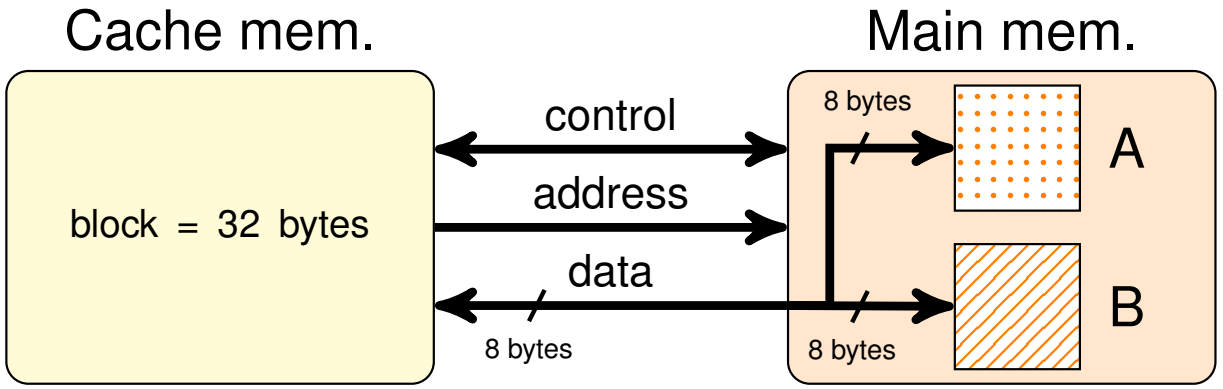


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

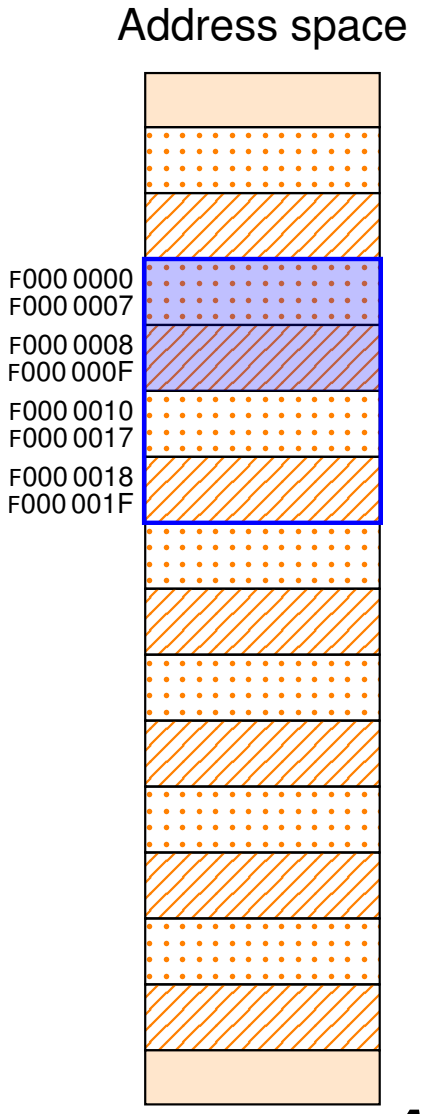




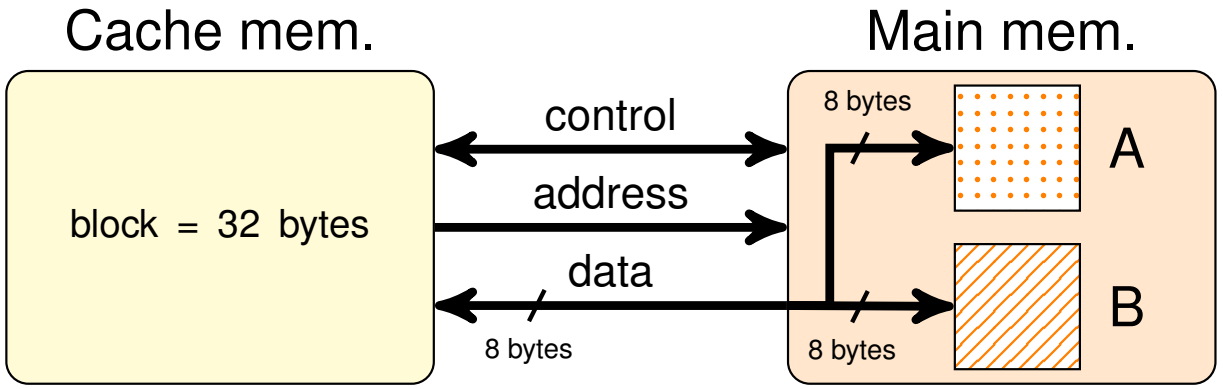
Interleaved memory



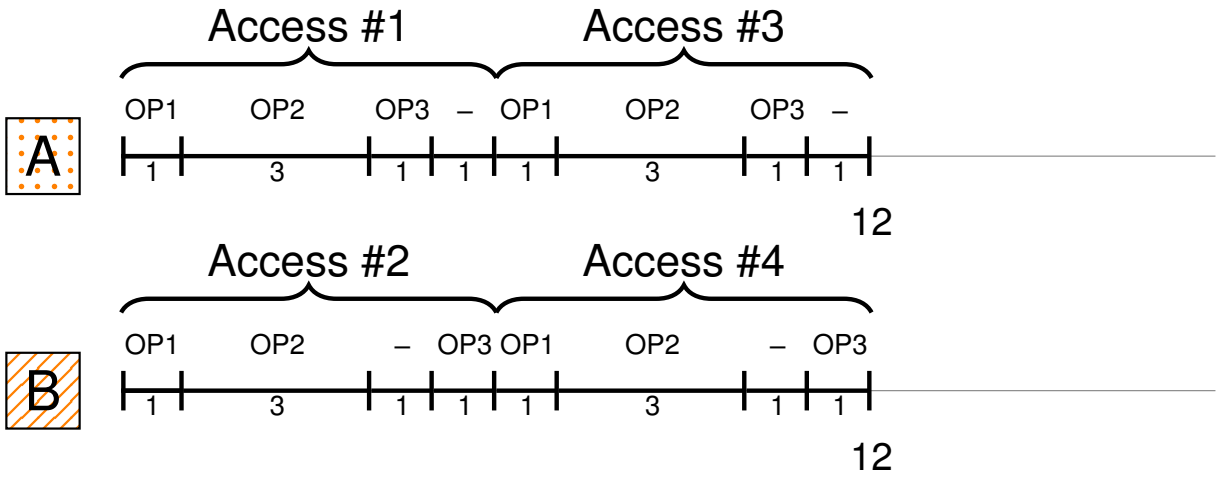
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

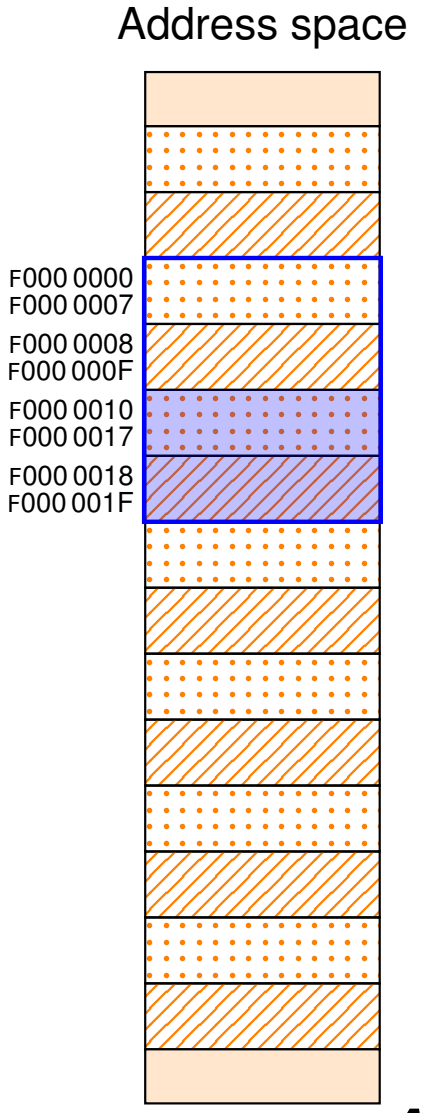


Interleaved memory

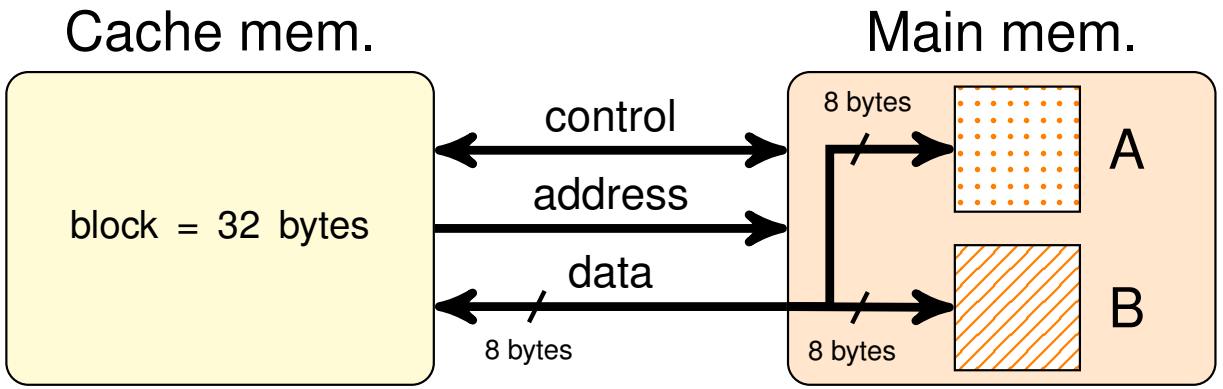


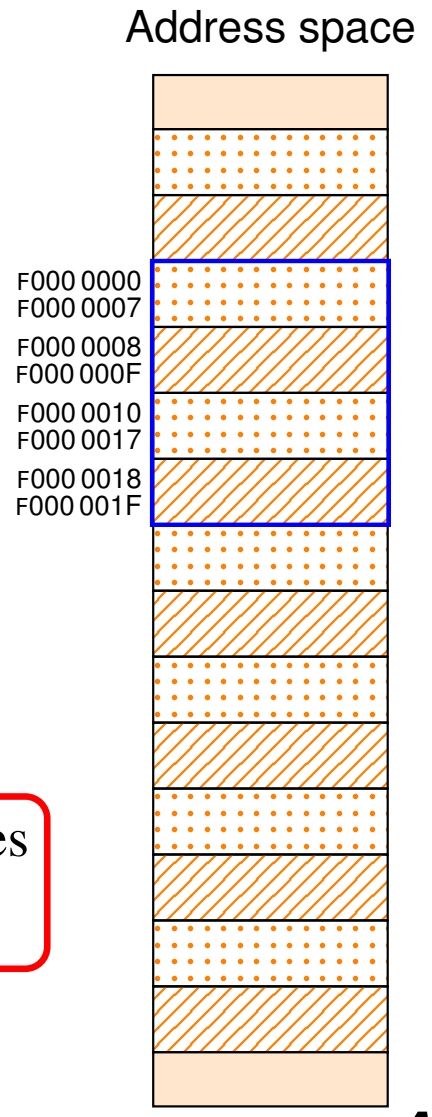
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle





Interleaved memory





OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

$$\text{Miss penalty} = \underbrace{(32/8)}_{\text{accesses}} \times \underbrace{(1/2 + 3/2 + 1)}_{\text{time/access}} = 12 \text{ clock cycles}$$

Data bus expansion

Transferences with interleaved memory are not concurrent

- The data bus does not have enough capacity

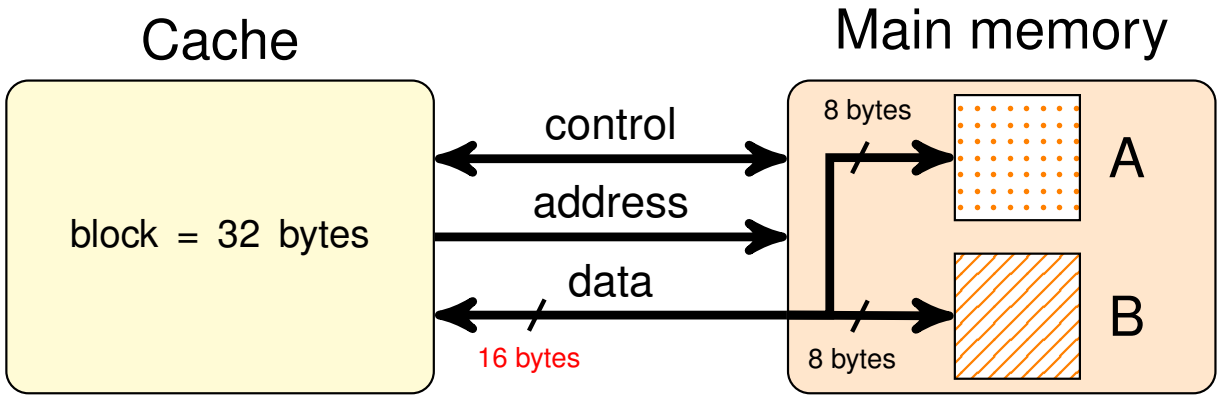
Key

Expand the capacity of the data bus

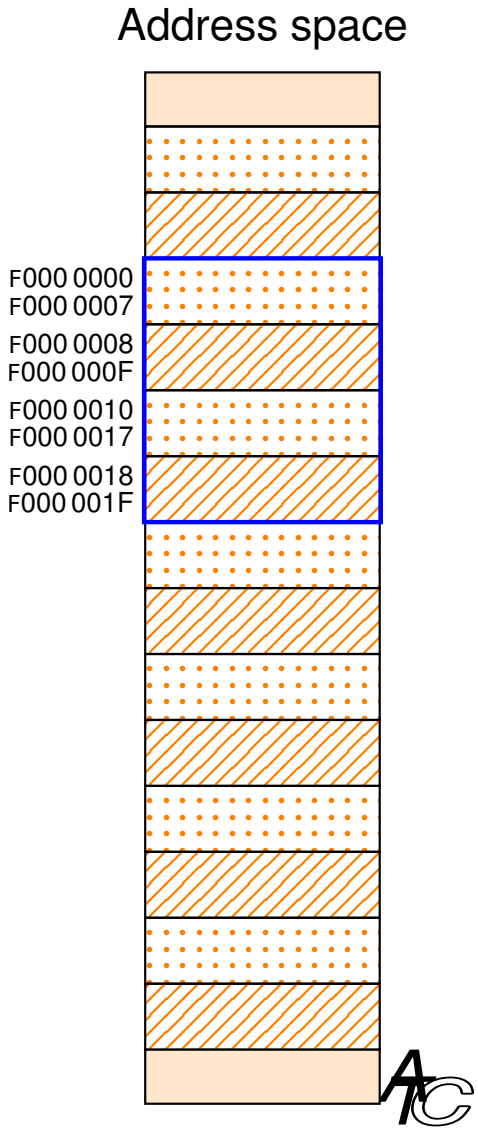
Limitation

- Very high cost $\Rightarrow$ performance/cost tradeoff

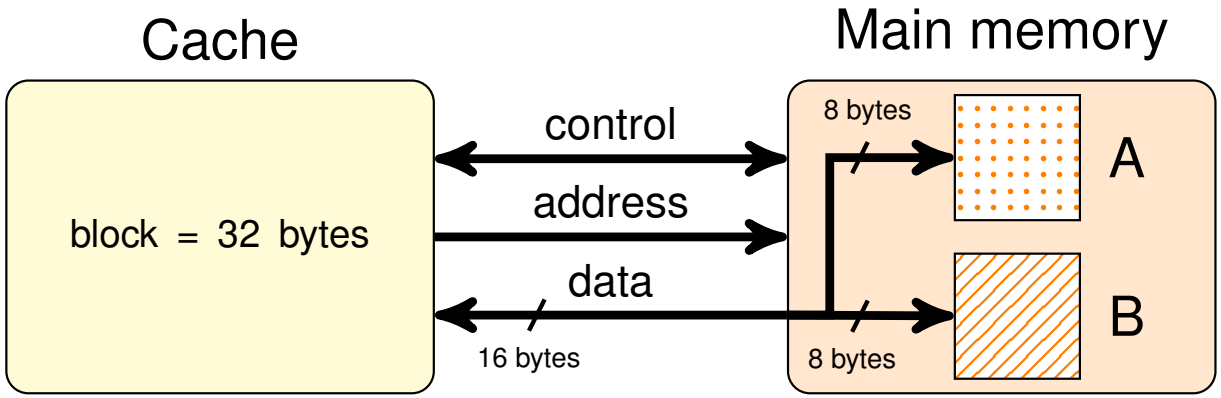
Interleaved memory and expanded data bus



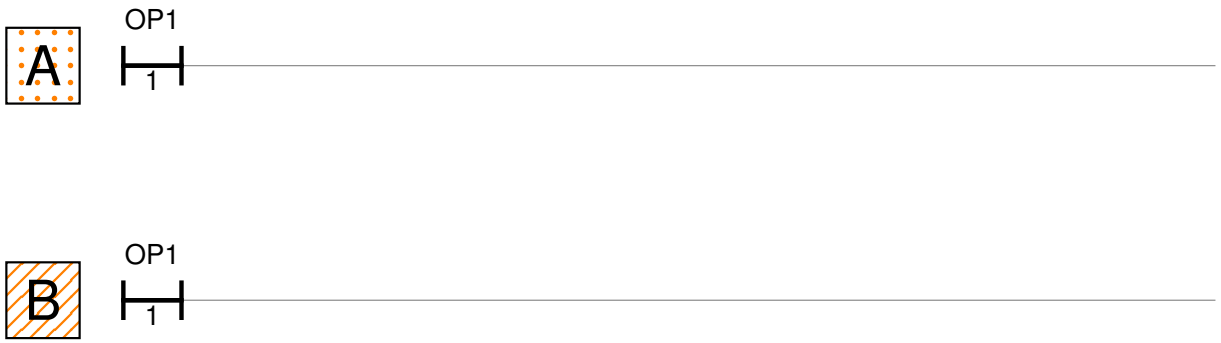
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

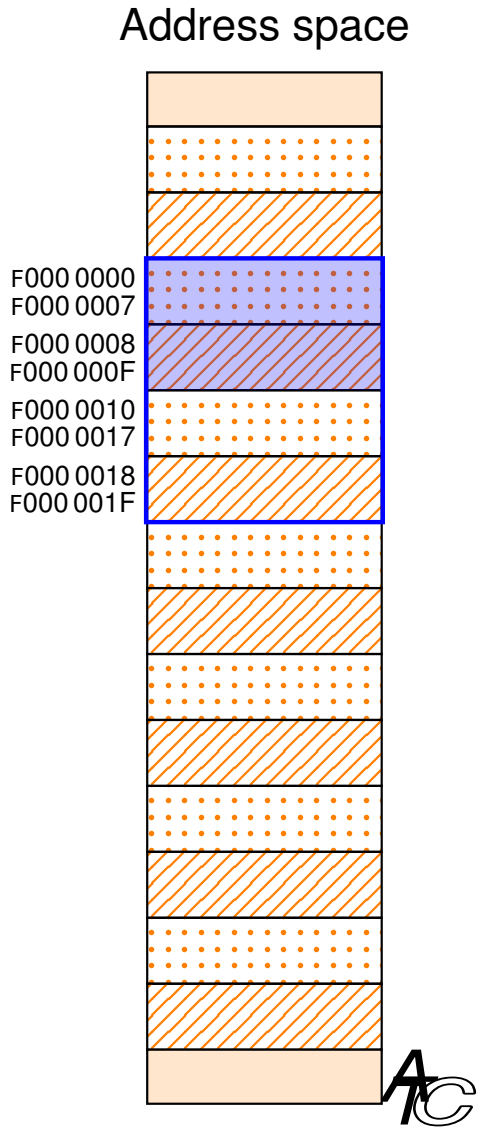


Interleaved memory and expanded data bus

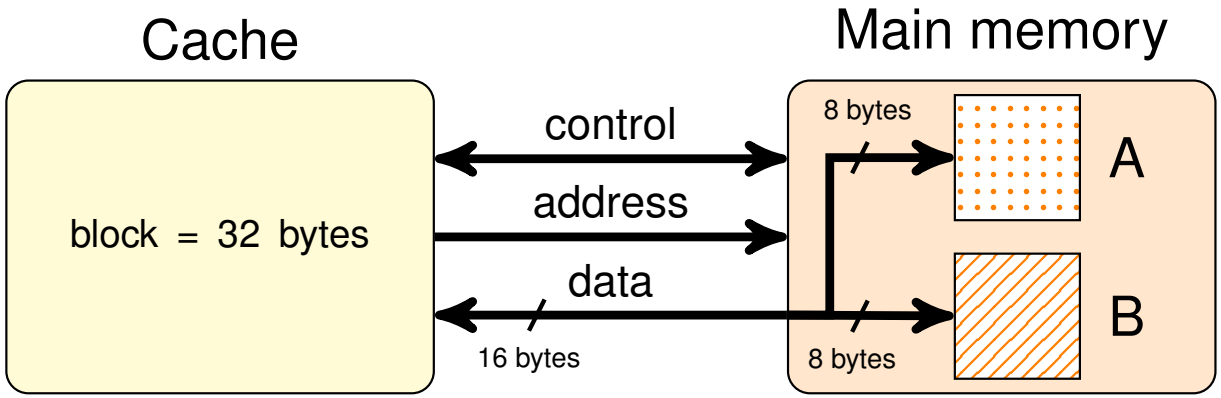


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

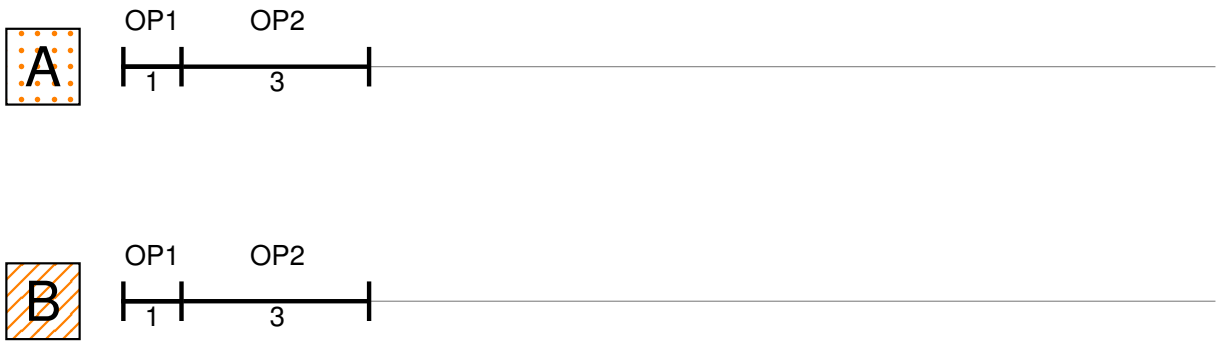


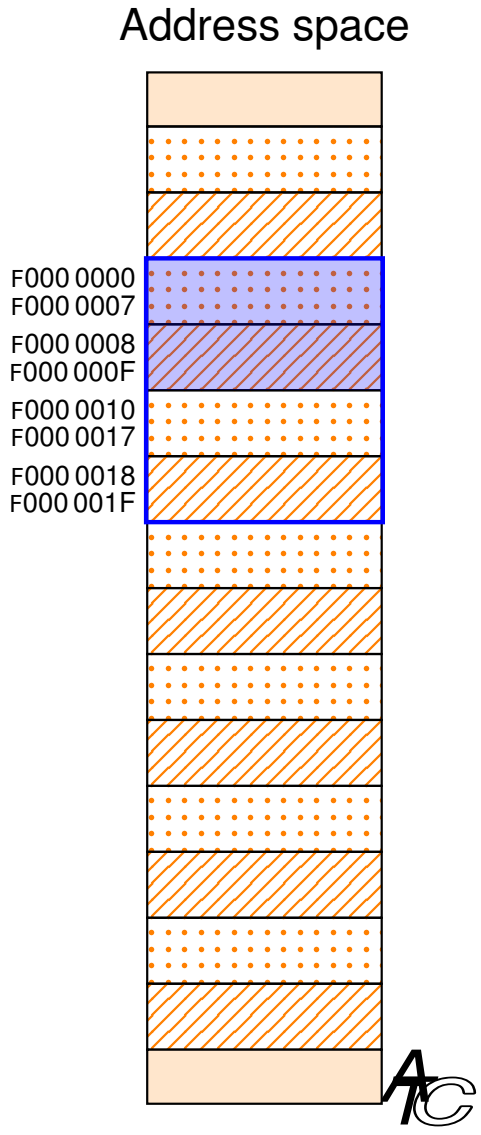


Interleaved memory and expanded data bus

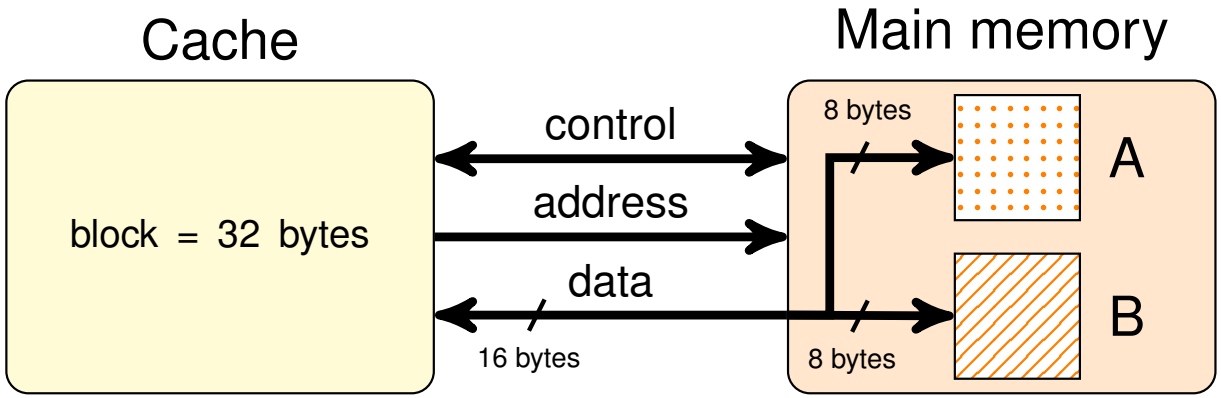


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

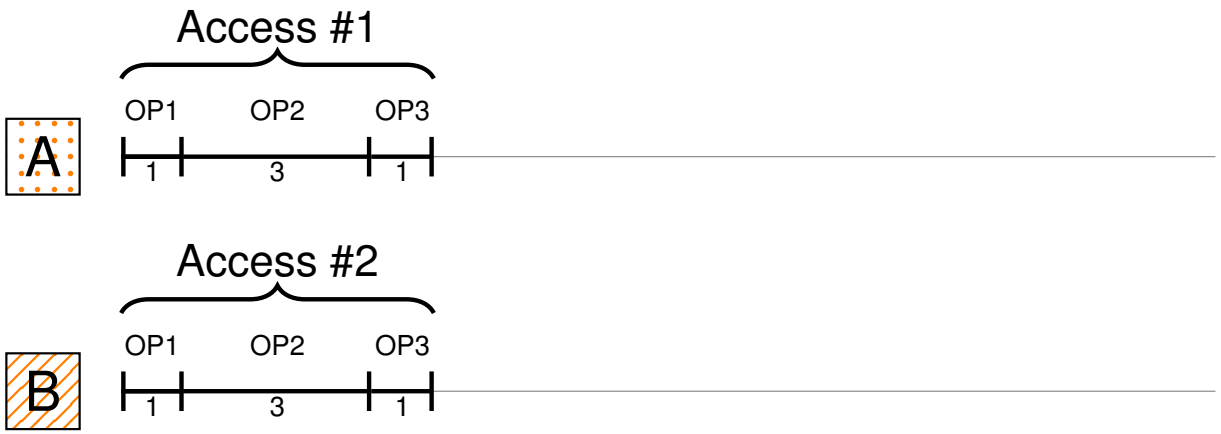


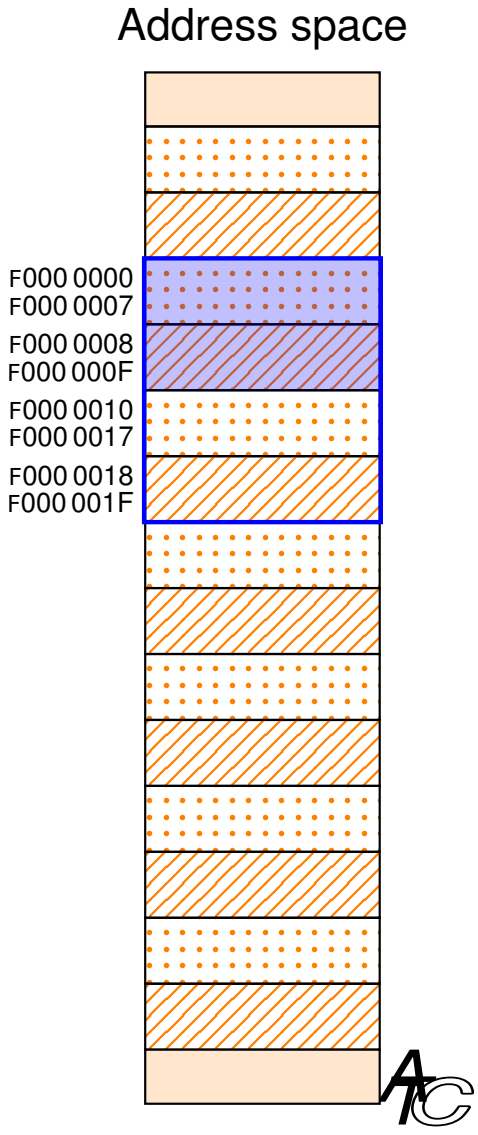


Interleaved memory and expanded data bus

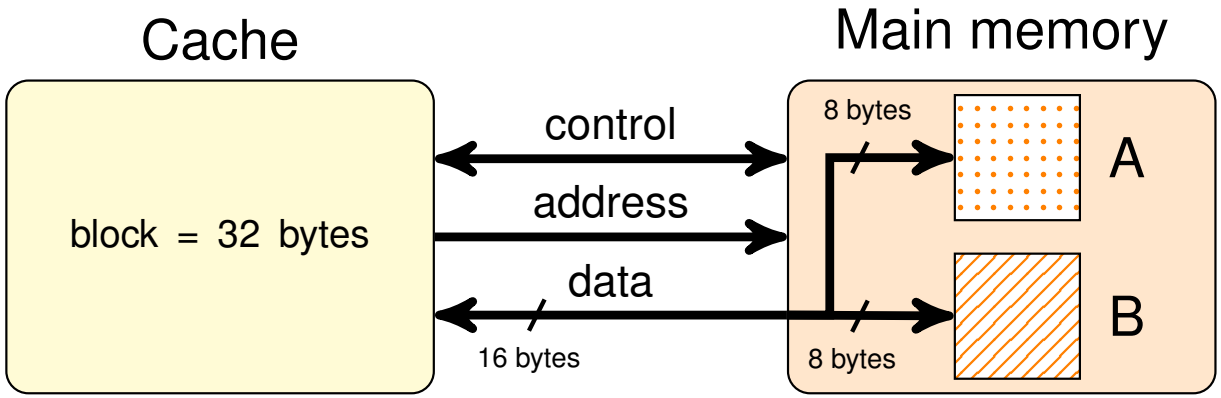


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

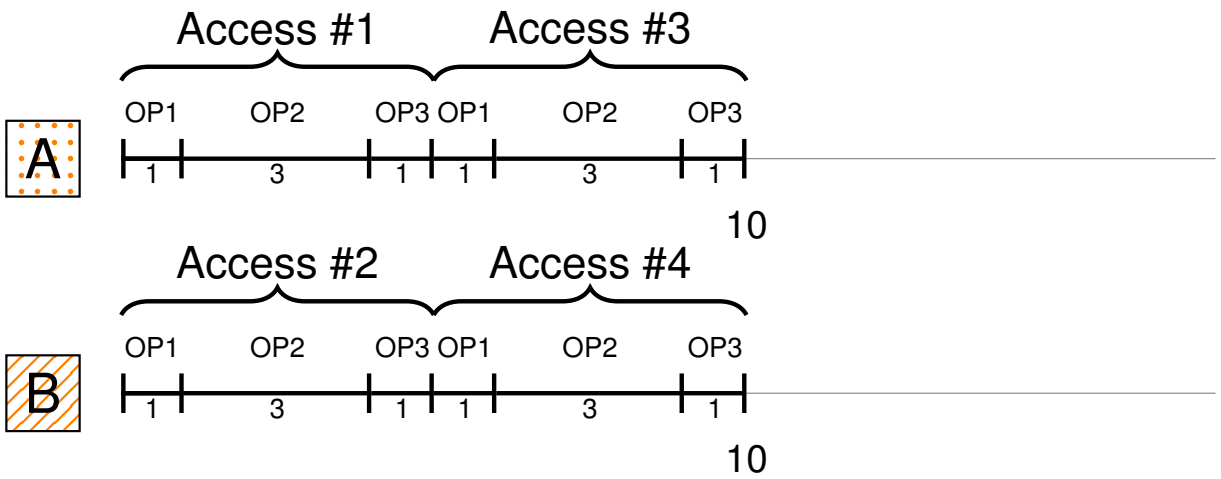


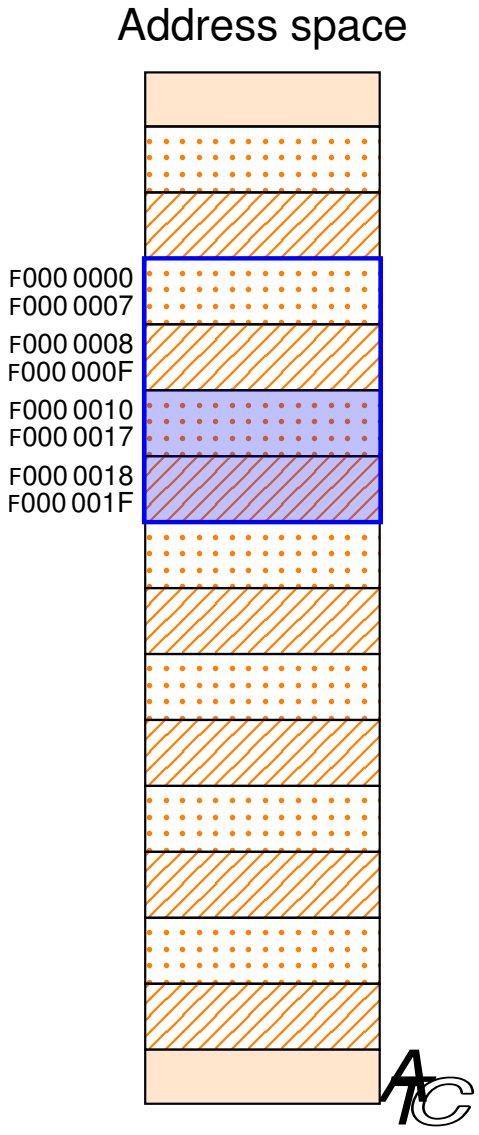


Interleaved memory and expanded data bus

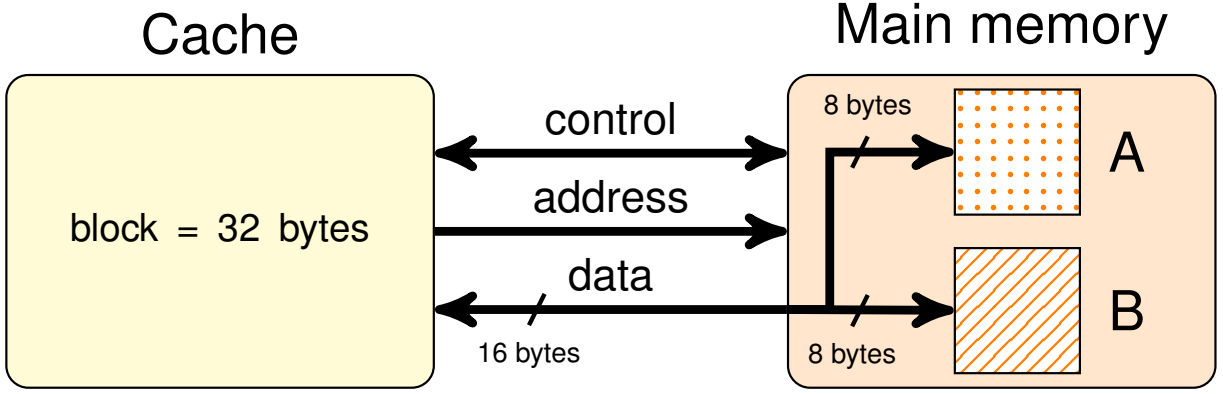


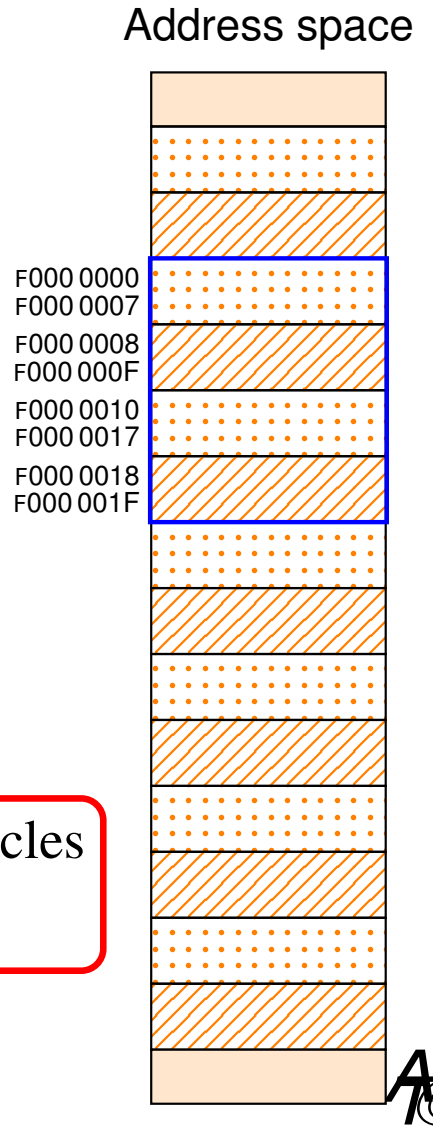
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle





Interleaved memory and expanded data bus





OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

$$\text{Miss penalty} = \underbrace{(32/8)}_{\text{accesses}} \times \underbrace{(1/2 + 3/2 + 1/2)}_{\text{time/access}} = 10 \text{ clock cycles}$$

Burst access mode

Take advantage of spatial locality

- block = consecutive memory locations

Key

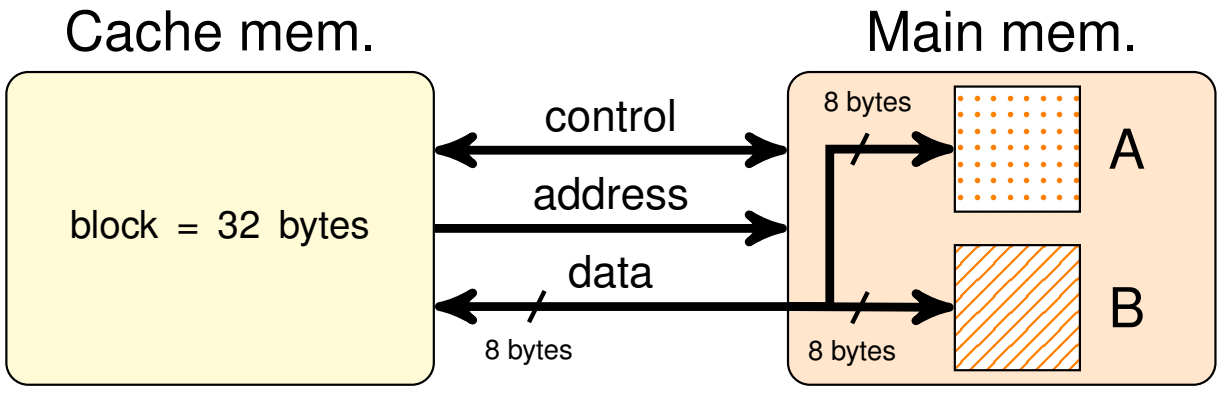
Send address only once

- Add intelligence to the memory

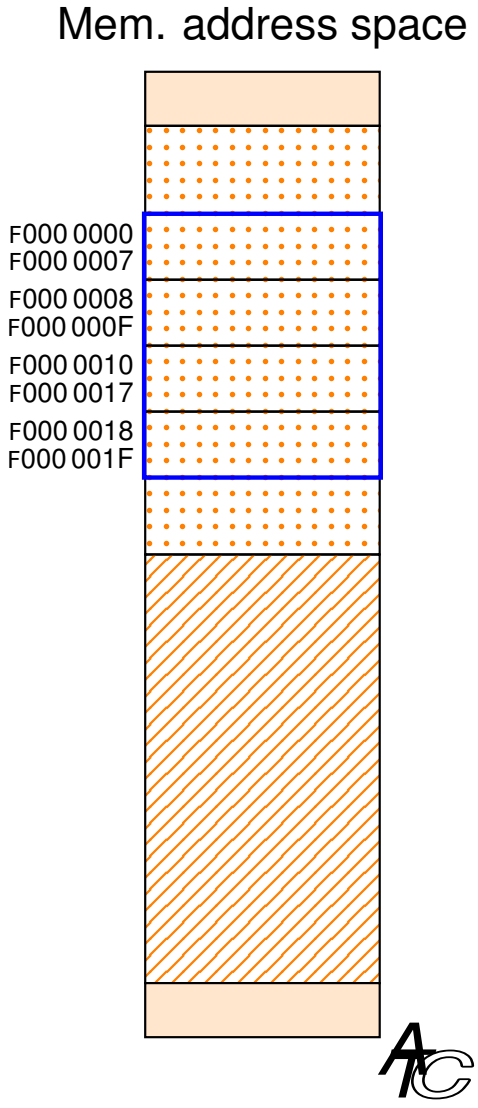
Advantages

- The OP1 operation is carried out only once
- Low cost

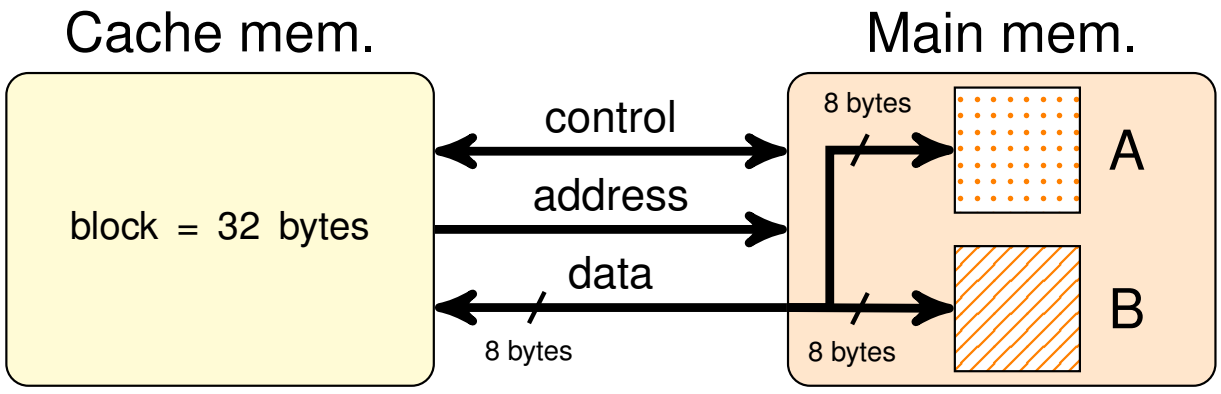
Non-interleaved memory and burst access mode



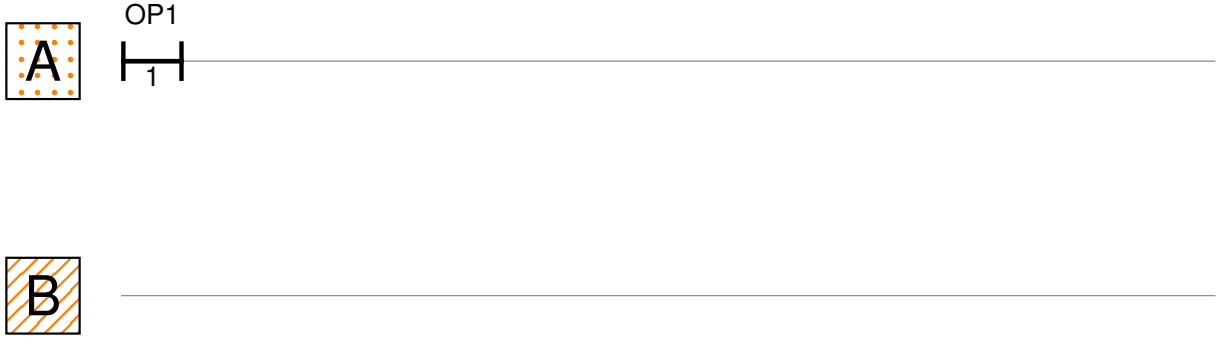
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles

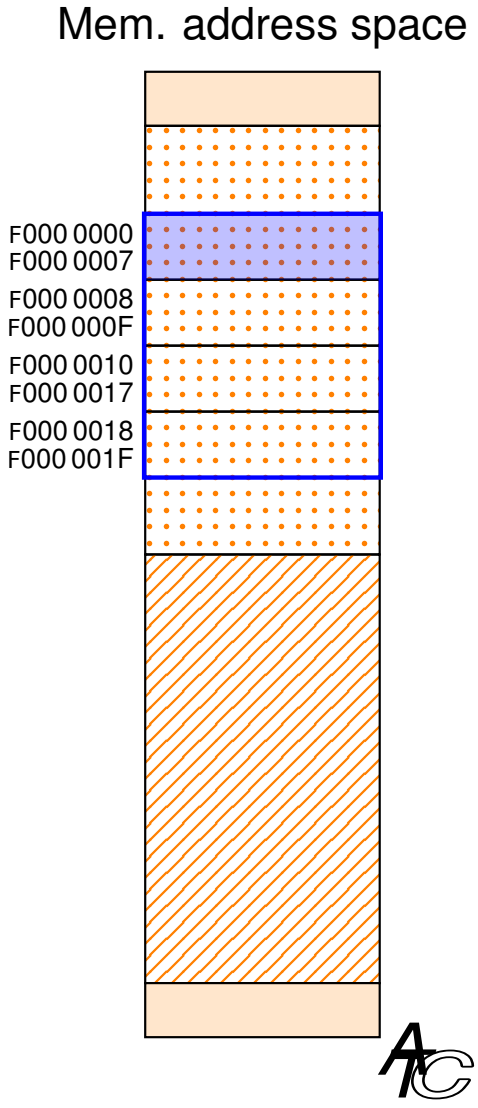


Non-interleaved memory and burst access mode

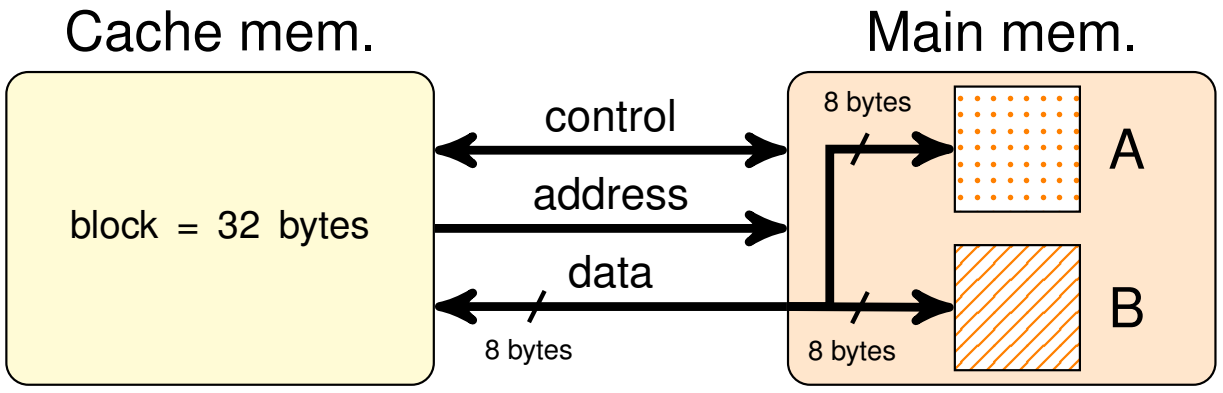


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles

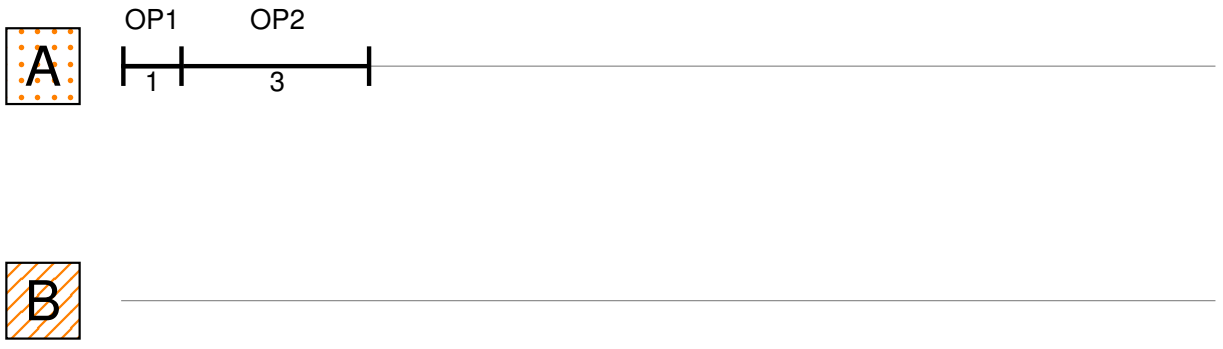


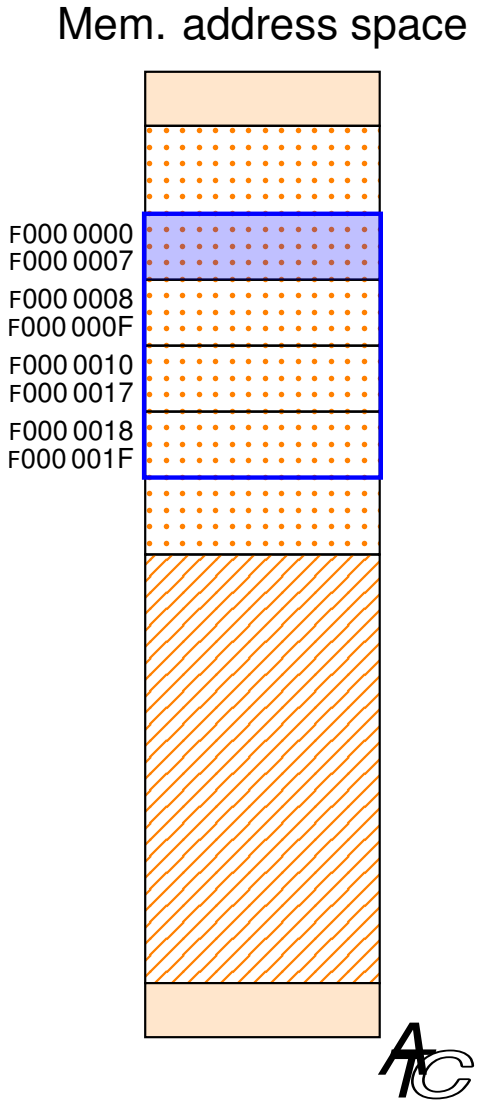


Non-interleaved memory and burst access mode

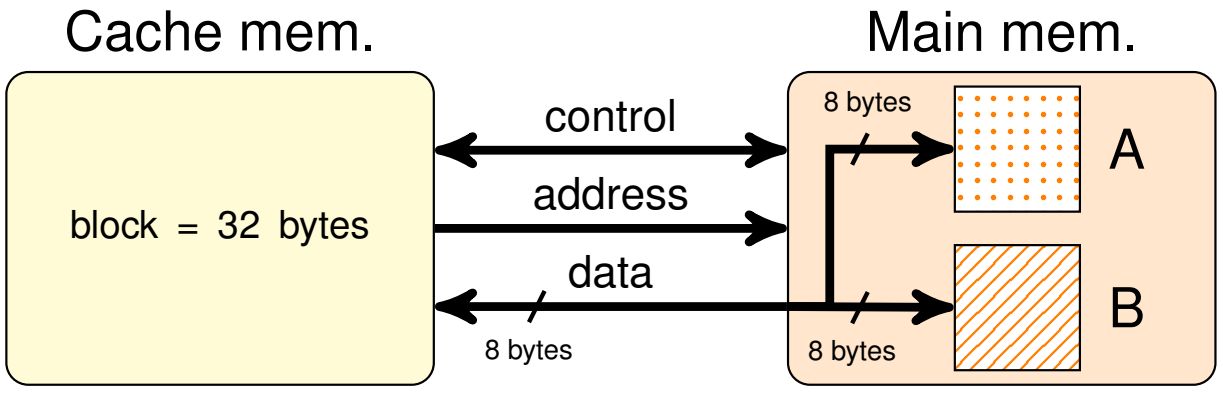


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles

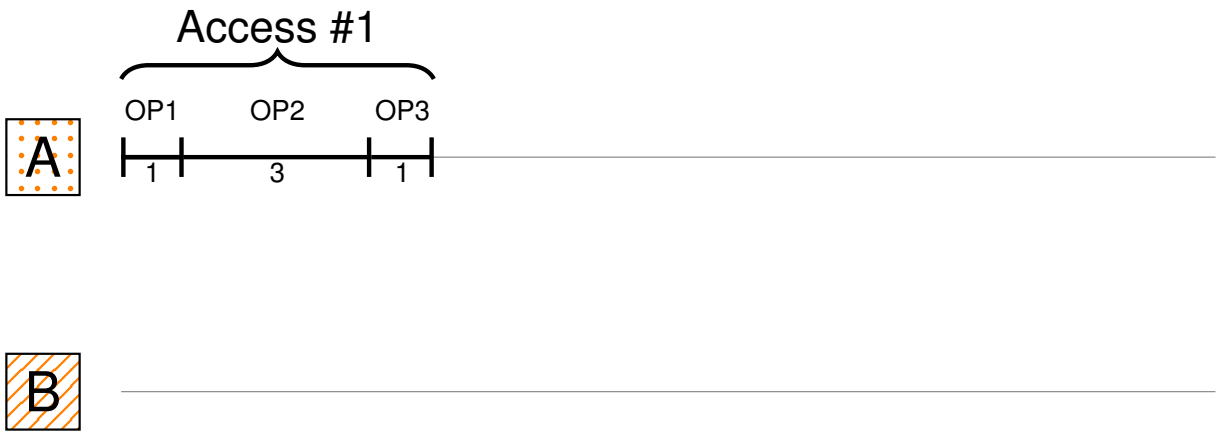


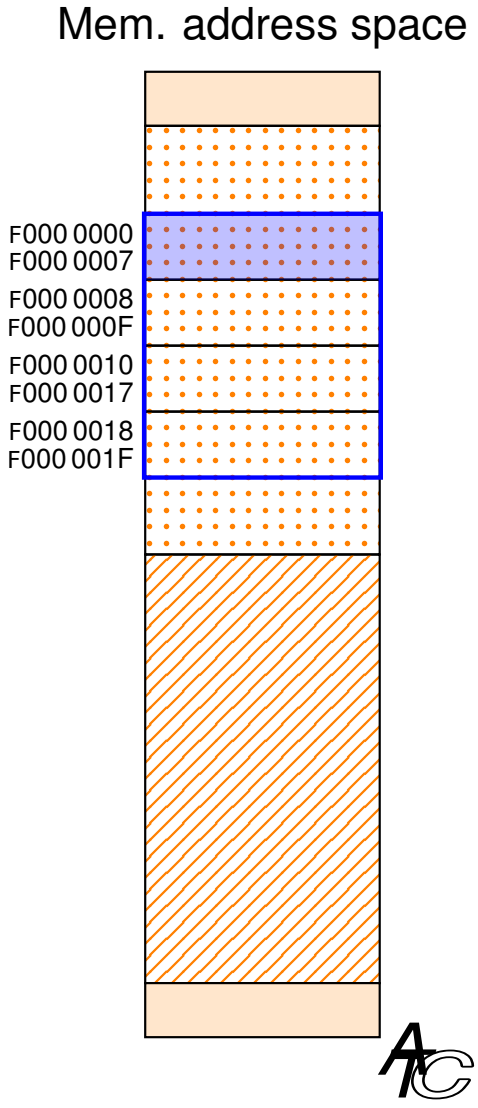


Non-interleaved memory and burst access mode

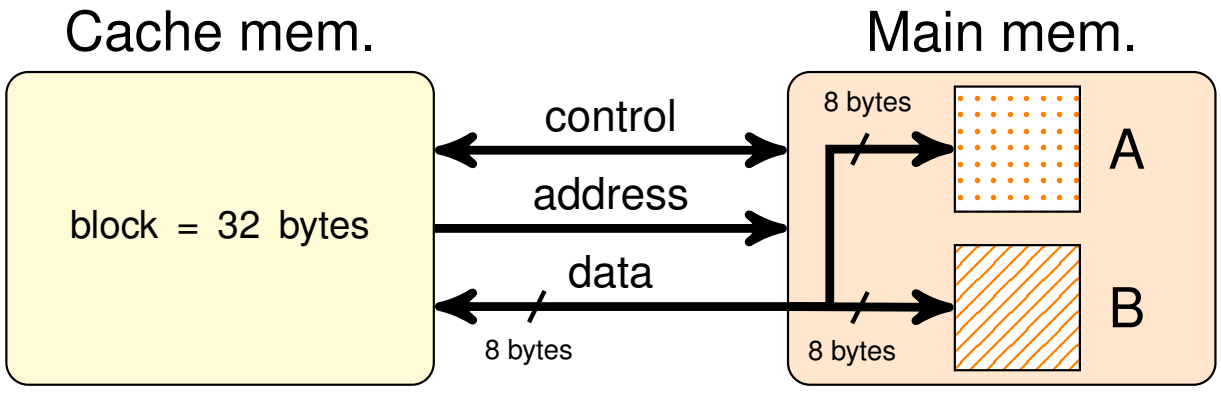


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles

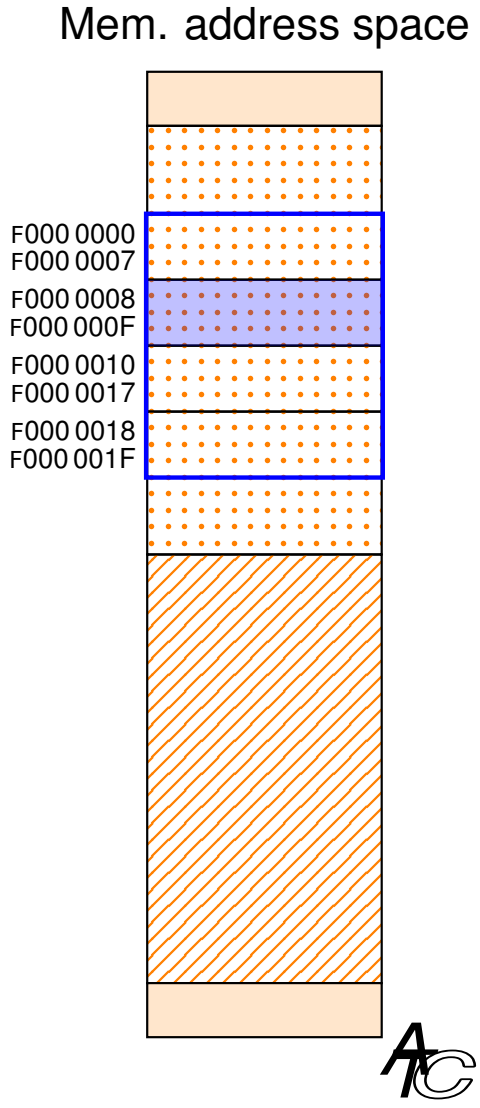




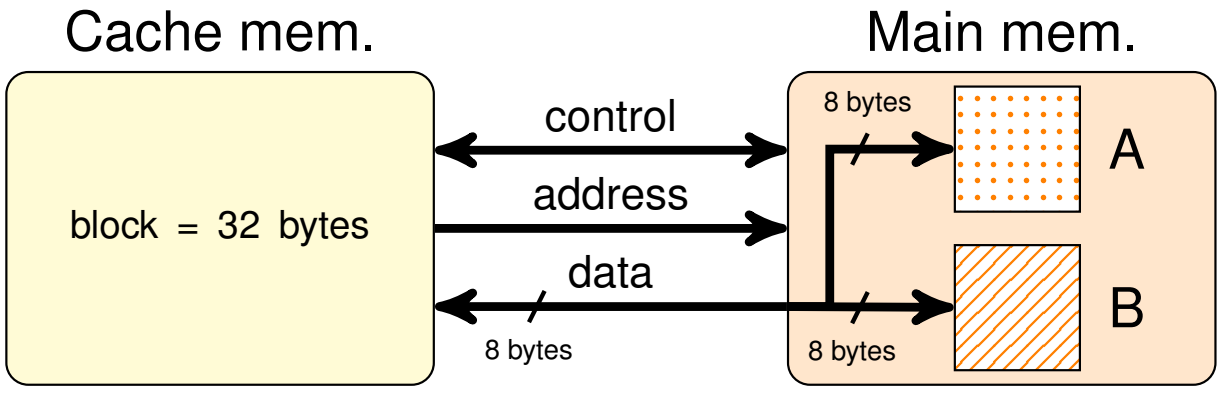
Non-interleaved memory and burst access mode



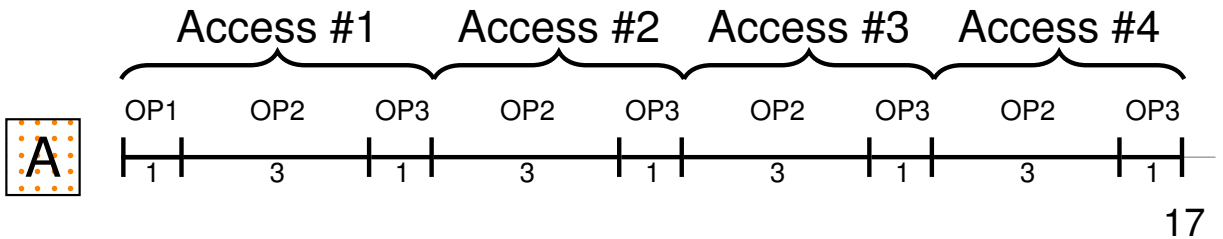
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles



Non-interleaved memory and burst access mode

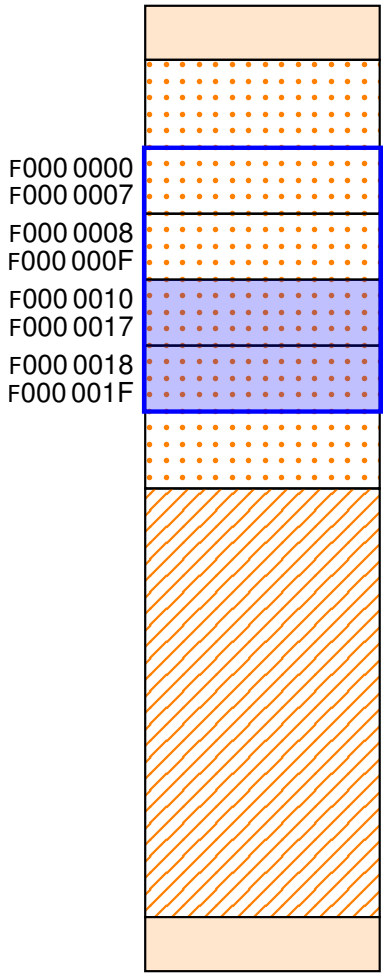


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles

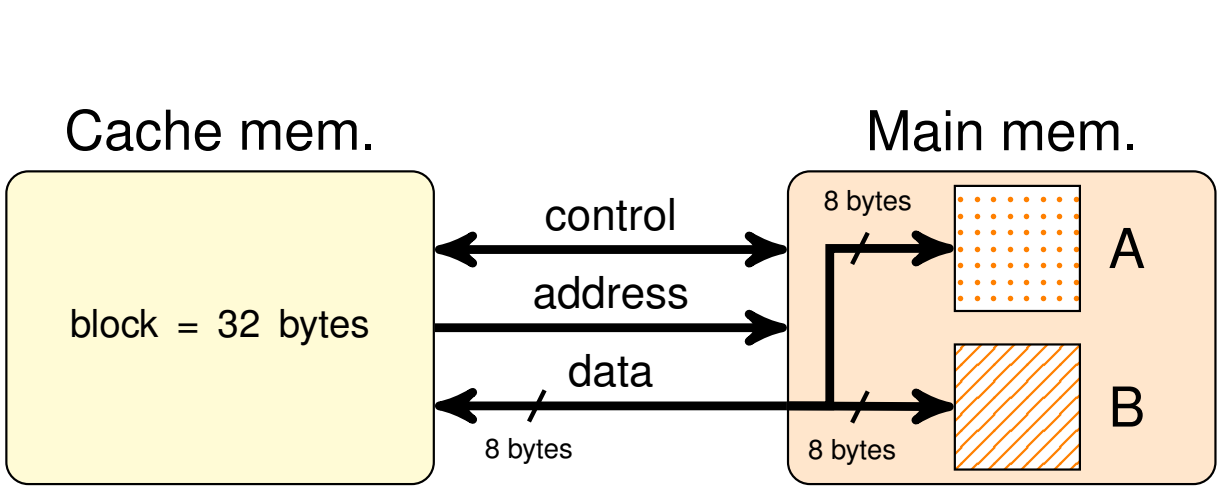


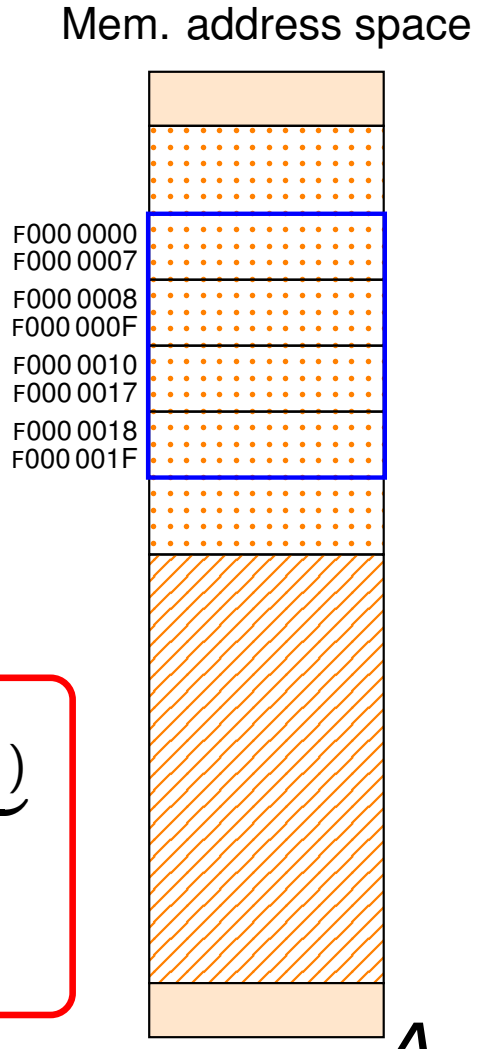
17

Mem. address space



Non-interleaved memory and burst access mode



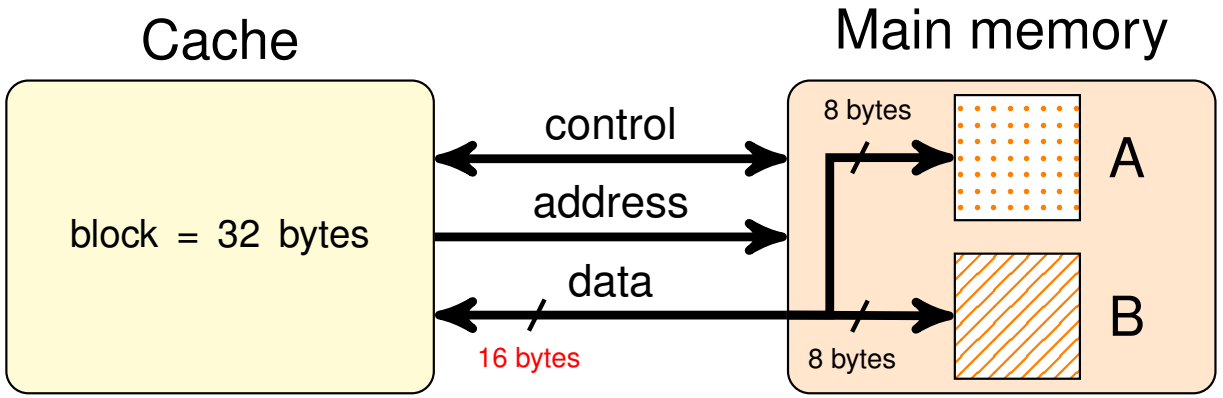


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycles

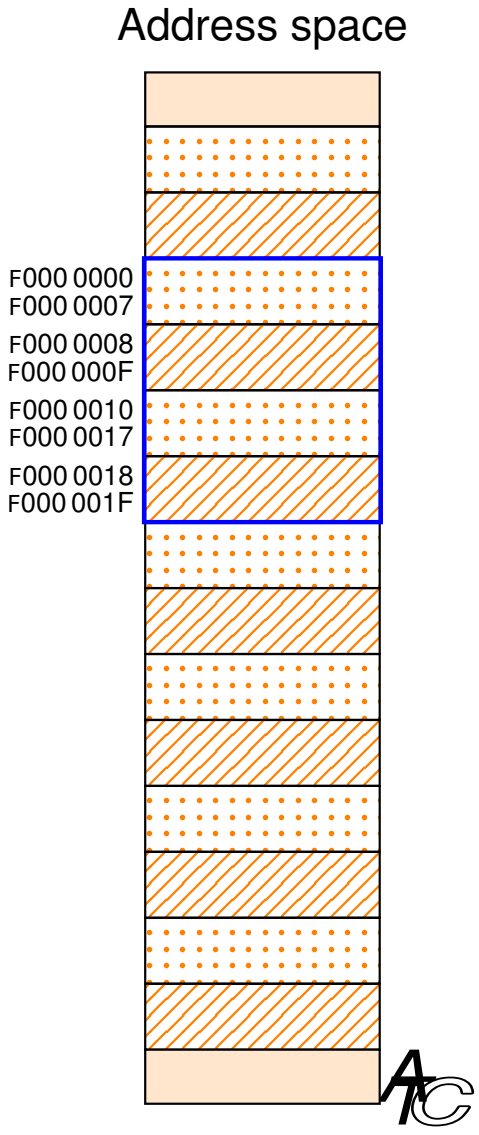
$$\text{Miss penalty} = \underbrace{1 \times (1+3+1)}_{1^{st}\text{ access}} + \underbrace{(32/8-1) \times (0+3+1)}_{\text{rest accesses}}$$

$$= 17 \text{ clock cycles}$$

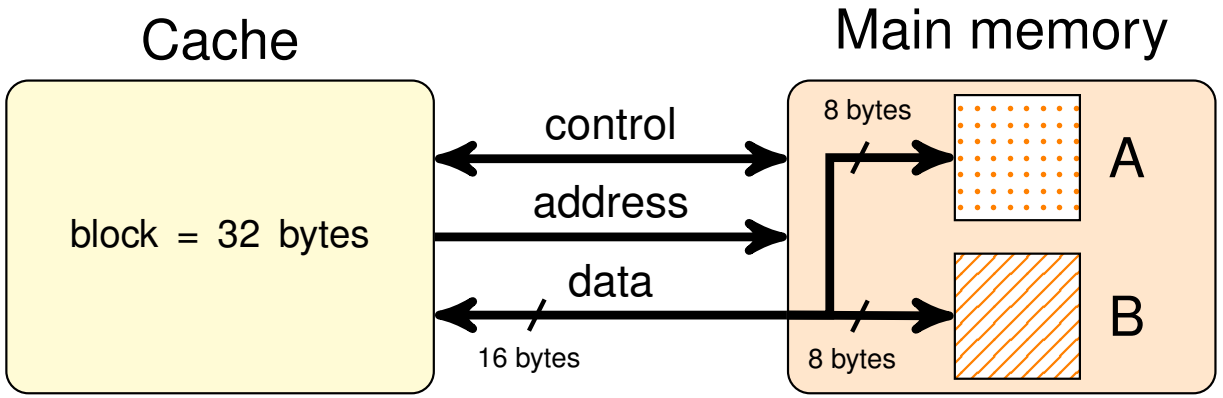
Interleaved mem., expanded bus, and burst mode



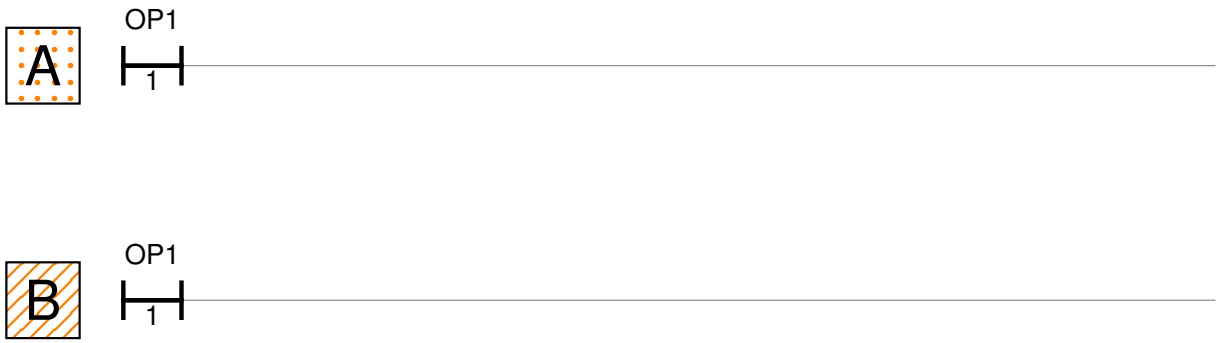
OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

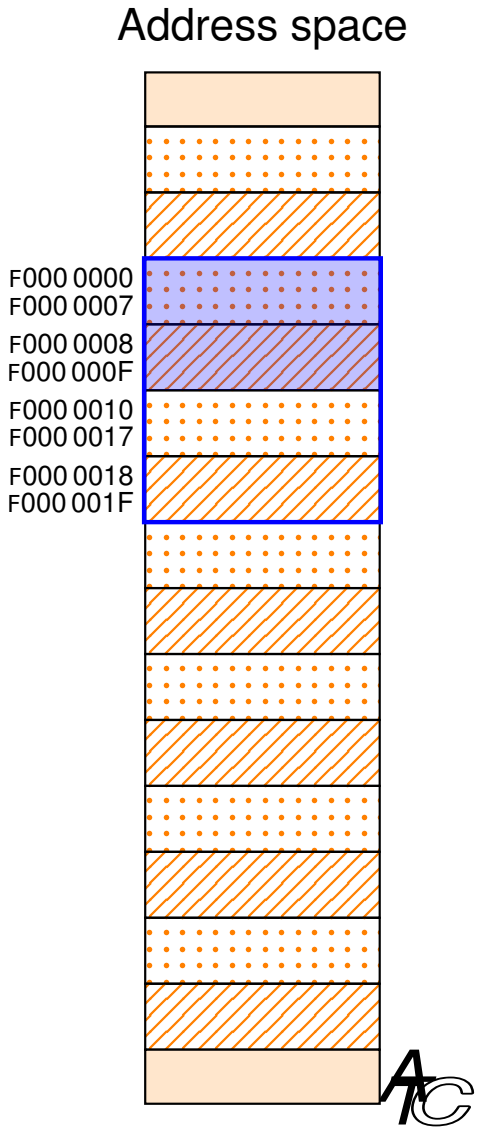


Interleaved mem., expanded bus, and burst mode

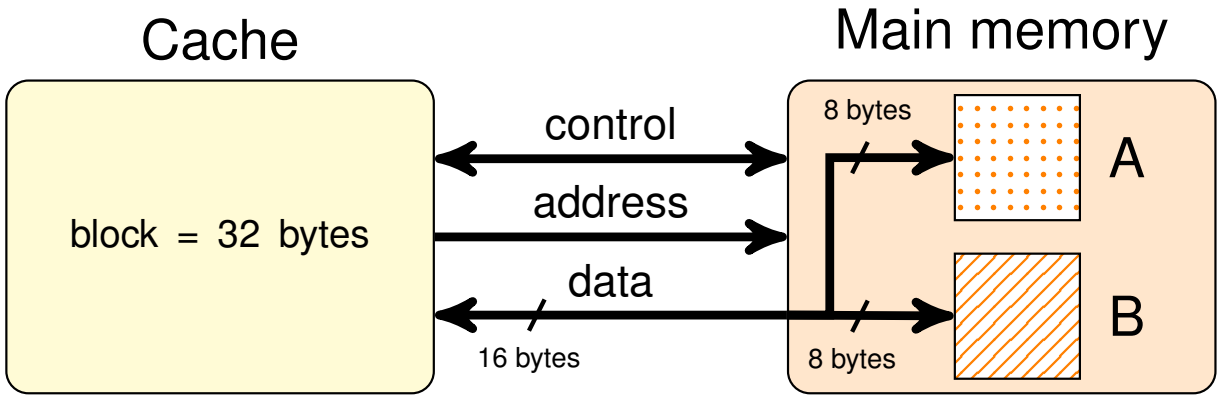


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

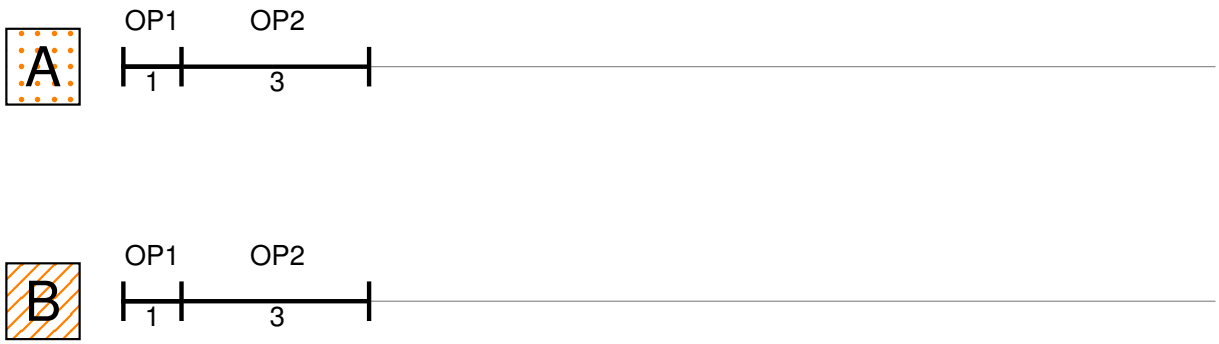


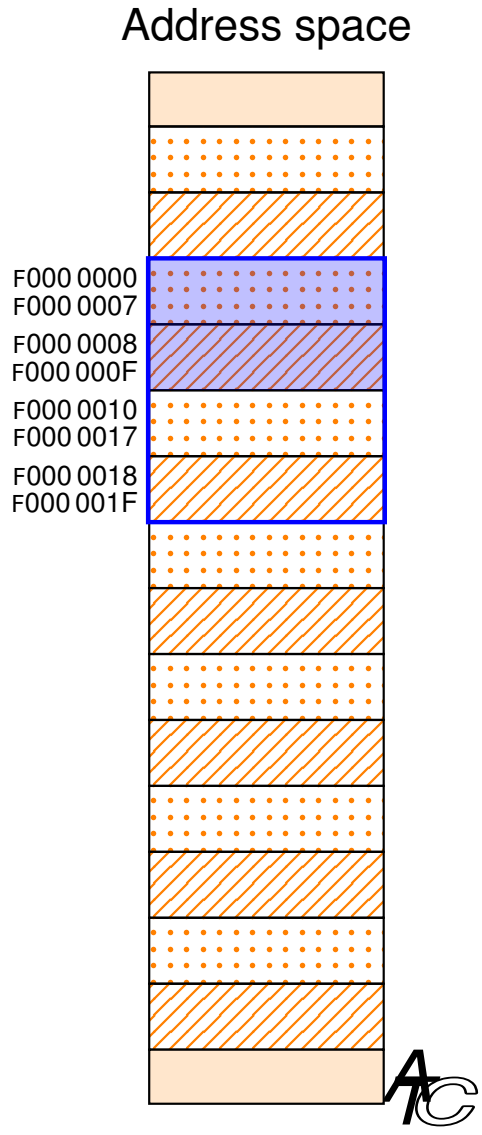


Interleaved mem., expanded bus, and burst mode

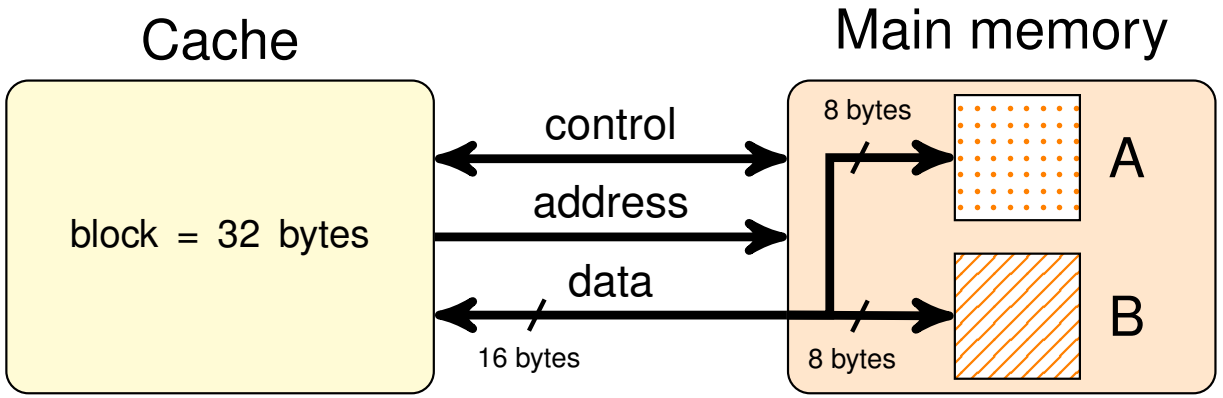


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

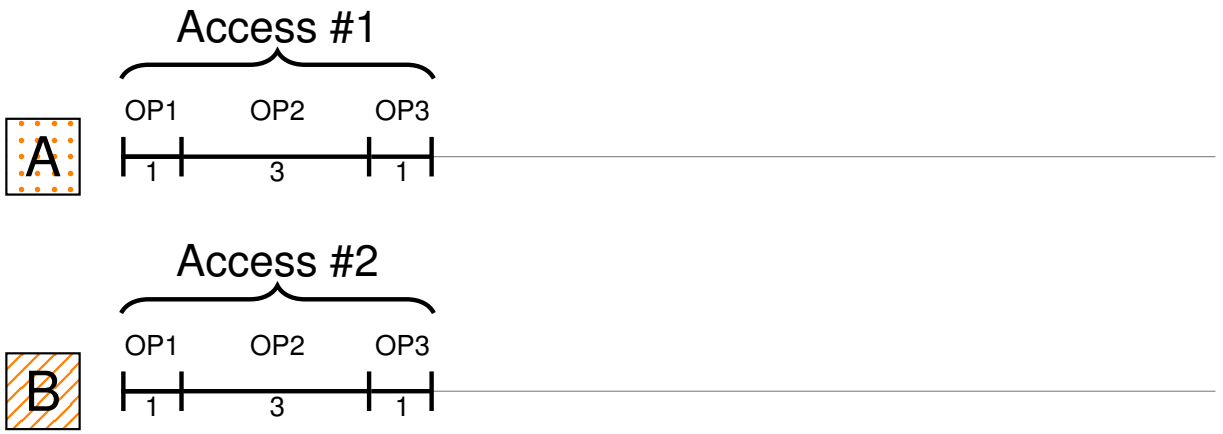


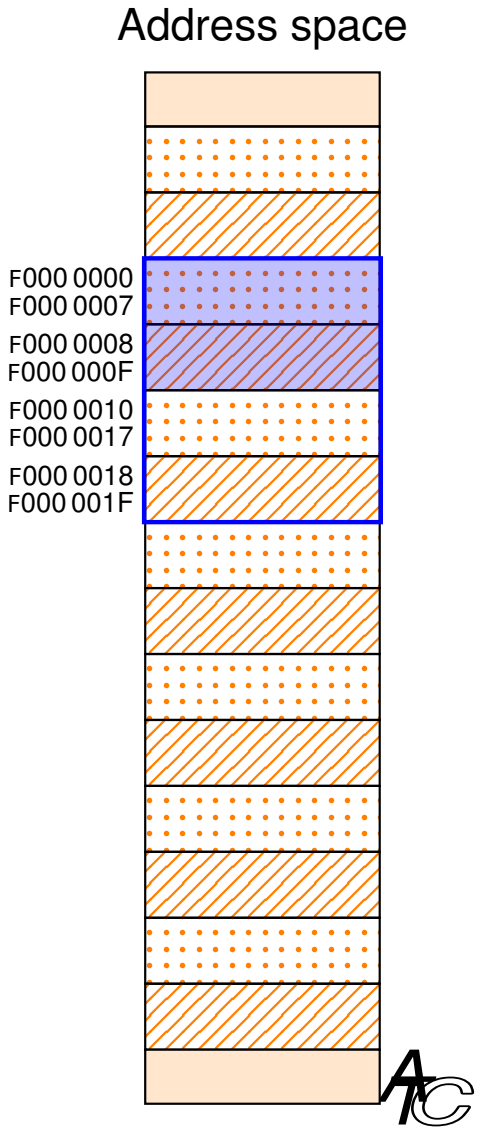


Interleaved mem., expanded bus, and burst mode

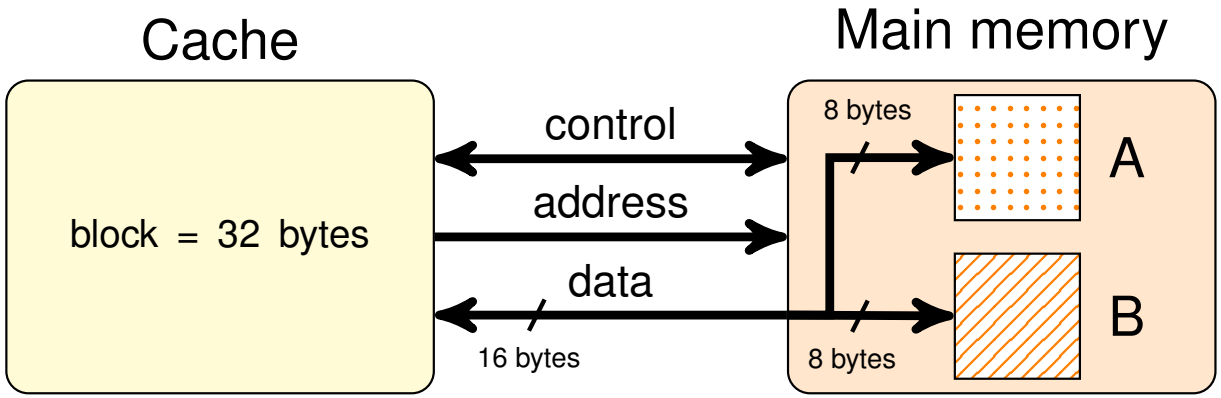


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

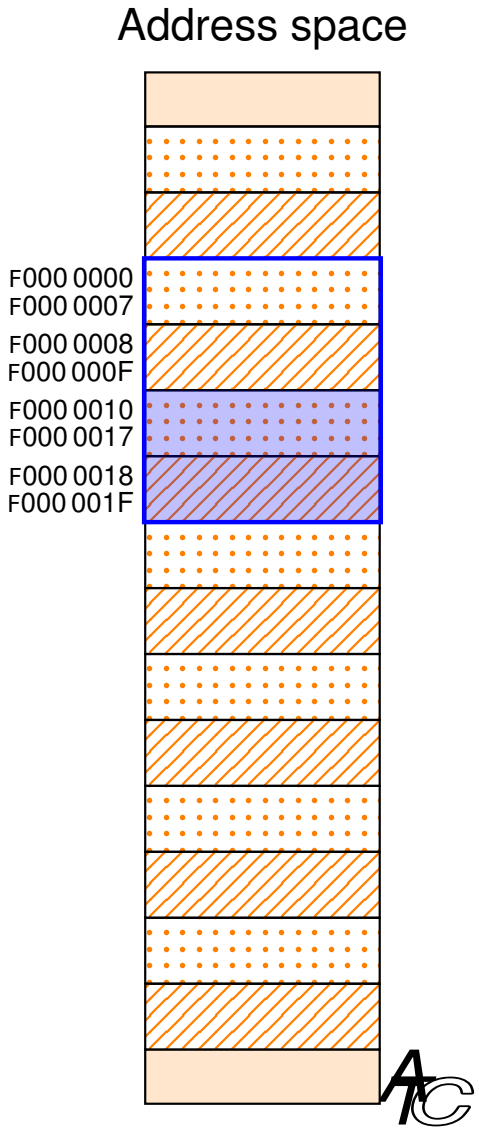




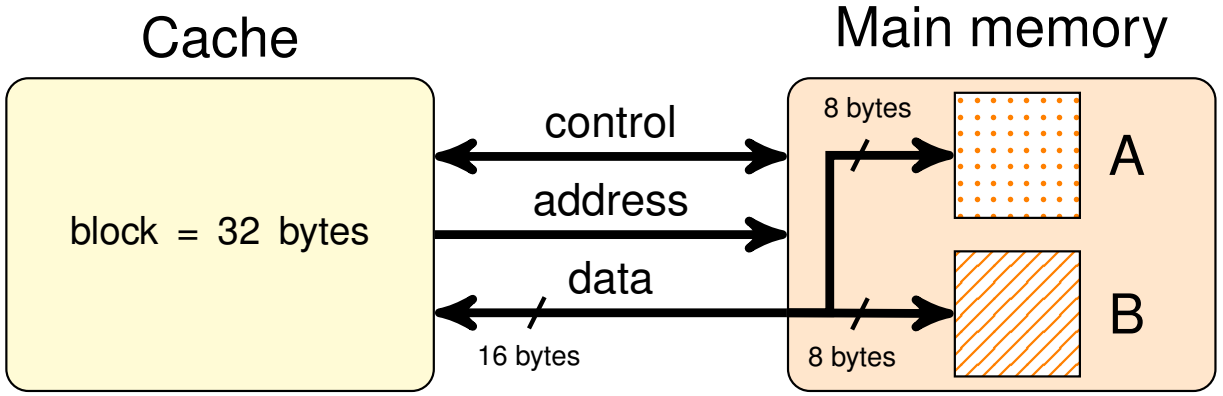
Interleaved mem., expanded bus, and burst mode

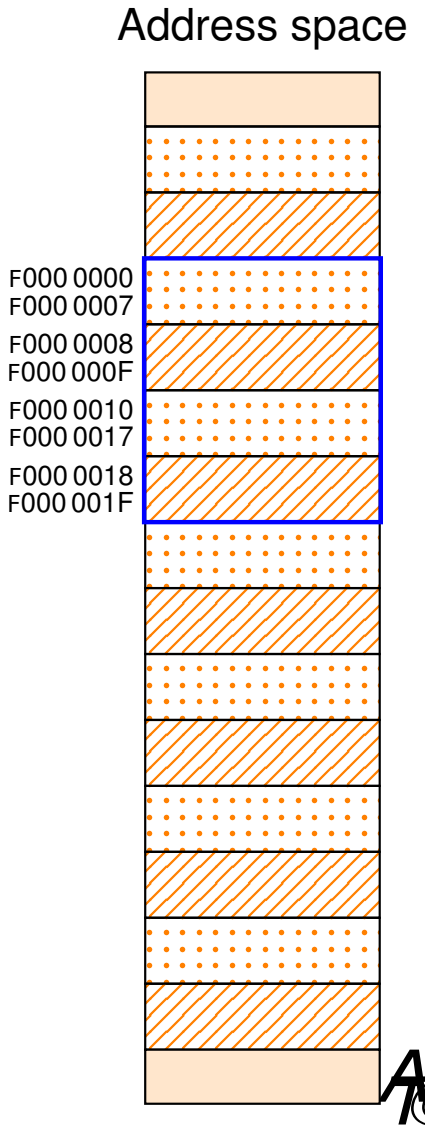


OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle



Interleaved mem., expanded bus, and burst mode





OP1 = 1 clock cycle; OP2 = 3 clock cycles; OP3 = 1 clock cycle

$$\text{Miss penalty} = \underbrace{2 \times (1/2 + 3/2 + 1/2)}_{1^{st}\text{ access}} + \underbrace{(32/8 - 2) \times (0 + 3/2 + 1/2)}_{\text{rest accesses}}$$

$$= 9 \text{ clock cycles}$$