

Chapter 10 Worked Examples

1 Cache Hits & Misses

1.1 Direct-Mapped Cache

A notional system has direct-mapped cache that is addressed via 16-bit addresses. The cache uses 8-byte blocks and has 8 total entries.

Direct-Mapped Cache; 8-byte blocks, 8 blocks										
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0	1DA	1	01	23	45	67	89	AB	CD	EF
1	345	1	F1	E2	D3	C4	B5	A6	97	80
2	2B7	0	–	–	–	–	–	–	–	–
3	09C	1	17	39	28	13	46	79	55	00
4	18E	0	–	–	–	–	–	–	–	–
5	2F6	1	11	22	33	44	55	66	77	88
6	3A4	1	99	00	FF	EE	DD	CC	BB	AA
7	06B	1	02	31	06	85	88	00	00	00

How many bits for the block offset? 3

How many bits for the set index? 3

How many bits for the tag? 10

	15	...	6	5	...	3	2	...	0
address	tag			set index			block offset		

For each of the examples, write address in binary, separate it into the three bitfields, and then re-express those bitfields in hexadecimal to use with the cache tables.

- Given an address of 0x7687

0b 0111 0110 1000 0111

0b 01 1101 1010 000 111

0x 1DA 0 7

- What is the block offset? 7
- What is the set index? 0
- What is the tag? 0x1DA

- Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0xEF
2. Given an address of 0x1D4E
- What is the block offset? 6
 - What is the set index? 1
 - What is the tag? 0x075
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
3. Given an address of 0x7215
- What is the block offset? 5
 - What is the set index? 2
 - What is the tag? 0x1C8
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
4. Given an address of 0x271C
- What is the block offset? 4
 - What is the set index? 3
 - What is the tag? 0x09C
 - Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0x46
5. Given an address of 0x3FA3
- What is the block offset? 3
 - What is the set index? 4
 - What is the tag? 0x0FE
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
6. Given an address of 0xBDAA
- What is the block offset? 2
 - What is the set index? 5
 - What is the tag? 0x2F6
 - Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0x33
7. Given an address of 0xE931

- What is the block offset? 1
- What is the set index? 6
- What is the tag? 0x3A4
- Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0x00

8. Given an address of 0xD638

- What is the block offset? 0
- What is the set index? 7
- What is the tag? 358
- Do we have a cache hit? If so, what is the value of the byte at that address? no

1.2 2-Way Set-Associative Cache

A notional system has a 2-way set associative cahce that is addressed via 16-bit addresses. The cache uses 8-byte blocks and has 8 total entries.

2-Way Set-Associative Cache; 8-byte blocks, 8 blocks										
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0	123	1	01	23	45	67	89	AB	CD	EF
	4A5	1	F1	E2	D3	C4	B5	A6	97	80
1	6B7	0	–	–	–	–	–	–	–	–
	69C	1	17	39	28	13	46	79	55	00
2	0FE	0	–	–	–	–	–	–	–	–
	1D2	1	11	22	33	44	55	66	77	88
3	3A4	1	99	00	FF	EE	DD	CC	BB	AA
	58B	1	02	31	06	85	88	00	00	00

How many bits for the block offset? 3

How many bits for the set index? 2

How many bits for the tag? 11

	15	...	5	4	3	2	...	0
address	tag			set index		block offset		

For each of the examples, write address in binary, separate it into the three bitfields, and then re-express those bitfields in hexadecimal to use with the cache tables.

1. Given an address of 0x2460

0b 0010 0100 0110 0000

0b 001 0010 0011 00 000

0x 123 0 0

- What is the block offset? 0

- What is the set index? 0
 - What is the tag? 0x123
 - Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0x01
2. Given an address of 0x98A1
- What is the block offset? 1
 - What is the set index? 0
 - What is the tag? 0x4C5
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
3. Given an address of 0xD6EA
- What is the block offset? 2
 - What is the set index? 1
 - What is the tag? 0x6B7
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
4. Given an address of 0x396B
- What is the block offset? 3
 - What is the set index? 1
 - What is the tag? 0x1CB
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
5. Given an address of 0x7F14
- What is the block offset? 4
 - What is the set index? 2
 - What is the tag? 0x3F8
 - Do we have a cache hit? If so, what is the value of the byte at that address? no
6. Given an address of 0x3A55
- What is the block offset? 5
 - What is the set index? 2
 - What is the tag? 0x1D2
 - Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0x66

7. Given an address of 0xB17E

- What is the block offset? 6
- What is the set index? 3
- What is the tag? 0x58B
- Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0x00

8. Given an address of 0x749F

- What is the block offset? 7
- What is the set index? 3
- What is the tag? 0x3A4
- Do we have a cache hit? If so, what is the value of the byte at that address? yes, 0xAA

Copy of notional caches for overhead projector or for handing out

A notional system has direct-mapped cache that is addressed via 16-bit addresses. The cache uses 8-byte blocks and has 8 total entries.

Direct-Mapped Cache; 8-byte blocks, 8 blocks										
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0	1DA	1	01	23	45	67	89	AB	CD	EF
1	345	1	F1	E2	D3	C4	B5	A6	97	80
2	2B7	0	–	–	–	–	–	–	–	–
3	09C	1	17	39	28	13	46	79	55	00
4	18E	0	–	–	–	–	–	–	–	–
5	2F6	1	11	22	33	44	55	66	77	88
6	3A4	1	99	00	FF	EE	DD	CC	BB	AA
7	06B	1	02	31	06	85	88	00	00	00

A notional system has a 2-way set associative cache that is addressed via 16-bit addresses. The cache uses 8-byte blocks and has 8 total entries.

2-Way Set-Associative Cache; 8-byte blocks, 8 blocks										
Index	Tag	Valid	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0	123	1	01	23	45	67	89	AB	CD	EF
	4A5	1	F1	E2	D3	C4	B5	A6	97	80
1	6B7	0	–	–	–	–	–	–	–	–
	69C	1	17	39	28	13	46	79	55	00
2	0FE	0	–	–	–	–	–	–	–	–
	1D2	1	11	22	33	44	55	66	77	88
3	3A4	1	99	00	FF	EE	DD	CC	BB	AA
	58B	1	02	31	06	85	88	00	00	00