

$t_0$  = Current Input data addr

$t_1$  = Current Output-data addr

$s_0$  = Start addr of Output-data addr

$t_3$  = Current Output-data addr at Validation

$t_4$  = Before Output-data addr at validation

Cache miss

Cache hit

## Inserting 2

IC : lw \$0, 0(\$t0) Now pointing Input-data[0]

t0 = 0x400 index = 0

... 0100 0000 0000

|   | Valid | Tag  | Data              |
|---|-------|------|-------------------|
| 3 | 0     |      |                   |
| 2 | 0     |      |                   |
| 1 | 0     |      |                   |
| 0 | 1     | 0x10 | Input-data[0] ~ 3 |

## Inserting 0

IC : lw \$0, 0(\$t0) Input-data[1]

t0 = 0x404 Index = 0

... 0000 0000 0100

|   | Valid | Tag  | Data              |
|---|-------|------|-------------------|
| 3 | 0     |      |                   |
| 2 | 0     |      |                   |
| 1 | 0     |      |                   |
| 0 | 1     | 0x10 | Input-data[0] ~ 3 |

40 : lw \$0, 0(\$t1)

t1 = 0x40D (Output-data)

... 0100 1000 0000

|   | Valid | Tag  | Data               |
|---|-------|------|--------------------|
| 3 | 0     |      |                    |
| 2 | 0     |      |                    |
| 1 | 0     |      |                    |
| 0 | 1     | 0x12 | Output-data[0] ~ 3 |

~~41: lw \$1, 0(\$3)~~

↑ pointing `Output-data[1]`

$t_3 = 0x4844$  (`Output-data + 4`)

... 0100 1000 0100

|   | Valid | Tag  | Data              |
|---|-------|------|-------------------|
| 3 | 0     |      |                   |
| 2 | 0     |      |                   |
| 1 | 0     |      |                   |
| 0 | 1     | 0x12 | Output-data 0 ~ 3 |

Inserting -1

`Input-data[2]`

~~42: lw \$0, 0(\$0)~~

$t_0 = 0x408$  index=0

... 0100 0000 1000

|   | Valid | Tag  | Data             |
|---|-------|------|------------------|
| 3 | 0     |      |                  |
| 2 | 0     |      |                  |
| 1 | 0     |      |                  |
| 0 | 1     | 0x10 | Input-data 0 ~ 3 |

~~43: lw \$0, 0(\$4)~~

`Output-data[1]`

$t_4 = 0x4844$

|   | Valid | Tag  | Data              |
|---|-------|------|-------------------|
| 3 | 0     |      |                   |
| 2 | 0     |      |                   |
| 1 | 0     |      |                   |
| 0 | 1     | 0x12 | Output-data 0 ~ 3 |

44: ~~lw \$1, 0(\$t3)~~  
Output-data[2]

$$t3 = \text{0x4ff}$$

|   | Valid | Tag  | Data             |
|---|-------|------|------------------|
| 3 | 0     |      |                  |
| 2 | 0     |      |                  |
| 1 | 0     |      |                  |
| 0 | 1     | 0x12 | Output-data 0 ~3 |

Insert -1 loop

Input-data[3]  
~~lw \$0, 0(\$t0)~~

$$t0 = \text{0x40C} \quad \text{index=0}$$

... 0100 0000 1100

|   | Valid | Tag  | Data            |
|---|-------|------|-----------------|
| 3 | 0     |      |                 |
| 2 | 0     |      |                 |
| 1 | 0     |      |                 |
| 0 | 1     | 0x10 | Input-data 0 ~3 |

45: ~~lw \$0, 0(\$t4)~~  
Output-data[2]

$$t4 = \text{0x4ff}$$

|   | Valid | Tag  | Data             |
|---|-------|------|------------------|
| 3 | 0     |      |                  |
| 2 | 0     |      |                  |
| 1 | 0     |      |                  |
| 0 | 1     | 0x12 | Output-data on 3 |

Act: [W S1, 0(16)]

Output-data[3]

t<sub>0</sub> = 0x4fc

| Valid | Tag  | Data             |
|-------|------|------------------|
| 0     |      |                  |
| 0     |      |                  |
| 0     |      |                  |
| 1     | 0x12 | Output_data 0 ~3 |

Act: [W S0, 0(16)]

Output-data[1]

| Valid | Tag  | Data             |
|-------|------|------------------|
| 0     |      |                  |
| 0     |      |                  |
| 0     |      |                  |
| 1     | 0x12 | Output_data 0 ~3 |

Act: [W S1, 0(16)]

Output-data[2]

| Valid | Tag  | Data             |
|-------|------|------------------|
| 0     |      |                  |
| 0     |      |                  |
| 0     |      |                  |
| 1     | 0x12 | Output_data 0 ~3 |

## Insert Sort

주연원 Sort 알고리즘에는 Cache에 사용된다

① Input에서 대상에 속하는지를 찾았던 Input-data를 찾는다

②, ③ Output-data가 삽입된 이후 어떤 위치와 원래 위치의 값은 바뀌어야 한다

Output-data의 값을 찾는다

Cache 밀리터가 16B의 4개의 entry로 되어있고 각 entry는 Input 또는 Output data의 주소 값을 저장할 수 있다.

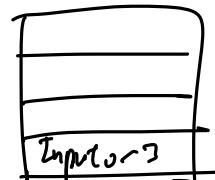
① 다음 ②, ③이 대상으로 되는지, (이전 값과 현재 값이 swap 되면 ②, ③을 한 번 더 봄) ① → ②의 결과에서 일관적으로 miss가 발생한다. ①과 ③의 index가 동일하므로 ②~③이 반복되는 과정에서는 4개 단위의 블록을 넘어서도 같은지만 가능하다. 또한 ②~③이 반복되는 과정에서는 4개 단위의 블록을 넘어서도 같은지만 가능하다. 각 블록의 인덱스가 4단위로 증가하는 점을 유형화하여 sort 대상을 따라가면서 아래와 같다.

| Input-data (15~16)              |
|---------------------------------|
| 2, 0, -1, -1, 3, 8, -4, 10      |
| -9, -16, 15, 13, 1, 4, -3, 14   |
| -8, -10, -15, 6, -13, -5, 9, 12 |
| -11, -14, -6, 11, 5, 7, -2, -12 |

1. Insert 2

Cache Empty → Tko ← Input 0~3 load

2



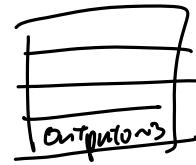
## 2. Insert 0

Find Input 1 at cache Index 0 (0)

load Output ans for validation at Index 0 (2)

Find Output 1 at cache Index 0

(0) 2 0



## 3. Insert -1

load Input ans at cache Index 0 (-1)

load Output ans for validation at Index 0 (0)

Find Output 2 at cache Index 0

(-1) 2 0 -1



## 4. Insert -1

load Input ans at Index 0

load Output ans for validation at Index 0

Find Output 0 at cache Index 0

swap!

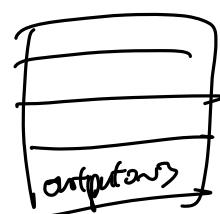
2 0 -1 -1

Find Output 1 at cache Index 0

2 0 -1 -1

Find Output 2 at cache Index 0

2 0 -1 -1



## 5. Insert 3

load Input 4 to 1 at Index 1

2 0 -1 -1 3

Find Output 3 at Index 0

2 0 -1 -1 3

load Output 4 to 1 at Index 1

2 0 -1 3 -1

swap

Find Output 2 at Index 0

2 0 -1 3 -1

Find Output 3 at Index 0

2 0 3 -1 -1

swap

Find Output 1 at Index 0

2 0 3 -1 -1

Find Output 2 at Index 0

2 3 0 -1 -1

swap

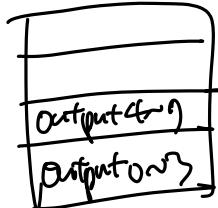
Find Output 0 at  $\text{Idx} 0$

Find Output 1 at  $\text{Idx} 0$

swap

2 3 0 -1 -1

3 2 0 -1 -1



## 6. Insert 8

load Input 4~7 at  $\text{Idx} 1$

3 2 0 -1 -1 8

load Output 4~7 at  $\text{Idx} 1$

3 2 0 -1 -1 8

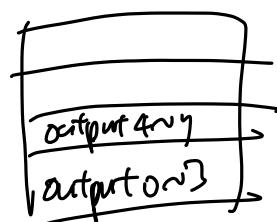
Find Output 5

3 2 0 -1 8 -1

swap  
↓

5 more swap  $\Rightarrow$  10 hits

8 3 2 0 -1 -1



## 7. Insert -4

load Input 4~7 at  $\text{Idx} 1$  8 3 2 0 -1 -1 -4

load Output 4~7 at  $\text{Idx} 1$

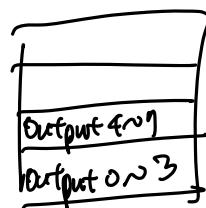
8 3 2 0 -1 -1 -4

Find Output 6

swap

2 hits for comparing

8 3 2 0 -1 -4 -1



## 8. Insert 10

load Input 4~7 at  $\text{Idx} 1$

8 3 2 0 -1 -4 -1 10

load Output 4~7 at  $\text{Idx} 1$

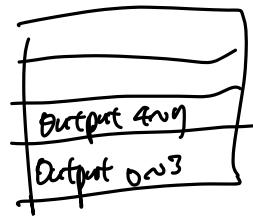
8 3 2 0 -1 -4 -1 10

Find Output 1

Swap

6 swaps  $\Rightarrow$  12 hits

10 8 3 2 0 -1 -4 -9



9. Insert -9

load Input 8w11 at Idx 2 10 8 3 2 0 -1 -4 -9 -9

Find Output 1

load Output 8w11 at Idx 2

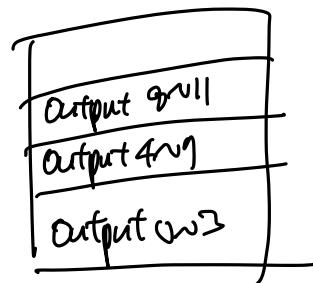


10. Insert -16

load Input 8w11 at Idx 2 10 8 3 2 0 -1 -4 -9 -9 -16

load Output 8w11 at Idx 2

Find Output 9



11. Insert 15

load Input 8w11 at Idx 2 10 8 3 2 0 -1 -4 -9 -9 -16 15

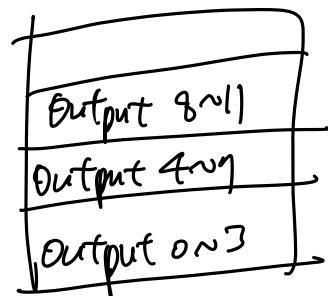
load Output 8w11 at Idx 2

Find Output 10

swap

g S

g swap  $\Rightarrow$  18 hits



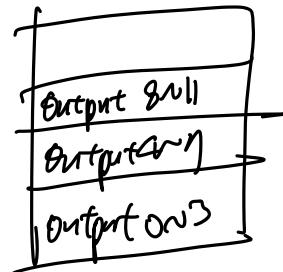
## 12. Insert 13

~~load Input 8null at Idx2~~ 15 10 8 3 2 0 -1 -4 -9 -9 -16 13  
~~load Output 8null at Idx2~~  
 Find Output 11

Swap

S

Swap  $\Rightarrow$  18 hits  
 +2 hits (compare with 15)



## 13. Insert 1

~~load Input 12~15 at Idx3~~ 15 13 10 9 3 2 0 -1 -4 -9 -9 -16 1

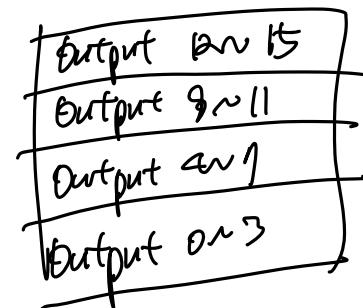
Find Output 11

~~load Output 12~15 at Idx3~~

Swap

S

Swap  $\Rightarrow$  10 hits  
 +2 hit



## 14. Insert 4

~~load Input 12~15 at Idx3~~

15 13 10 9 3 2 1 0 -1 -4 -9 -9 -16 4

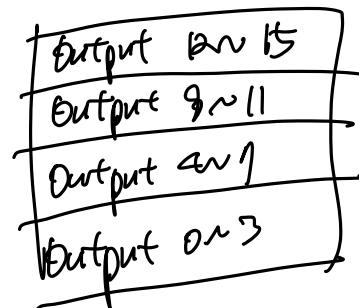
~~load Output 12~15 at Idx3~~

Find Output 13

Swap

S

Swap  $\Rightarrow$  16 hit  
 +2 hit



### 15. Insert -3

~~load Input 12~15 at Idx3~~

15 13 10 8 4 3 2 1 0 -1 -3 -4 -7 -9 -16 -3

~~load Output 12~15 at Idx3~~

Find Output 14

swap

S

| 3 swap  $\Rightarrow$  6 hit  
+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 9~11  |
| Output 4~7   |
| Output 0~3   |

### 16. Insert 14

~~load Input 12~15 at Idx3~~

15 13 10 9 4 3 2 1 0 -1 -3 -4 -7 -9 -16 14

~~load Output 12~15 at Idx3~~

Find Output 15

swap

S

| 3 swap  $\Rightarrow$  26 hit  
+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 9~11  |
| Output 4~7   |
| Output 0~3   |

### 17. Insert -9

~~load Input 16~19 at Idx0~~

15 14 13 10 / 9 4 3 2 / 1 0 -1 -3 / -4 -7 -9 -16 / 8

Find Output 15

~~load Output 16~19 at Idx0~~

swap

S

| swap  $\Rightarrow$  2 hit  
+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 9~11  |
| Output 4~7   |
| Output 16~19 |

### (8). Insert -10

load Input 16~19 at Index 0    15 14 13 10 / 8 4 3 2 / 10 -1-3 / -4 -1 -9 -9 / -16 -10

load Output 16~19 at Index 0

Find Output 19

Swap

+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 4~7   |
| Output 16~19 |

### (9). Insert -15

load Input 16~19 at Index 0    15 14 13 10 / 8 4 3 2 / 10 -1-3 / -4 -1 -9 -9 / -10 -16 -15

load Output 16~19 at Index 0

Find Output 18

swap

+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 4~7   |
| Output 16~19 |

### (10). Insert 6

load Input 16~19 at Index 0    15 14 13 10 / 8 4 3 2 / 10 -1-3 / -4 -1 -9 -9 / -10 -15 -16 6

load Output 16~19 at Index 0

Find Output 19

Swap

5

| 3 Swap  $\Rightarrow$  26 hit

+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 4~7   |
| Output 16~19 |

## 21. Insert -3

~~load Input 20~23 at idx1~~

15 14 13 10 / 8 6 4 3 / 2 1 0 -1 / -3 -4 -7 -8  
-9 -10 -15 -16 -17

Find Output 19

~~load Output 20~23 at idx1~~

swap

{ Swap  $\Rightarrow$  2 hit

+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 20~23 |
| Output 16~19 |

## 22. Insert -5

~~load Input 20~23 at idx1~~

15 14 13 10 / 8 6 4 3 / 2 1 0 -1 / -3 -4 -7 -8

~~load Output 20~23 at idx1~~

-9 -10 -13 -15 / -16 -5

Find Output 21

swap

{ Swap  $\Rightarrow$  12 hit

+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 20~23 |
| Output 16~19 |

## 23. Insert 9

~~load Input 20~23 at idx1~~

unloaded  
15 14 13 10 / 8 6 4 3 / 2 1 0 -1 / -3 -4 -5 -7  
-8 -9 -10 -13 / -15 -16 9

~~load Output 20~23 at idx1~~

Find Output 22

swap

{3 Swap  $\Rightarrow$  26 hit

~~load Output 4~7 at idx 1~~

Find Output 8

Swap

S

3 swap  $\Rightarrow$  6 hit

~~load Output 0~3 at idx 0~~

Find Output 4

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 4~7   |
| Output 0~3   |

## 24. Insert 12

~~load Input 20~23 at idx 1~~

~~load Output 20~23 at idx 1~~

Find Output 23

Swap

S

2 swap  $\Rightarrow$  4 hit

~~load Output 1b~1f at idx 0~~

Find Output 20

Swap

S

1 swap  $\Rightarrow$  22 hit

~~load Output 4~7 at idx 1~~

15 14 13 10 / 9 8 6 4 / 3 2 1 = / -1 -3 -4 -5  
-7 -8 -9 -10 / -13 -15 -16 12

swap

S

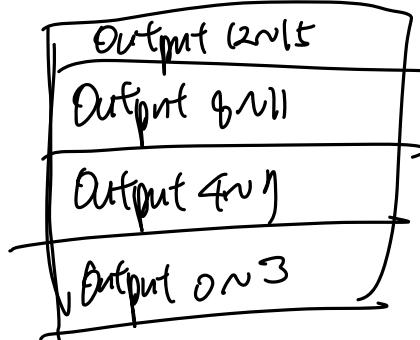
3 swap  $\Rightarrow$  6 hit

~~load Output 0~3 at idx 0~~

swap

Find Output 2

Find Output 3



25. Insert -11

~~load Input 24~21 at idx 2~~

~~load Output 20~23 at idx 1~~

~~load Output 24~21 at idx 2~~

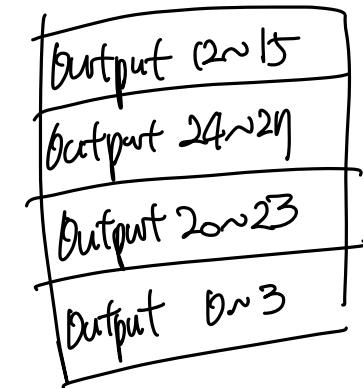
Swap

S

2 swap  $\Rightarrow$  4 hit

+2 hit

15 14 13 12 / 10 9 8 6 | 4 3 2 1 / 0 -1 -3 -4  
 -5 -7 -8 -9 / -10 -11 -13 -15 / -16 -14



26. Insert -14

~~load Input 24~21 at idx 2~~

~~load Output 24~21 at idx 2~~

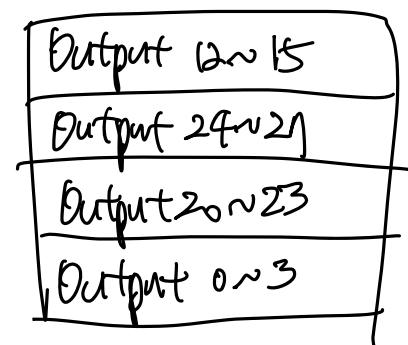
Find Output 25

Swap

S

( swap  $\Rightarrow$  2 hit ) +2 hit

15 14 13 12 / 10 9 8 6 | 4 3 2 1 / 0 -1 -3 -4  
 -5 -7 -8 -9 / -10 -11 -13 -15 / -16 -14



27. Insert -6

load Input 24~21 at idx2 15 14 13 12 / 10 9 8 6 / 4 3 2 1 / 0 -1 -3 -4

load Output 24~21 at idx2 -5 -7 -8 -9 / -10 -11 -13 -14 / -15 -16 -6

Find Output 26

swap

↓

5 swap  $\Rightarrow$  10 hit

load Output 16~19 at idx0

Find Output 20

swap

↓

2 swap  $\Rightarrow$  4 hit

+2 hit

|              |
|--------------|
| Output 12~15 |
| Output 24~21 |
| Output 20~23 |
| Output 16~19 |

28. Insert 11

load Input 24~21 at idx2 15 14 13 12 / 10 9 8 6 / 4 3 2 1 / 0 -1 -3 -4  
-5 -6 -7 -8 / -9 -10 -11 -12 / -14 -15 -16 11

load Output 24~21 at idx2

Find Output 21

swap

↓

14 swap  $\Rightarrow$  28 hit

~~load Output 8~11 at idx 2~~

Find Output 12

swap

↓

3 swap  $\Rightarrow$  blif

~~load Output 4~7 at idx 1~~

Find Output 8

swap

↓

3 swap  $\Rightarrow$  blif

~~load Output 0~3 at idx 0~~

Find Output 4

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 4~7   |
| Output 0~3   |

29, Insert 5

~~load Input 28~31 at idx 3~~

15 14 13 12 / 11 10 9 8 / 6 4 3 2 ) 10 -1 -3  
-4 -5 -6 -7 / -8 -9 -10 -11 / -13 -14 -15 -16 5

~~load Output 24~27 at idx 2~~

~~load Output 28~31 at idx 3~~

swap

↓

3 swap  $\Rightarrow$  blif

~~load Output 20~23 at idx 1~~

Find Output 24

swap

↓

3 swap  $\Rightarrow$  blif

~~load Output 1b~19 atIdx 0~~

Find Output 20

swap  
S

3 swap  $\Rightarrow$  shift

~~load Output 12~15 atIdx 3~~

Find Output 16

swap  
S

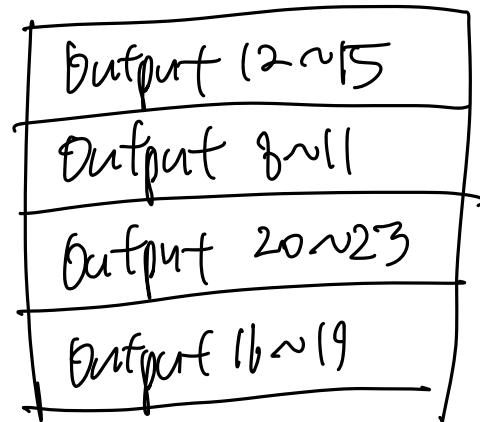
3 swap  $\Rightarrow$  shift

~~load Output 8~11 atIdx 2~~

Find Output 12

swap  
S

2 swap  $\Rightarrow$  4 hif  
 $\rightarrow$  2 hif



### 30. Insert ↗

~~load Input 26~31 atIdx 3~~ 15 14 13 12 11 10 9 8 | 6 5 4 3 / 2 1 ~ - 1

~~load Output 26~31 atIdx 3~~ 3 - 4 - 5 - 6 | - 7 - 8 - 9 - 10 / - 11 - 12 - 13 - 14 - 15 / - 16 ↗

Find Output 29

swap

~~load Output 24~29 atIdx 2~~

Find Output 28

swap

5

1 swap  $\Rightarrow$  2 hit

load Output 12~15 at idx 3

Find Output 6

swap

5

2 swap  $\Rightarrow$  6 hit

load Output 8~11 at idx 2

Find Output 12

swap

5

3 swap  $\Rightarrow$  6 hit

load Output 4~7 at idx 1

Find Output 8

|              |
|--------------|
| Output 12~15 |
| Output 8~11  |
| Output 4~7   |
| Output 6~9   |

### 31. Insert -2

load Input 28~31 at idx 3 15 14 13 12/11 10 9 8/7 6 5 4 / 3 2 1 0

load Output 28~31 at idx 3 -1 -3 -4 -5 / -6 -7 -8 -9 / -10 -11 -13 -14 -15 -6 -2

Find Output 30

swap

5

1 swap  $\Rightarrow$  2 hit

load Output 26~27 at idx 2

Find Output 28

swap

S

3 swap  $\Rightarrow$  6 hit

~~load Output 20~23 at idx 1~~

Find Output 24

swap

S

6 swap  $\Rightarrow$  12 hit

+2 hit

|              |
|--------------|
| Output 25~31 |
| Output 24~29 |
| Output 20~23 |
| Output 16~19 |

### 32. Insert -12

~~load Input 24~31 at idx 3~~

— / — / — / —

~~load Output 25~31 at idx 3~~

— / — / -9 -10 +1 -13 / -14 -15 -16 -17

Find Output 31

swap

S

3 swap  $\Rightarrow$  6 hit

+2 hit

|              |
|--------------|
| Output 25~31 |
| Output 24~29 |
| Output 20~23 |
| Output 16~19 |

Sorting End!

a. how many memory blocks are replaced?

83 cache memory block

b. final hit ratio

hit ratio = hit request / Entire request

= hit request / hit request + miss request

= 464 / 464 + 83

= 0.85

c. final state of cache

|   | Valid | tag  | Data |     |     |     |
|---|-------|------|------|-----|-----|-----|
| 3 | 1     | 0x13 | -13  | -14 | -15 | -16 |
| 2 | 1     | 0x13 | -9   | -10 | -11 | -12 |
| 1 | 1     | 0x13 | -5   | -6  | -7  | -8  |
| 0 | 1     | 0x13 | -1   | -2  | -3  | -4  |

← 0x400 /  $\frac{0100}{\text{tag}} \frac{1100}{\text{idx}} \frac{0000}{\text{offset}}$