Negative Ints Binary

Piotr Karamon

<2022-06-27 Mon>

Contents

1 Sign bit

Let's say we want to write (-5) in binary. 5 = 101
Sign can be written as a bit.

- 0 means +
- 1 means −

-5 in a 32bit int looks like this: 10000000000000000000000000000101
-5 in a 4bit system(3 bits are used for numbers, 1 bit for sign): 1101

positive	binary	negative	binary
0	0000	-0	1000
1	0001	-1	1001
2	0010	-2	1010
3	0011	-3	1011
4	0100	-4	1100
5	0101	-5	1101
6	0110	-6	1110
7	0111	-7	1111

1.1 Problems with sign bit

- we have two representations of zero (0000 and 1000)
- normal binary addition does not work

```
+5 + (-3) = +2
+5 = 0101
-3 = 1011
-2 = 0000
FALSE
```

2 One's Complement(1C)

- we still have the sign bit with the same properties in the first solution
- when we change the sign of a number we negate every bit. We turn every 0 into 1 and every 1 into 0
- tables

postive	binary	negative	binary
0	0000	-0	1111
1	0001	-1	1110
2	0010	-2	1101
3	0011	-3	1100
4	0100	-4	1011
5	0101	-5	1010
6	0110	-6	1001
7	0111	-7	1000

2.1 Worth noting points

- we still have two zeros (0000,1111)
- addition now works

```
+5 + (-3) = ?
+5 = 0101
-3 = 1100
     0101
     1100
```

```
    ----
(1)0001
      1
    ----
    0010 = 2
```

3 Two's complement(2C)

We still negate every bit, but now we also add 1 (0001)

```
-0 = 1111
        1
        0000 = 0
```

- in 2C we don't care about left over 1 in the last addition
- the same table but in 2C

postive	binary	negative	binary
0	0000	-8	1000
1	0001	-1	1111
2	0010	-2	1110
3	0011	-3	1101
4	0100	-4	1100
5	0101	-5	1011
6	0110	-6	1010
7	0111	-7	1001

our range is $[-8, 7]$

- 8 bit int range $[-128, 127]$
- 16 bit int range $[-32768, 32767]$
- 32 bit int range $[-2147483648, 2147483647]$
- addition

  ```
  +5 = 0101
  -3 = 1101
  +2 = 0010
  ```