

Chapter 3. Data Representation

March, 2016
Seungjae Baek

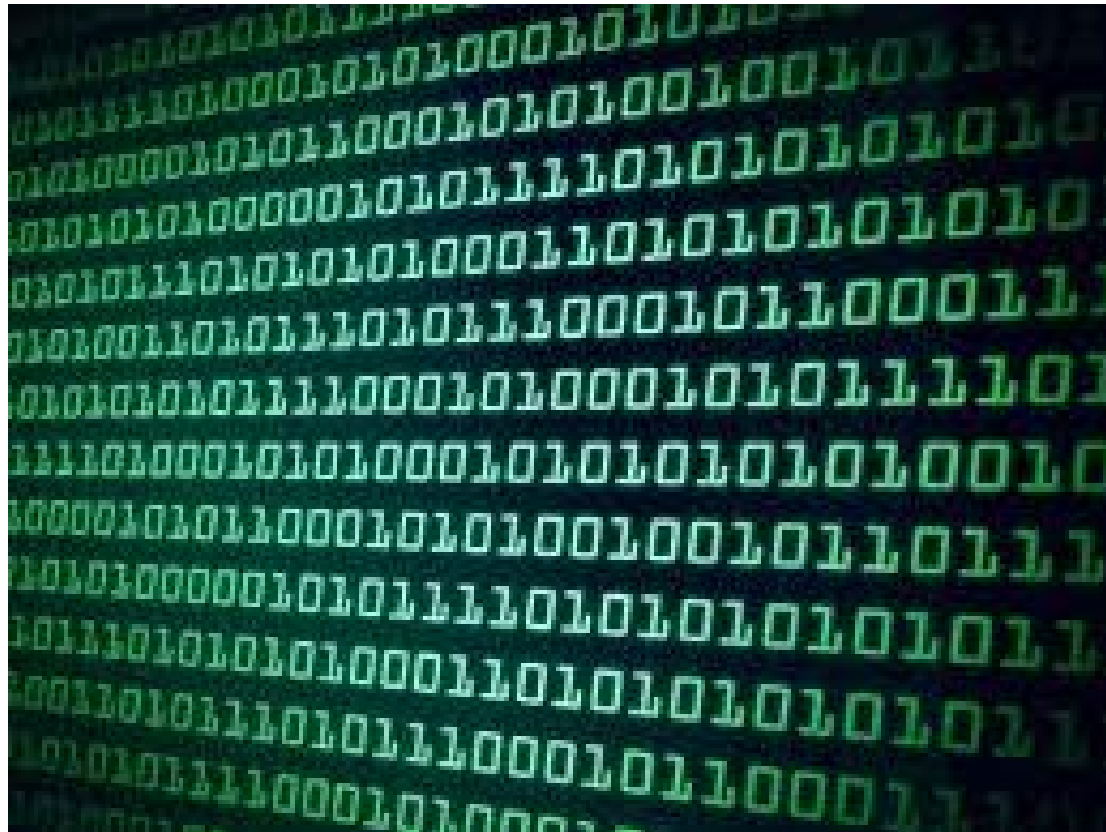
Dept. of software
Dankook University

<http://embedded.dankook.ac.kr/~baeksj>

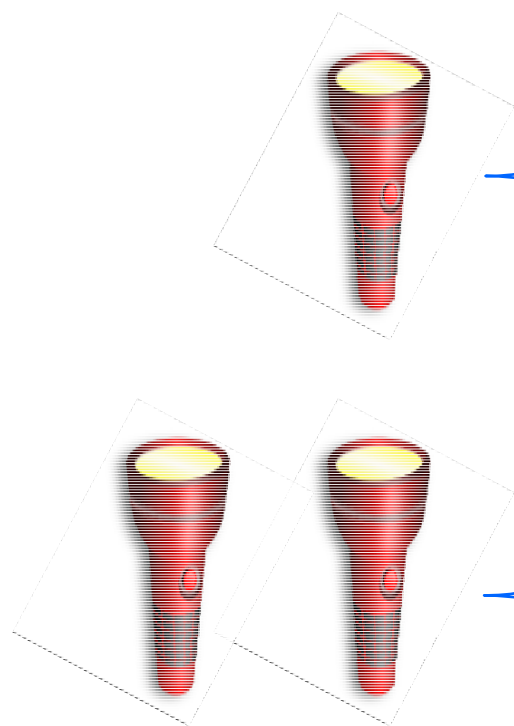
- 컴퓨터가 데이터를 표현하는 방식
- 정수와 실수의 표현 방식
- 비트 연산자

■ 2 진수 (Binary)

- ✓ 두 개의 기호를 이용해서 데이터를 표현하는 방식
- ✓ 컴퓨터에서는 내부 소자 특성상 2진수를 사용함



■ 2 진수 (Binary) 예



Turn on : 1
Turn off : 0

Turn on : 1 Turn on : 1
Turn on : 1 Turn off : 0
Turn off : 0 Turn on : 1
Turn off : 0 Turn off : 0

$2^1 = 2$

$2^2 = 4$

Q: 플래시 라이트의 개수가 **N** 이라면 표현할 수 있는 총 데이터 개수는 ?

데이터 표현 방식(3/7)

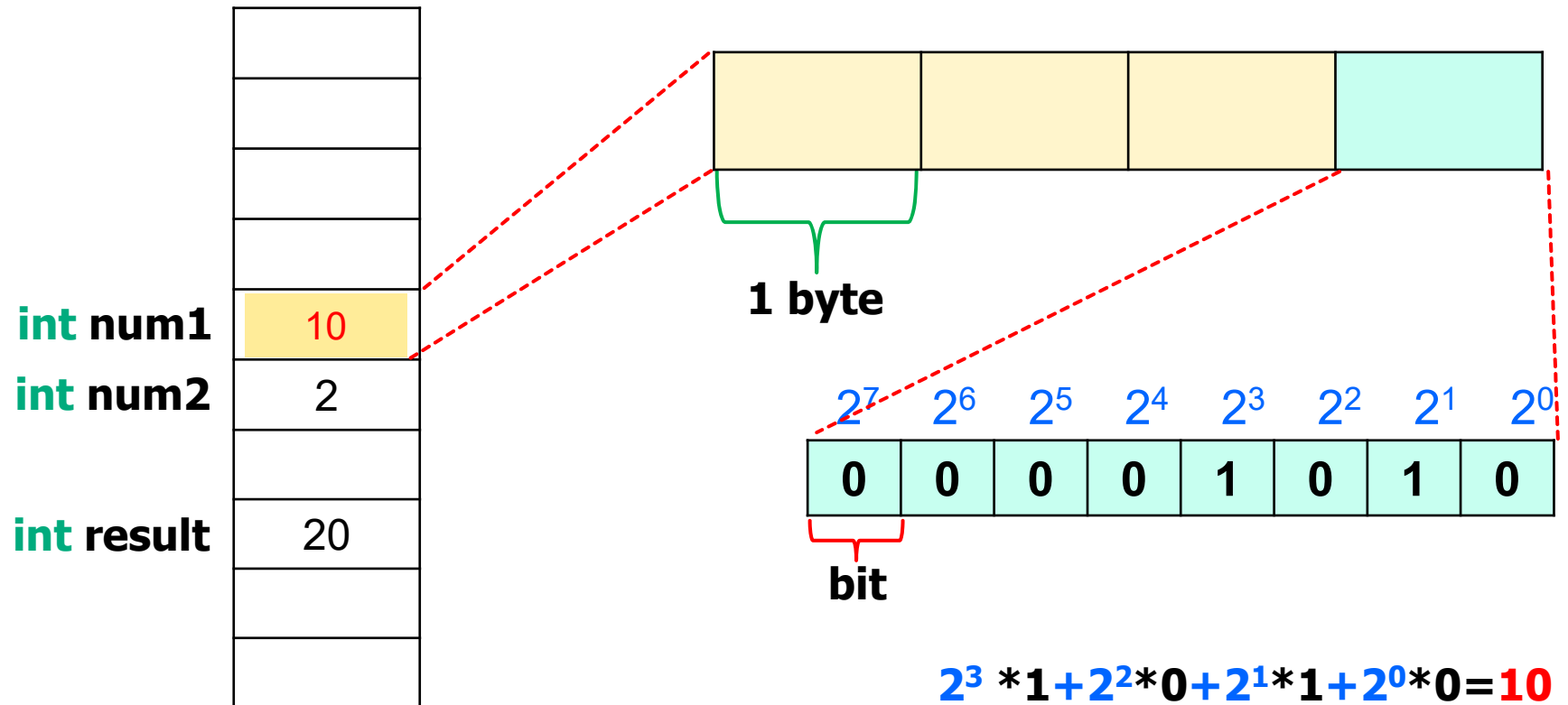
- 10진수(Decimal), 16진수 (Hexadecimal), N 진수
 - ✓ 10진수 – 열 개의 기호를 이용해서 데이터를 표현하는 방식
 - ✓ N진수 – N 개의 기호를 이용해서 데이터를 표현하는 방식

2진수	10진수	16진수
0	0	0
1	1	1
10	2	2
11	3	3
100	4	4
101	5	5
110	6	6
111	7	7
1000	8	8
1001	9	9
1010	10	A

2진수	10진수	16진수
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F
10000	16	10
10001	17	11

■ 비트 (bit)와 바이트 (byte)

- ✓ 비트 - 컴퓨터가 표현하는 최소의 단위 (2진수 값 하나 저장)
- ✓ 바이트 - 8개의 비트 묶음이 하나의 바이트



<물리 메모리>

Q: 1 byte 는 몇 bit 인가?

- C 언어의 진수 표현 (8진수, 16진수)
 - ✓ C 언어에서는 8진수, 10진수, 16진수 표현이 가능
 - ✓ 8진수 : 0으로 시작
 - ✓ 16진수 : 0x로 시작

진수 표현 예

```
int num1 = 10;  
int num2 = 0xA;  
int num3 = 012;  
int num4 = 0xA37B;
```

Q: num4를 2진수와 10진수로 변환한 값은?

■ C 언어의 진수 표현 예

```
#include <stdio.h>

int main(void)
{
    int num1=0xA7, num2=0x43;
    int num3=032, num4=024;

    printf("0xA7의 10진수 정수 값: %d \n", num1);
    printf("0x43의 10진수 정수 값: %d \n", num2);
    printf(" 032의 10진수 정수 값: %d \n", num3);
    printf(" 024의 10진수 정수 값: %d \n", num4);
    printf("%d-%d=%d \n", num1, num2, num1-num2);
    printf("%d+%d=%d \n", num3, num4, num3+num4);
    return 0;
}
```


■ 진수 변환 문제

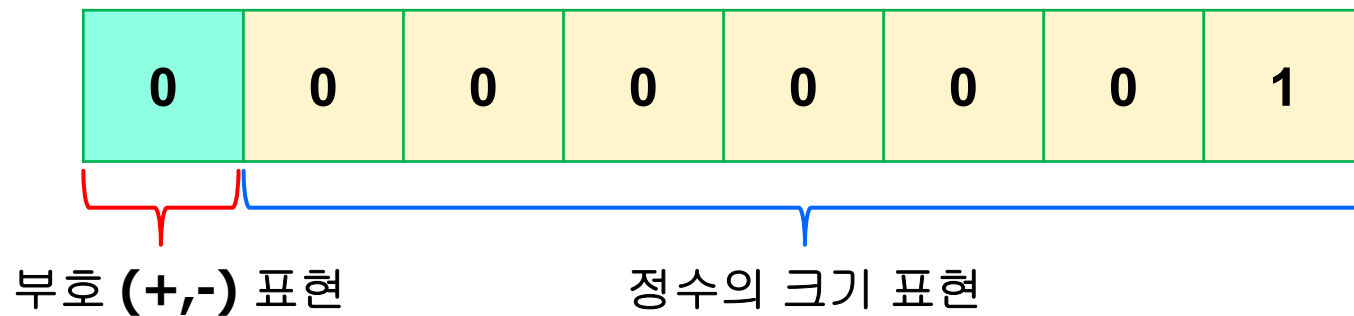
- ✓ num1~num4를 2진수와 8진수로 변환하시오.

진수 표현 예

```
int num1 = 0xB812C;  
int num2 = 0xDA984E;  
int num3 = 0563;  
int num4 = 0214;
```

■ 정수의 표현 방식

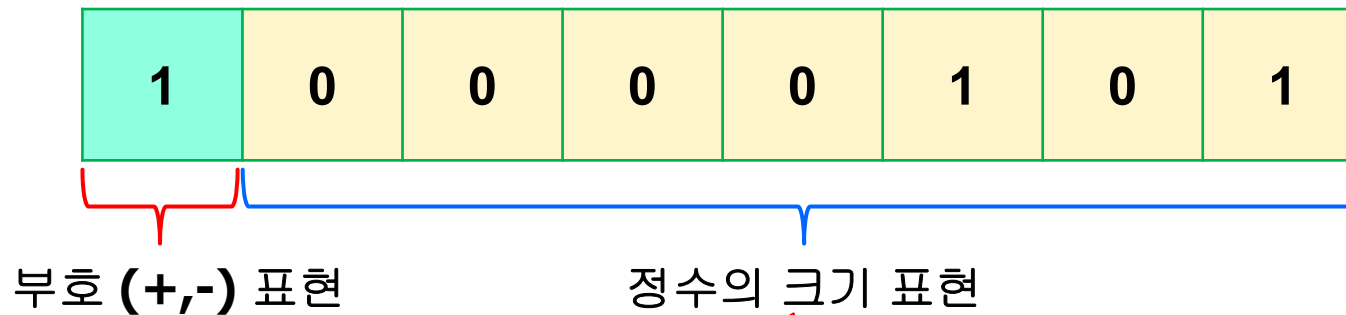
- ✓ 기본적으로 C 언어에서 정수는 4 바이트로 표현
- ✓ 설명을 위해 1바이트 사용
- ✓ 양수와 음수 표현 방식은 다름



■ 부호 비트

- ✓ 왼쪽의 최 상위 비트로 정수의 부호를 표시
- ✓ MSB (Most Significant Bit)
- ✓ 0 (양수), 1 (음수)

- 정수의 표현 방식
 - ✓ 음수의 표현 (틀린 경우)



Step 1 : $2^2 * 1 + 2^0 * 1 = 5$

Step 2 : -5

**Q: 10000101 이 -5가 맞는가?
검증 방법은?**

■ 정수의 표현 방식

✓ 검증 : $5 + (-5) = 0$

5

0	0	0	0	0	1	0	1
---	---	---	---	---	---	---	---

+

-5

1	0	0	0	0	1	0	1
---	---	---	---	---	---	---	---

-10

1	0	0	0	1	0	1	0
---	---	---	---	---	---	---	---

- 정수의 표현 방식
 - ✓ 음수의 표현 (맞는 표현)
 - 2의 보수법

5 0 0 0 0 0 1 0 1

Step 1: 보수 구하기

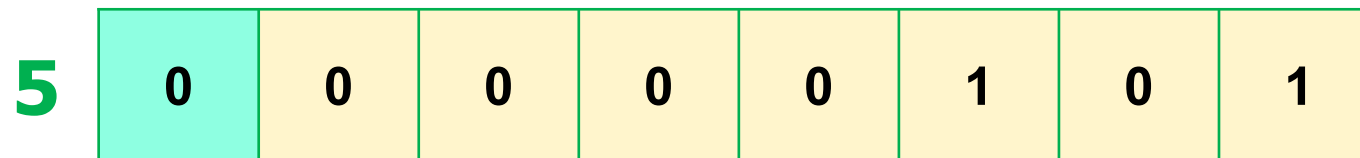
1 1 1 1 1 0 1 0

Step 2: 보수에 1 더하기

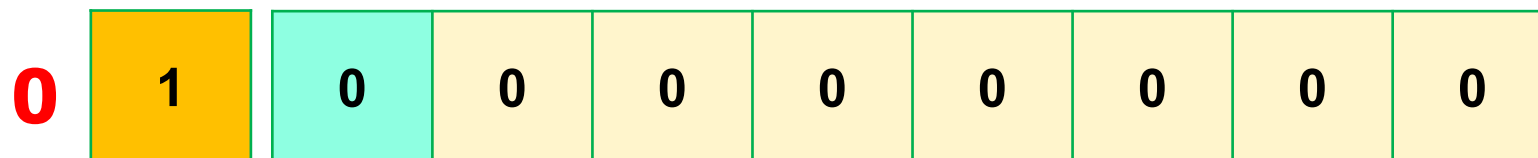
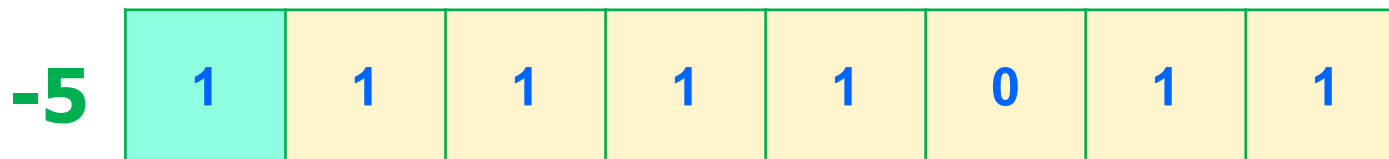
-5 1 1 1 1 1 0 1 1

■ 정수의 표현 방식

✓ 검증 : $5 + (-5) = 0$



+



■ 정수 예제

- ✓ 1) num1, num2 의 음의 2진수를 구하시오

변수 예

```
int num1 = 31;  
int num2 = 17;
```

- ✓ 2) 각 2 진수를 10진수로 변환하시오.

2진수 예

```
// 음의 2진수 11111011;  
// 음의 2진수 11011001;
```


■ 부동 소수점의 오차 예

```
#include <stdio.h>
int main(void)
{
    int i;
    float num=0.0;

    for(i=0; i<100; i++)
        num+=0.1;

    printf("0.1을 100번 더한 결과: %f \n", num);
    return 0;
}
```

Results

0.1을 100번 더한 결과: 10.000002

■ 비트 연산자 (Bitwise Operators)

✓ A = 60 (0011 1100)

✓ B = 13 (0000 1101)

연산자	설명	예
&	비트 단위 AND	(A&B)=12, 0000 1100
	비트 단위 OR	(A B)=61, 0011 1101
^	비트 단위 XOR	(A^B)=49, 0011 0001
~	모든 비트를 반전 (단항 연산자)	~A=-61, 1100 0011
<<	왼쪽 피연산자를 왼쪽으로 비트 이동	A<<2=240, 1111 0000
>>	왼쪽 피연산자를 오른쪽으로 비트 이동	A>>2=15, 0000 1111

Tip: 왼쪽 **shift**는 **1**비트 이동할 때 정수 값의 배수로 증가
오른쪽 **shift**는 **1**비트 이동할 때 정수 값 나누기 **2**로 감소

■ 비트 연산자 예

```
#include <stdio.h>

int main(void)
{
    int num1 = 15;
    int num2 = 20;
    int num3 = num1 & num2;

    printf("AND 연산의 결과: %d \n", num3);
    return 0;
}
```

Results

AND 연산의 결과: 4

Q: $A|B$, A^B , $\sim A$ 의 결과는 무엇인가?

■ 비트 연산자 예

```
#include <stdio.h>
int main(void)
{
    int num=15;
    int result1 = num<<1;
    int result2 = num<<2;
    int result3 = num<<3;
    printf("1칸 이동 결과: %d \n", result1);
    printf("2칸 이동 결과: %d \n", result2);
    printf("3칸 이동 결과: %d \n", result3);
    return 0;
}
```

Results

```
1칸 이동 결과: 30
2칸 이동 결과: 60
3칸 이동 결과: 120
```

Q: num을 오른쪽으로 shift 하면?

■ 비트 연산자 예

```
#include <stdio.h>
int main(void)
{
    int num=-16
    printf("2칸 오른쪽 이동의 결과: %d \n", num>>2);
    printf("3칸 오른쪽 이동의 결과: %d \n", num>>3);
    return 0;
}
```

Results

```
2칸 오른쪽 이동의 결과: -4
3칸 오른쪽 이동의 결과: -2
```

- 컴퓨터가 데이터를 표현하는 방식
 - ✓ 기본 단위는 비트와 바이트
- 정수와 실수의 표현 방식
 - ✓ 정수의 음수 표현
 - ✓ 실수는 근사치로 표현
- 비트 연산자
 - ✓ 비트 연산을 통해 곱셈, 나눗셈 수행 가능