2주 2강

데이터의 2진수 표현

데이터의 2진수 표현

1. 일반적인 디지털 장치에서는 2진수로 양의 정수, 음의 정수 그리고 소수를 표현

2. 2진수는 0, 1, 부호 및 소수점의 기호를 이용하여 수를 표현

3. 부호가 있고 소수점을 포함하는 동일 값의 10진수와 2진수를 나타낸 예
   $(-13.625)_{10} = (-1101.101)_2$

4. 2진법으로 부호를 갖는 정수와 소수를 표현하려면 추가적으로 부호와 소수점의 기호를 사용하여야 하므로 단순한 진법 변환으로 해결되지 않는다.

2 부호가 존재하는 2진 정수의 표현

- 디지털 장치에서는 부호를 구분할 수 있는 (+)와 (-)같은 별도의 기호는 존재하지 않고 최상위 비트 자리를 부호 비트로 할당하고 0이면 양수의 의미를, 1이면 음수의 의미를 갖는다.
- 나머지 비트들은 적절한 형태로 크기 값을 표현

부호화 - 크기 표현(signed-magnitude representation)

1의 보수 표현(1's complement representation)

2의 보수 표현(2's complement representation)

2. 보수를 이용한 부호를 갖는 2진수의 표현 (1)

1. 1의 보수(1's complement) 표현: 모든 비트들을 반전 (0 → 1, 1 → 0)

2. 2의 보수(2's complement) 표현: 모든 비트들을 반전하고, 결과값에 1을 더한다.

- 보수를 이용한 2진수의 부호 변경

$(+9)_{10} = (0\ 0001001)_2$

$(-9)_{10} = (1\ 1110110)_2$ (1의 보수)

$(-9)_{10} = (1\ 1110111)_2$ (2의 보수)

$(+35)_{10} = (0\ 0100011)_2$

$(-35)_{10} = (1\ 1011100)_2$ (1의 보수)

$(-35)_{10} = (1\ 1011101)_2$ (2의 보수)

2. 보수를 이용한 부호를 갖는 2진수의 표현 (2)

3. 보수를 이용하면 부호비트가 자연스럽게 변경되고 그 크기도 적절한 형태로 변경된다.

4. 2의 보수는 0에 대한 표현이 하나만 존재한다. 그리고 앞 절의 부호 없는 뺄셈 연산에서 보았던 것처럼 산술 연산이 용이하다.

5. 2의 보수는 가장 효율적이기 때문에 컴퓨터를 비롯한 디지털 장치에 부호를 갖는 2진수를 표현하는데 사용이 된다.

3. 10진수의 2의 보수로 표현된 2진수 변환 과정 (1)

1 10진수 $(-25)_{10}$를 2의 보수로 표현된 2진수를 변환하는 과정

1단계. 10진수를 부호가 없는 2진수로 변환한다.

$$(25)_{10} = (11001)_2$$

2단계. 부호 비트를 삽입한다.

$$(25)_{10} = (011001)_2$$

3단계. 1의 보수를 구한다.

$$(011001)_2 \Rightarrow (100110)_2$$

3. 10진수의 2의 보수로 표현된 2진수 변환 과정 (2)

1 10진수 $(-25)_{10}$를 2의 보수로 표현된 2진수를 변환하는 과정

4단계. 2의 보수를 구한다.

$$(100110)_2 \Rightarrow (100111)_2$$

따라서 다음 결과를 얻을 수 있다.

$$(-25)_{10} \Rightarrow (100111)_2$$

4. 2의 보수로 표현된 2진수를 10진수로 변환 (1)

1. 2의 보수로 표현된 양의 정수(최상위 비트: $a_{n-1} = 0$)는 부호 비트를 제외한 크기의 비트들은 실제의 크기를 나타낸다.

- 부호 없는 2진수를 10진수로 변환하는 방법과 동일

$$A = a_{n-2}\times2^{n-2} + a_{n-3}\times2^{n-3} + \ \ldots\ a_1\times2^1 + a_0\times2^0$$

2 2의 보수로 표현된 음의 정수(최상위 비트: $a_{n-1} = 1$)

- 부호 비트의 해당하는 최상위 비트의 자릿수를 2의 승수로 표현하고 (-)를 붙여서 음수가 되도록 한다. 나머지 비트는 양의 정수와 동일

$$A = -2^{n-1} + (a_{n-2}\times2^{n-2} + a_{n-3}\times2^{n-3} + \ldots\ a_1\times2^1 + a_0\times2^0)$$

예

$(10101110)_2$

$= -128 + (1 \times 2^5 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1)$

$= (-82)_{10}$

4. 2의 보수로 표현된 2진수를 10진수로 변환 (3)

2 2의 보수로 표현된 음의 정수(최상위 비트: $a_{n-1} = 1$)

- 2진수 음의 정수를 보수를 이용하여 양의 정수로 만들고 이것을 10진수로 변환. 그리고 최종 단계에서 (-) 부호를 붙이는 방식이다.

예

1단계 : 2의 보수를 이용하여 음수를 양수로 변환

(10101110 → 01010010)

2단계 : $(01010010)_2$

$= -(1 \times 2^6 + 1 \times 2^4 + 1 \times 2^1)$

$= -(64 + 16 + 2) = (-82)_{10}$

5. 2진수의 표현 범위 (1)

1 2의 보수를 사용한 3비트 이진수 표현의 예

$$+3 = (011)_2$$

$$+2 = (010)_2$$

$$+1 = (001)_2$$

$$+0 = (000)_2$$

$$-1 = (111)_2$$

$$-2 = (110)_2$$

$$-3 = (101)_2$$

$$-4 = (100)_2$$

- 표현할 수 있는 수의 범위는 -4 ~ 3이 된다.
  이것은 $-2^{3-1} \sim 2^{3-1}-1$로 표현된다.

5. 2진수의 표현 범위 (2)

2 n비트 데이터의 경우로 일반화해서 수의 범위를 나타내면 다음과 같다.

$$-2^{n-1} \leq N \leq 2^{n-1}-1$$

6. 부호가 있는 8비트 이진수의 표현 (1)

- **부호화-크기 표현 :** $-(2^7 - 1) \sim +(2^7 - 1)$
- **1의 보수 :** $-(2^7 - 1) \sim +(2^7 - 1)$
- **2의 보수 :** $-2^7 \sim +(2^7 - 1)$

6. 부호가 있는 8비트 이진수의 표현 (2)

십진수	부호화-크기 표현	1의 보수	2의 보수
127	01111111	01111111	01111111
126	01111110	01111110	01111110
: :	: :	: :	: :
1	00000001	00000001	00000001
+0	00000000	00000000	00000000
-0	10000000	11111111	x
-1	10000001	11111110	11111111
-2	10000010	11111101	11111110
: :	: :	: :	: :
-126	11111110	10000001	10000010
-127	11111111	10000000	10000000
-128	x	x	10000000

다음 시간

2주 3강. 문자데이터 표현과 2진 연산