Binary Representation of #s.

Many Formats for representation of binary #s.

- one's complement
- sign & mag.
- two's complement → most commonly used.

Read # in two's complement with $\infty$ precision.

$$ x = X_m \left( -b_0 + \sum_{i=1}^{\infty} b_i \ 2^{-i} \right) $$

$X_m =$ arbitrary scale factor  
$|x| < X_m$  
$b_i =$ either zero or 1
\( b_0 = \text{sign bit} \rightarrow \begin{cases} b_0 = 0 & -x_m \leq x \leq x_m \\ b_0 = 1 & -x_m \leq x < 0 \end{cases} \)

With finite \# of bits \((B+1)\) we get representation:

\( x_B = Q_B \lceil x \rceil = x_m \left( -b_0 + \sum_{i=1}^{B} b_i 2^{-i} \right) \)

\( x_B \) is quantized version of \( x \).

Smallest difference between any \( z \) in quantized domain \( \Delta = x_m 2^{-B} \)

5.87923 \( \approx \) only use 2 bits \( \approx 5.87924 \)
- quantized #s are in the range

\[-X_m \leq \hat{x} \leq X_m\]

\[X_B = b_0 \circ b_1 b_2 b_3 \ldots b_8\]

binary point

5.3924 → 5.4 Rounding

5.3 Truncation
Start a real number $X$ to get $X_B$, one can either round off or truncate:

Show Fig. 6.37(a), 6.37(b) in OXS.

Quantization error

$$e = Q_B[x] - X$$

$$= X_B - X$$

2′s complement:

Rounding error $-\frac{1}{2} < e < +\frac{1}{2}$

Truncation error $0 \leq e < \frac{1}{2}$
Interesting property of two complex numbers:

1. Natural overflow:

   Add few $\#s$, if the final sum doesn't overflow, then result is correct even though the intermediate results overflow.

2. Trade-off between overflow & rounding error:

   $X \uparrow \rightarrow$ overflow is less likely but $\Delta \uparrow$, $e \uparrow$
\[ x_n \downarrow \rightarrow \text{overflow is more likely.} \]
\[ \text{but } A \uparrow \text{ e} \downarrow \]

- But B can come too near.

- Keep \( x_n \) large to minimize chance of overflow.

- But keep B large to keep D, e small.

Multiplication also introduces overflow.