Show Fig 6.39, 6.40, 6.41 OKS.

Binary Representation of #s.

Sept. 23, 2008

Many Formats for representation of binary #s.

- one's complement
- sign & mag.
- Two's complement. ← most commonly used.

- Real # in two's complement with ∞ 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 = sign bit $\longrightarrow \begin{cases} b_0 = 0 & 0 \le x \le X_m \\ b_0 = 1 & -X_m \le x < 0 \end{cases}$

---

with finite # of bits $(B+1)$ we get representation:

$$\hat{X}_B = Q_B[x] = X_m \left(-b_0 + \sum_{i=1}^{B} b_i 2^{-i} \right)$$

$\hat{x}$ = is quantized version of x.

smallest difference between any 2 #s in quantized domain $\Delta = X_m 2^{-B}$

5.87923, only use 2 bits 5.87924

- quantized #s are in the range.

$$-X_m \leq \hat{x} < X_m$$

$$\hat{X}_B = b_0 \; \square \; b_1 \; b_2 \; b_3 \ldots\ldots \; b_B$$

binary point.

$5.3924 \longrightarrow \begin{cases} 5.4 & \text{rounding} \\ 5.3 & \text{Truncation} \end{cases}$

Start a real number x
to get $\hat{x}_B$, one can either
rounding or Truncation:

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

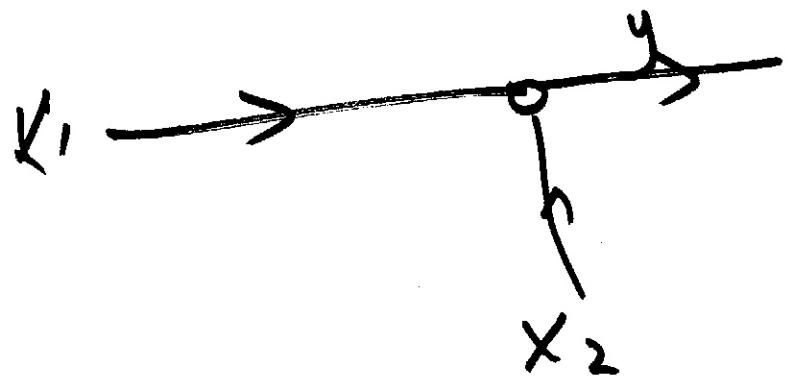
Quantization error

$$e = Q_B[x] - x$$
$$= \hat{x}_B - x$$

2's complement:

rounding error $-\Delta/2 < e < +\Delta/2$

Truncation error $0 \leq e \leq \Delta$



$|x_1 + x_2| < 10{,}000$

$y = x_1 + x_2$

$|x_1| < 5000$

$|x_2| < 5000$

overflow.

Saturate

natural overflow

6.38

O&S

- Interesting property of Twos Complement.
  + natural overflow:

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

- Tradeoff between overflow & rounding error.

$X_m \uparrow \longrightarrow$ overflow is less likely but $\Delta \uparrow$, $e \uparrow$

$X_m \downarrow \longrightarrow$ overflow is more likely.

but $D \downarrow$ $e \downarrow$

But B can come to rescue.

- Keep X_m large to minimize chances of overflow
- But keep B large to keep D, e small.

Multiplication also introduces
- overflow
- underflow