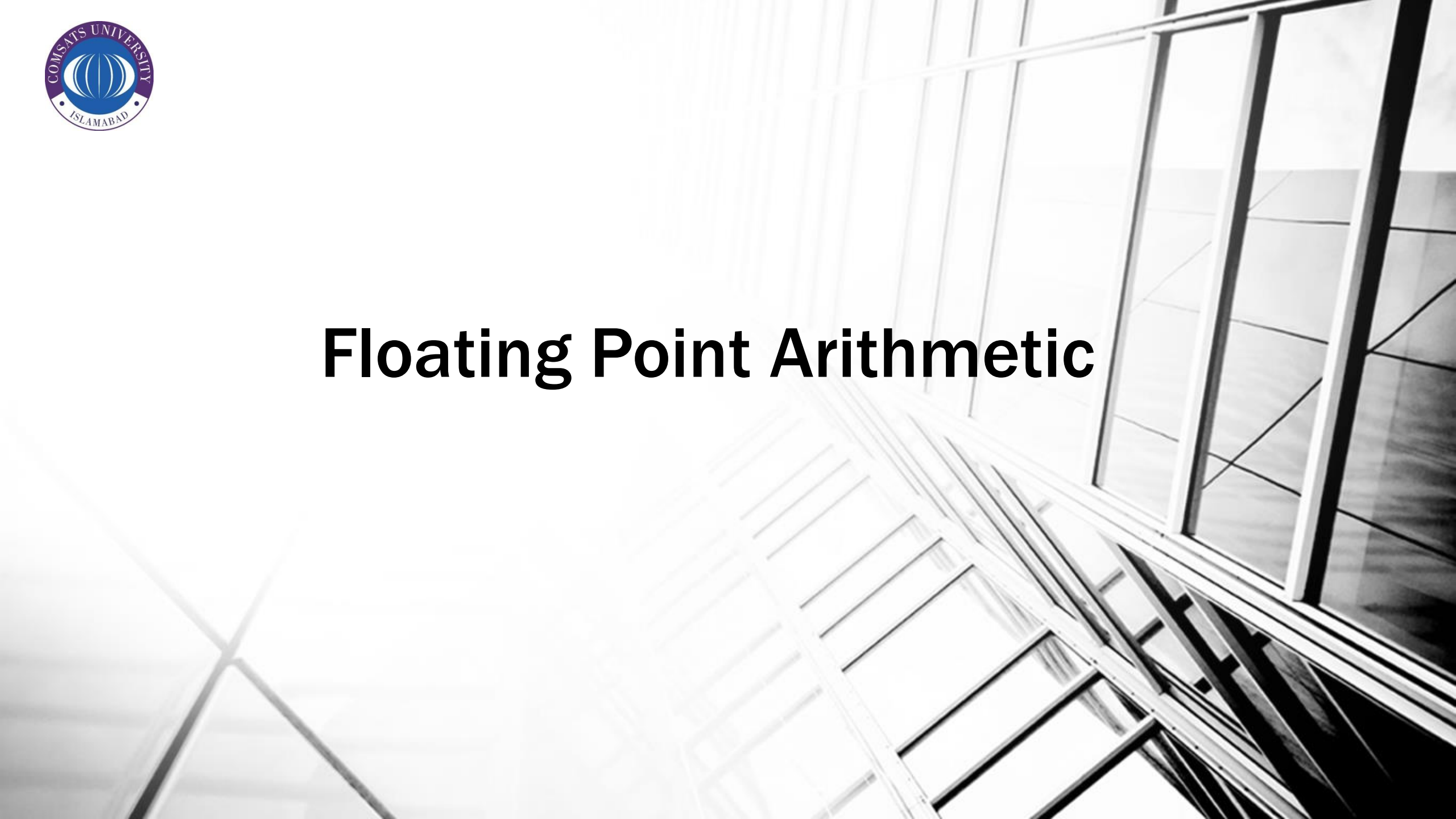
Computer Organization & Architecture

Moazzam Ali Sahi



Floating Point Arithmetic

Example 1 (5.75 + 14)

5.75 + 14 = 19.75

IEEE representation of ‘5.75’:

0 10000001 01110000000000000000000

[0—129—0.437500]

IEEE representation of ‘14’:

0 10000010 11000000000000000000000

[0—130—0.750000]

Preliminary exponent: MAX(129, 130) = 130

Example 1 Continue...

Mantissa:

- Shift 1.01110. . . by 130−129=1 bit
- the signs are identical → ADD
- add 0.10111 and 1.11000:

$$0.\ 1\ 0\ 1\ 1\ 1$$

$$+\ 1.\ 1\ 1\ 0\ 0\ 0$$

$$1\ 0.\ 0\ 1\ 1\ 1\ 1$$

- Normalize exponent and mantissa:

$$1.001111 \times 2^{(131-127)}$$

Example 1 Continue...

IEEE representation of result:

0 10000011 00111100000000000000000

[0—131—0.234375]

Example 2 (5.75 - 14)

5.75 - 14 = -8.25

IEEE representation of ‘5.75’:

0 10000001 01110000000000000000000

[0—129—0.437500]

IEEE representation of ‘14’:

0 10000010 11000000000000000000000

[0—130—0.750000]

Preliminary exponent: MAX(129, 130) = 130

Example 2 Continue...

Mantissa:

- Shift 1.01110. . . by 130−129=1 bit
- the signs are different → SUB

Example 2 Continue...

From 0.10111 subtract 1.11000:

0 0. 1 0 1 1 1

2's-complement→ +1 0. 0 1 0 0 0

1 0. 1 1 1 1 1

sign of difference is negative

2's-complement of result: 0 1. 0 0 0 0 1

- positive S_a, thus result negative
- no normalization required

Example 2 (5.75 - 14)

IEEE representation of result:

1 10000010 00001000000000000000000

[1—130—0.031250]