Computer Arithmetic: Part IV

Floating Point Representation (IEEE-754)

- ✓ A number F is represented as a triplet <s, E, M>
- ✓ $F = (-1)^S M * 2^E$

s_n	$e_k e_{k-1} \dots e_0$	$m_{n-1} m_{n-2} \dots m_0 . m_{-1} \dots m_{-m}$
Sign	Designates the positions of the radix point	Signed fixed point number, either an integer or a fractional number

- ✓ Sign bit indicating negative =1 or positive =0
- ✓ M is called the Mantissa, and is normally a fraction in the range of [1.0-2.0]
- ✓ E is called the exponent, which weights the number by power of 2.

Encoding:

- Single-precision numbers: total 32 bits, E 8 bits, M 23 bits
- Double-precision numbers: total 64 bits, E 11 bits, M 52 bits

s	E	M

- Range of E: $1 \le E \le 254$ (all 0s and all 1s are reserved for special number)

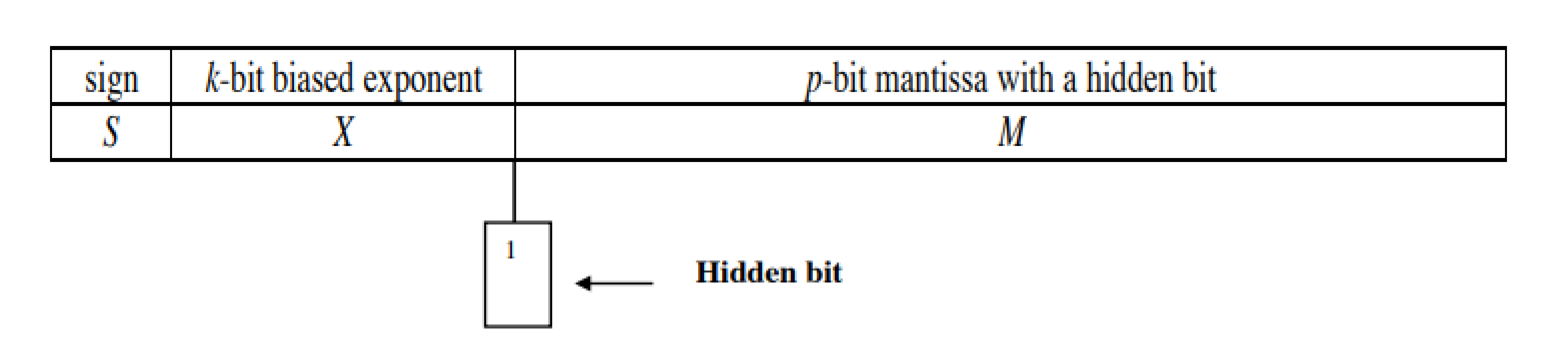
- Encoding Exponent with bias value: E = Exponent + Bias
  - (Bias : Single Precision = 127, Double Precision = 1023)

- Encoding Mantissa M
  - The mantissa is coded with an implied leading 1 (i.e. in 24 bits).

    $M = 1\,.\,xxxx...x$
  - Here, *xxxx…x* denotes the bits that are stored for the mantissa. We get the extra leading bit for free.

Bias

- The value stored is offset from the actual value by the exponent bias, also called a biased exponent
- Biasing is done so that exponents can be +ve or –ve, in two's complement



- The true exponent, x, is found by subtracting a fixed number from the biased exponent, X. This fixed number is called the bias. For a k-bit exponent, the bias is $2^{k-1}-1$, and the true exponent, x and X are related by

$$x = X - (2^{k-1}-1)$$

Example: In single precision, if exponent bias X = 134, then x = 134 -127= 7

Floating Point Addition/Subtraction

- Two numbers: $M1 \times 2^{E1}$ and $M2 \times 2^{E2}$, where $E1 > E2$ (say).
- Basic steps:
  - Select the number with smaller exponent (in this case E2) and shift its mantissa right by (E1-E2) positions
  - Set the exponent of the result equal to the larger exponent (i.e. E1)
  - Carry out M1 ± M2, and determine the sign of the result.
  - Normalize the resulting value, if necessary.

Example: Addition

- N1 = 135.75 , N2 = 2.375
- N1 = $(135.75)_{10} = (10000111.11)_2 = 1.000011111 * 2^7$
- N2 = $(2.375)_{10} = (10.011)_2 = 1.0011 * 2^1$
- Adjust Mantissa- By shifting N2 right by 7-1 = 6 positions and add:
  - N1 in 24 bits = 1000 0111 1100 0000 0000 0000
  - N2 in 24 bits after right shifting 6 = 0000 0010 0110 0000 0000 0000

1	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
						1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

  - Sign bit = 0; Exponent = 7 , Biased Exponent = 127+7 = 134 = 1000 0110
  - Mantissa = 000 1010 0010 0000 0000 0000
  - Putting it all together = $(0100\ 0011\ 0000\ 1010\ 0010\ 0000\ 0000\ 0000\ )_2$ = 0x430A2000

Subtraction

Suppose we want to subtract F2 = 224 from F1 = 270.75

$F1 = (270.75)_{10} = (100001110.11)_2 = 1.0000111011 \times 2^8$

$F2 = (224)_{10} = (11100000)_2 = 1.11 \times 2^7$

Shift the mantissa of F2 right by 8 – 7 = 1 position, and subtract:

1000 0111 0110 0000 0000 0000

111 0000 0000 0000 0000 0000

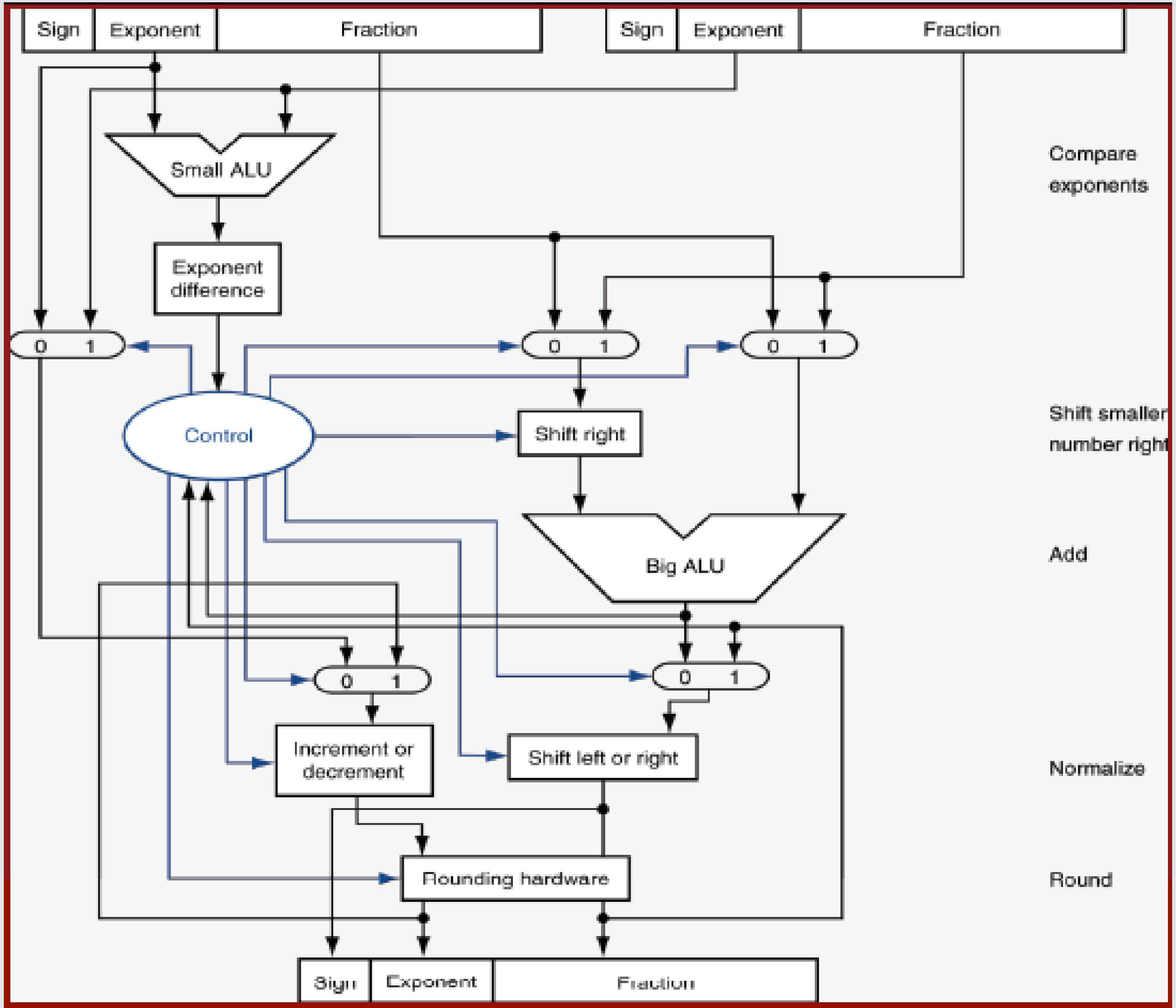
0001 0111 0110 0000 0000 0000 000

For normalization, shift mantissa left 3 positions, and decrement E by 3.

Result: 1.01110110×2^5

Sign = 0
Exponent = 5, Exponent with Biased = 127 + 5 = 132 = 1000 0100
Mantissa = 0111 0110 0000 0000 0000 000
Putting it altogether = 0100 0010 0011 1011 0000 0000 0000 0000
= 0x423B0000

Addition and Subtraction Hardware



Floating Point Multiplication

- Two numbers: $M1 \times 2^{E1}$ and $M2 \times 2^{E2}$
- Basic steps:
  - Add the exponents *E1* and *E2* and subtract the *BIAS*. Here E1 and E2 are the biased exponents.
  - Multiply *M1* and *M2* and determine the sign of the result.
  - Normalize the resulting value, if necessary.

Multiplication Example

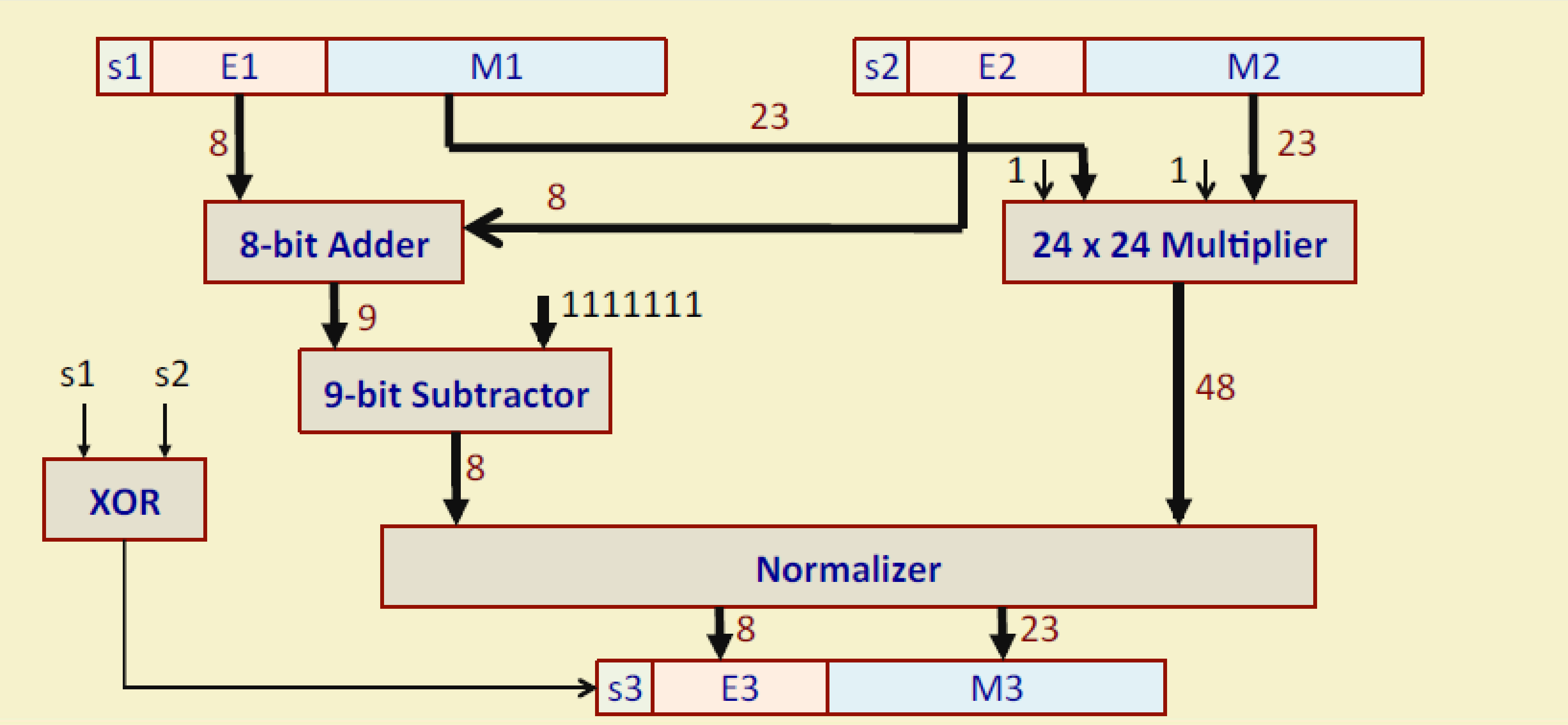
- Suppose we want to multiply F1 = 270.75 and F2 = -2.375

F1 = $(270.75)_{10} = (100001110.11)_2 = 1.0000111011 \times 2^8$

F2 = $(-2.375)_{10} = (-10.011)_2 = -1.0011 \times 2^1$

- Add the exponents: 8 + 1 = 9
- Multiply the mantissas: 1.01000001100001
- Result: $1.01000001100001 \times 2^9$
- Sign bit = 1
- Exponent = 9, Biased Exponent = 127 + 9 = 136 = 1000 1000
- Mantissa = 0100 0001 1000 0100 0000 000
- Putting it altogether = 1100 0100 0010 0000 1100 0010 0000 0000 = 0xC420C200

Hardware of Multiplication



Floating Point Division

- Two numbers: $M1 \times 2^{E1}$ and $M2 \times 2^{E2}$
- Basic steps:
- Subtract the exponents *E1* and *E2* and add the *BIAS*. Here E1 and E2 are the biased exponents
- Divide *M1* by *M2* and determine the sign of the result.
- Normalize the resulting value, if necessary.

Division Example

- Suppose we want to divide F1 = 270.75 by F2 = -2.375

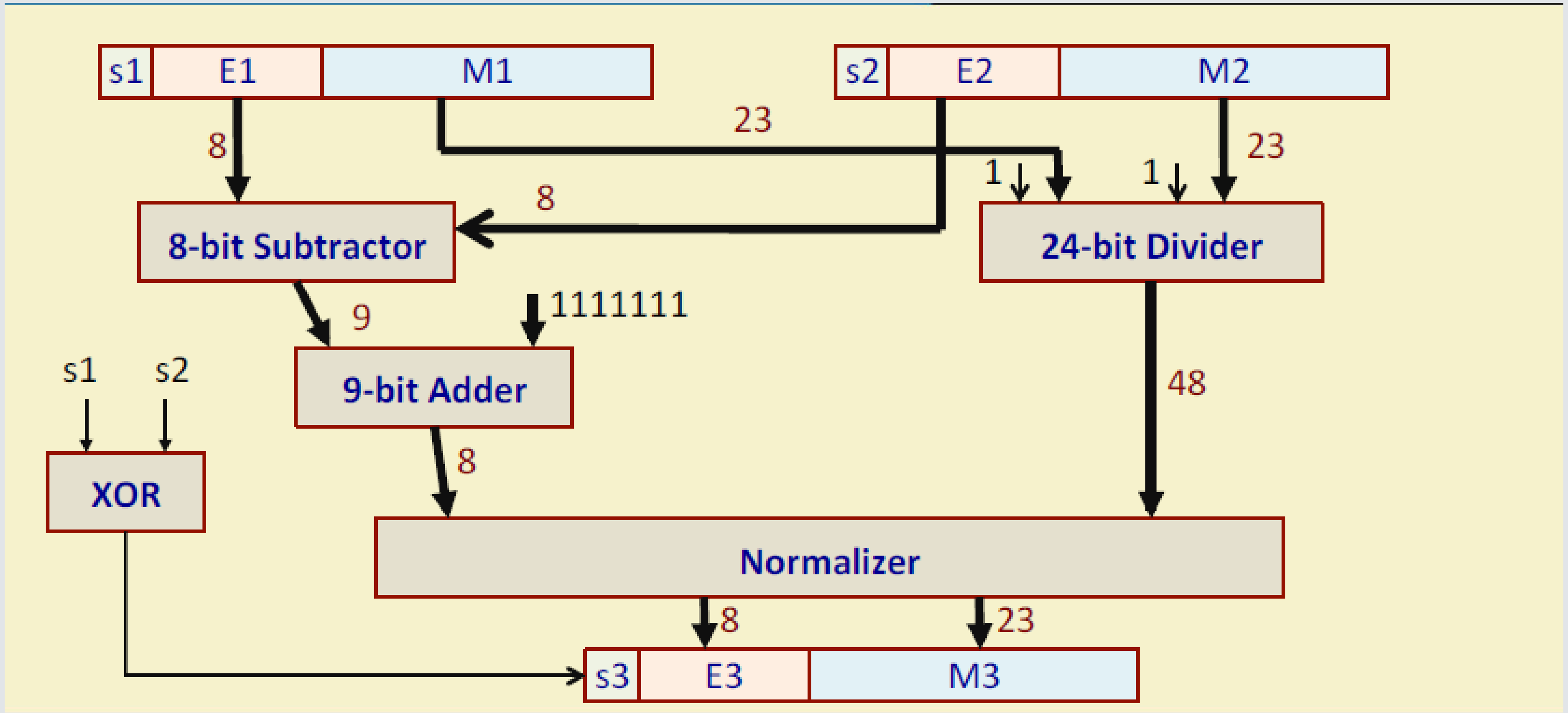
$$F1 = (270.75)_{10} = (100001110.11)_2 = 1.0000111011 \times 2^8$$

$$F2 = (-2.375)_{10} = (-10.011)_2 = -1.0011 \times 2^1$$

- Subtract the exponents: 8 – 1 = 7
- Divide the mantissas: 0.1110010
- Result: 0.1110010×2^7
- After normalization: 1.110010×2^6
- Sign bit = 1
- Exponent = 6, Biased Exponent = 6 + 127 = 133 = 1000 0101
- Mantissa = 1100 1000 0000 0000 0000 000
- Putting it altogether = 1100 0010 1110 0100 0000 0000 0000 0000

  = 0xC2E40000

Division Hardware



Thank You