COMPUTER ORGANIZATION AND ARCHITECTURE
UNIT –III
TOPIC- FLOATING POINT MULTIPLICATION PART-1

Floating Point Multiplication

Introduction

Floating point multiplication is comparatively easy than the floating point addition algorithm but off course consumes more hardware than fixed point multiplier circuit.

The multiplication of two floating point numbers requires that we multiply the mantissas and add the exponents. No comparison of exponents or alignment of mantissa is necessary.

Flowchart

The multiplication algorithm can be subdivided into four parts :-

1. Check for zeros.

2. Add the exponents.

3. Multiply the mantissa.

4. Normalize the product.

1. Check for zeros.

$BR=0.5812 \times 10^5$

$QR=0.234 \times 10^7$

If BR or QR = 0

Result i.e., $AC \leftarrow 0$

2. **Add the exponents.**

In multiplication there is no need to align mantissa, we need to add the exponents which will become the exponent of the result.

$a^m \times a^n = a^{m+n}$

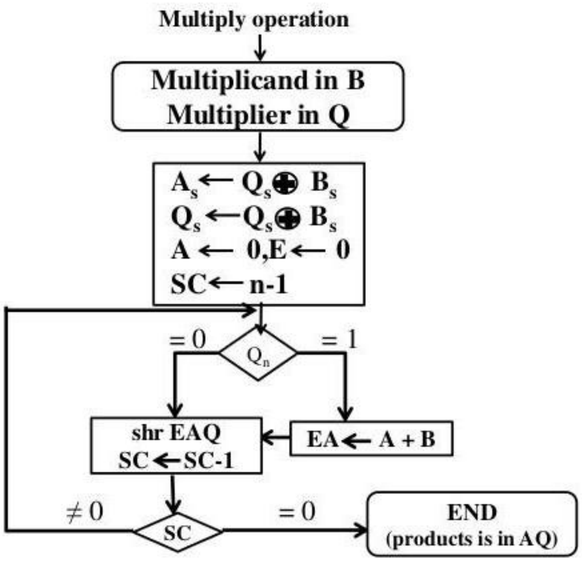
BR=0.5812x10^5

QR=0.234x10^7

$10^5 \times 10^7 =$
10^{5+7}
10^{12}

3. Multiply the mantissa.

Use signed magnitude multiplication to perform multiplication of mantissa.



4. Normalize the product.

let the result is

1.0101101x10^{12}

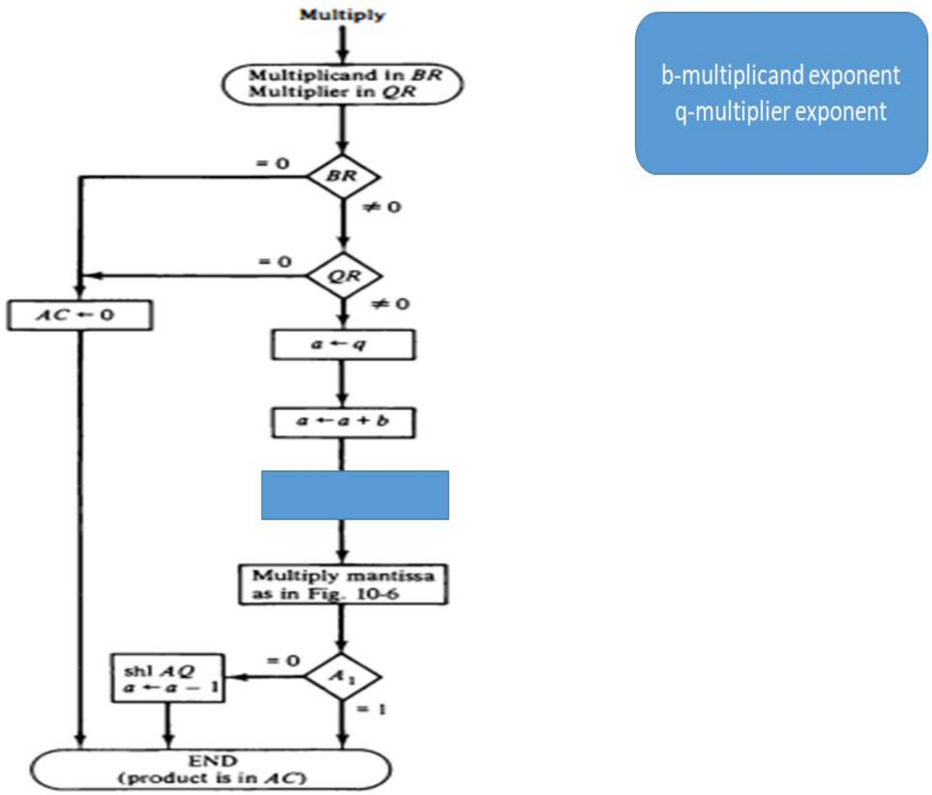
0.101101x10^{11}

Msb of mantissa=0 underflow

NORMALIZED

To overcome underflow, perform a shift left and exponent is decreased.

Figure 10-16 Multiplication of floating-point numbers.



- The two operands are checked to determine if they contain a zero.If either operand is equal to zero, the product in the AC is set to zero and the operation is terminated. If neither of the operands is equal to zero, the process continues with the exponent addition.
- The exponent of the multiplier is in q and the adder is between exponents a and b. It is necessary to transfer the exponent from q to a, add the two exponents, and transfer the sum into a.
- The multiplication of the mantissas is done as in the fixed-point case with the product residing in A and Q. Overflow cannot occur during multiplication, so there is no need to check for it.
- The product may have an underflow, so the most significant bit in A is checked. If it is a 1, the product is already normalized. If it is a 0, the mantissa in AQ is shifted left and the exponent decremented.
- Although the low-order half of the mantissa is in Q, we do not use it for the floating-point product. Only the value in the AC is taken as the product.