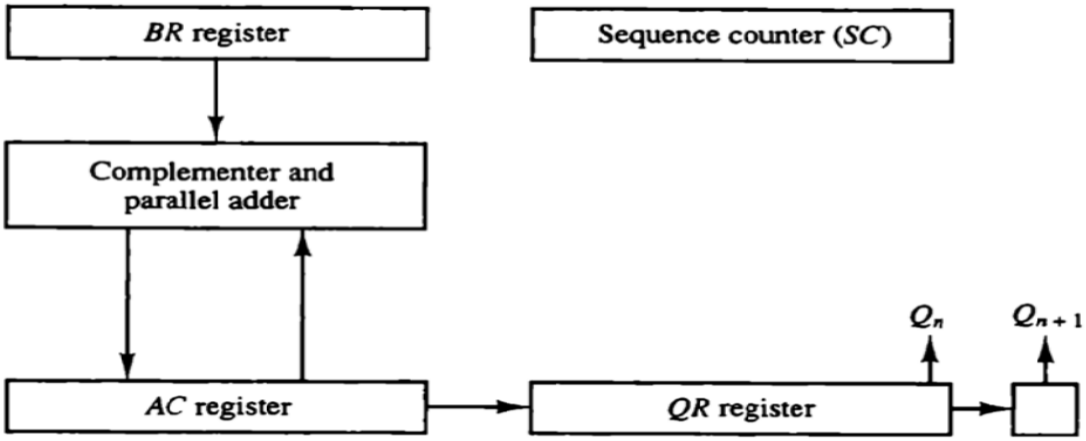
COMPUTER ORGANIZATION AND ARCHITECTURE
UNIT –III
TOPIC- BOOTH'S MULTIPLICATION ALGORITHM PART-1

Introduction

The Booth Algorithm is a widely used technique for multiplying binary integers particularly in signed 2's complement representation.
It was proposed by Andrew D in 1951.
This algorithm is an efficient way to perform multiplication, minimizing the number of additions and subtractions.

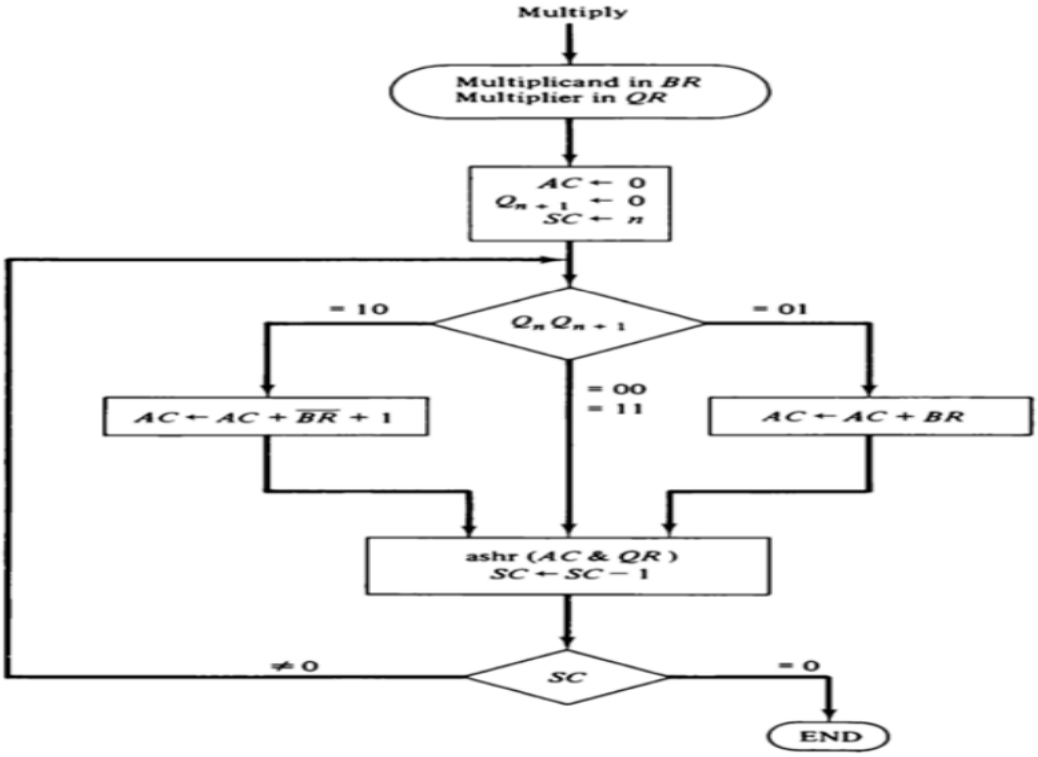
Hardware Implementation

Hardware for Booth algorithm.



• The multiplicand is stored in BR register and the multiplier is stored in QR register.
• AC is a temporary register initialized to 0.
• The complementer has a mode bit which can be 0 or 1.
• If the mode bit is 0, complementer transfers the contents of BR register to the parallel adder.
• If the mode bit is 1, complementer transfers the complement of BR register to the parallel adder.
• Parallel adder performs addition of BR and AC registers and register is transferred to AC register.
• An arithmetic shift right is performed on AC and QR registers by checking the values of Q_n and Q_{n+1} bits.
• Q_n is LSB of the multiplier.
• Q_{n+1} is an extra bit or flip flop initialized to 0 in the algorithm.

Flowchart



Flowchart of Booth's multiplication

1. Initially, we set the AC and Q_{n+1} registers value to 0.
2. SC represents the number of Multiplier bits (Q), and it is a sequence counter that is continuously decremented till equal to the number of bits (n) or reached to 0.
3. A Q_n represents the last bit of the Q, and the Q_{n+1} shows the incremented bit of Q_n by 1.
4. On each cycle of the booth algorithm, Q_n and Q_{n+1} bits will be checked on the following parameters as follows:

    - When two bits Q_n and Q_{n+1} are 00 or 11, we simply perform the arithmetic shift right operation (ashr) of AC and QR by 1 and SC is decremented by 1.
    - If the bits of Q_n and Q_{n+1} is shows to 01, the multiplicand bits (M) will be added to the AC (Accumulator register). Then, we perform the right shift operation to the AC and QR bits by 1 and SC is decremented by 1.
    - If the bits of Q_n and Q_{n+1} is shows to 10, the multiplicand bits (M) will be subtracted from the AC (Accumulator register). Then, we perform the right shift operation to the AC and QR bits by 1 and SC is decremented by 1.

5. The operation continues until SC reaches 0.
6. Results of the multiplication (product) will be stored in the AC and QR registers with 2's complement representation for negative numbers.

Advantages of Booth's Algorithm

- This algorithm is used to multiply signed binary numbers, handling both positive and negative numbers effectively without requiring separate logic for negative numbers.
- The algorithm minimizes the number of additions and subtractions required during multiplication, leading to a faster computation compared to traditional methods.
- Booth's algorithm is well-suited for hardware implementation because it reduces the complexity of arithmetic operations, which is crucial for devices with limited resources like embedded systems.

Disadvantages of Booth's Algorithm

- It can be more difficult to understand and implement compared to traditional multiplication methods. It requires careful handling of sign bits.
- It is primarily designed for signed binary numbers. To use it for unsigned numbers, modifications are required, making it less versatile for all types of binary multiplication.
- The algorithm involves multiple iterations of addition, subtraction, and shifting, which may lead to higher latency compared to simpler multiplication methods.
- Although it reduces the number of operations, the algorithm may still consume more power due to the multiple cycles involved in shifting and adding or subtracting the multiplicand.