Multiplication & Division

Dr. Fatemah K Al Assfor

Multiplication of Unsigned Integers

There are three types of high speed multiplier:

- **Parallel Multiplier (Very fast)**
- **Sequential Multiplier (used in DSP applications)**
- **Array Multiplier**

Sequential Multiplier

- **Used in small systems, where high speed operation is unnecessary.**
- **It generates partial products sequentially and adds each newly generated product to previously accumulated partial product (PR).**

Note

The key of this approach is that the sum of appropriately shifted multiplicand.

Example: Design 4X4 unsigned multiplier (Assume $M=5_{10}$ and $Q=12_{10}$)

Multiplicand (M) size (in bits)

Multiplier (Q) size (in bits)

Product size = 2*4 = 8bit

Note The sequential multiplier has **3- registers**:

* **A:** Accumulator (initially cleared to zero, and when the algorithm is terminated this register holds the high order part of the final product).

* **M:** Multiplicand (always hold the multiplicand number).
* **Q:** Multiplier (It is initialized with the multiplier number, and when the algorithm is terminated it holds the lower order part of the result).

The block diagram of **(nXn)** unsigned sequential multiplier is

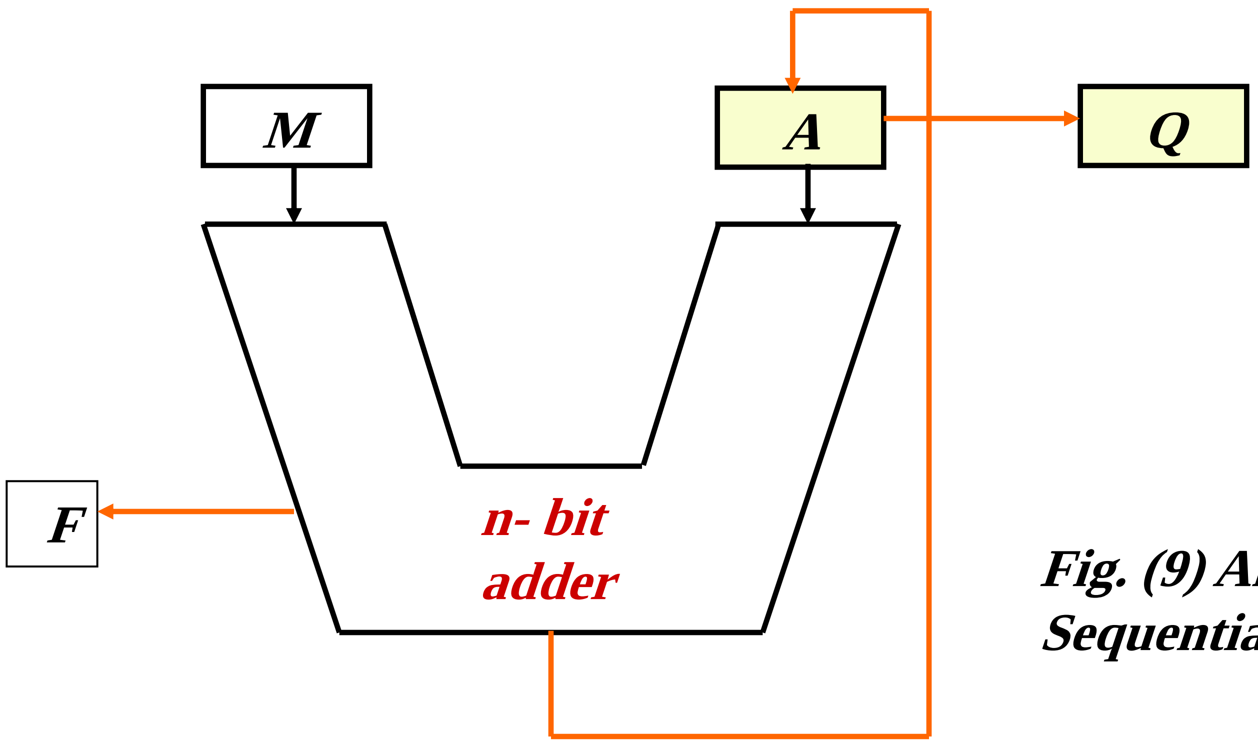

Fig. (9) An nXn-Sequential multiplier

Observation: For (nXn) Sequential Multiplier

- **In each iteration, the operation A= A+M is performed only when Q[0] = 1.**

- **The content of register pair AQ is right shifted in both cases:**

  Q[0] = 0 or 1

The Worst Case of the Multiplier:

The worst case is when all the bits of the multiplier number (Q) are 1s)

⇒ (n) additions + (n) times Right shifts are needed.

In fact, whenever the sequential multiplier contains fewer 1s ⇒ high speed is achieved.

Speeding up Multiplier

To overcome the problem of many additions which is slow down the speed of multiplication:

The multiplier operand (Q) is recoded (or encoded) in such a way such that the string of 1s that may occur in the (multiplier operand) can be converted to a string of 0s surrounded by the digits 1 and $\bar{1}$ (i.e. -1).

***This encoded technique used here is called "Booth Recoding (or encoding)".**

Booth recoding is used with any multiplier type and for unsigned and signed numbers (i.e. the numbers enter to the multiplier circuit are assumed to be 2's complement numbers)

Return

Booth recoding can be implemented by modifying the sequential multiplier hardware. This is done by:

*** Extending the Q- register size from n-bit to(n+1) bits so that the extra position will initially hold the fictitious “0" .**

*** Makes the n-bit parallel adder to perform add / subtract.**

Example: Multiply the following numbers M= -4_{10} and Q= 7_{10}

M= -4_{10} = (1100) in 2's complement representation

and Q= 7_{10} = (0111)

Comments	M	A	Q q_0 q_{-1}	Size
Initialization	1100	0000	0 1 1 1(0)	4

Division Algorithms

Division is the most complex of the four basic arithmetic operations and the hardest one to speedup. Thus, dividers are more expensive and/ or slower than multipliers.

Several classes of algorithms exist for this operation like:

i- Restoring Division Algorithm

ii- Non- Restoring Division Algorithm

iii- SRT Division Algorithm

The implementation of these algorithms can be either sequential, combinational, or both.

Sequential Divider

The implementation of this divider consists of n- iterations of the recurrence. This means that the H/W of the recurrence step is reused for all the iterations and the partial remainder (PR) and the final remainder is updated in a register.

In general, dividing two number unsigned integer numbers using division algorithm will provide a quotient and a remainder.

Consider unsigned integer Numbers:

Perform the division operation (D/X) , such that $D = Q.X + R$

where D: Dividend X: Divisor Q: Quotient R: Remainder

and $0 \leq R \leq D$

Note:

In the digit recurrence algorithms for binary system, a 1-digit (1 bit)/ iteration is generated.

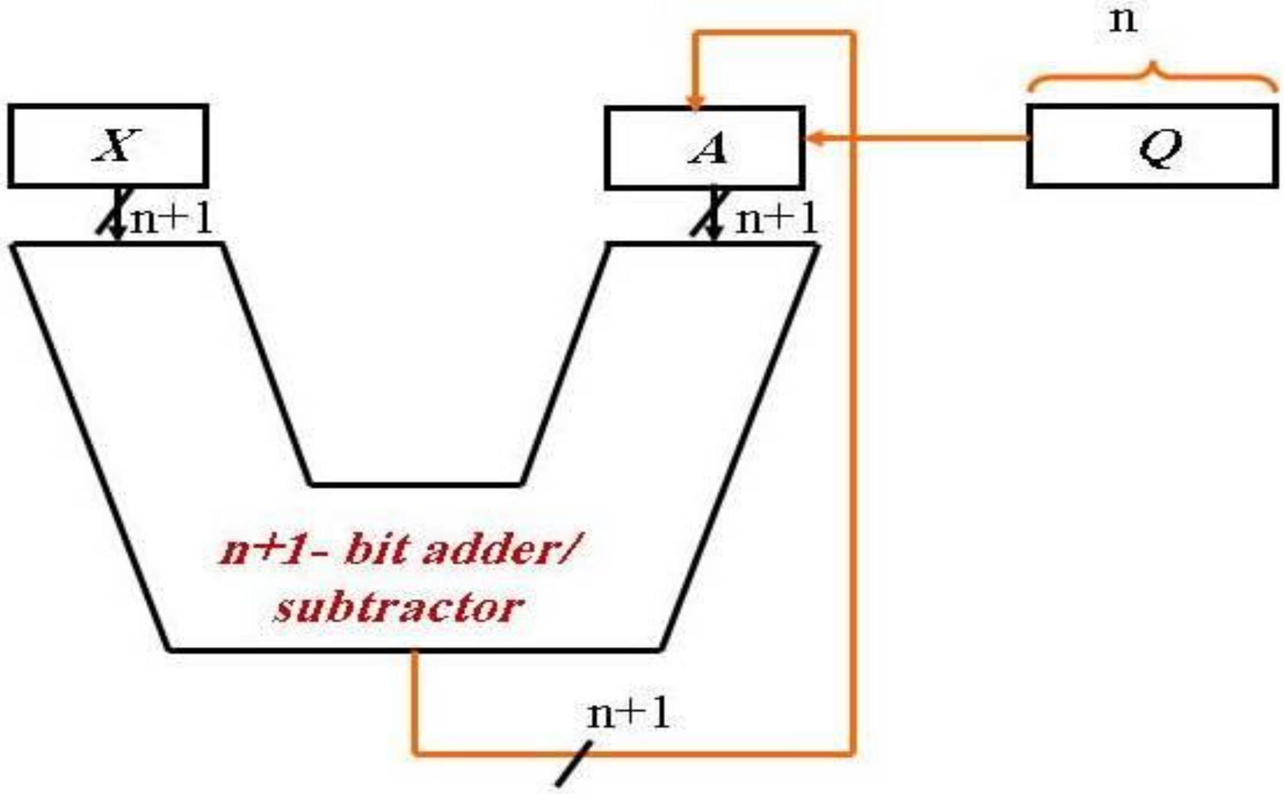

Figure: Block Diagram of Unsigned Restoring Division Algorithm

Example:

Using restoring division algorithm Divide 19/3

Comments	X (n+1)	A (n+1)	Q (n)	Size (n)
Initialization	000011	000000	10011	5

ii- Non- Restoring Division Algorithm

- The restoring division algorithm needs n- subtractions and extra additions when dividing two n-bit numbers.

- The speed of the restoring division algorithm can be improved by eliminating the restoration step.

This elimination can be done if step3 in the restoring division algorithm is performed first, step 1, then step2, and computation starts immediately after the subtraction:

Under this condition, one of the following must be performed:

i- if sign of A is positive, then: a- ShL(AQ)

b -A= A-X

c- Since A[4]=1, set Q[0] =0

ii- if sign of A is negative, then: a– ShL(AQ)

b-A= A+X

c- Since A[4]=0 , set Q[0] =1

Example: Using non-restoring division algorithm, perform the division operation 11/3