

Experiment No. : 10

Aim: To implement a restoring division algorithm.

Theory:

Restoring Division Algorithm is used to divide two unsigned integers. This algorithm is used in Computer Organization and Architecture. This algorithm is called restoring because it restores the value of **Accumulator (A)** after each or some iterations. There is one more type i.e. Non-Restoring Division Algorithm in which value of A is not restored. Let the dividend $Q = 0110$ and the divisor $M = 0100$.

Restoring division:

Restoring division operates on fixed-point fractional numbers and depends on the following assumptions: The following division methods are all based on the form $Q = A / M$ where

Q = Quotient

A = Numerator (dividend)

M = Denominator (divisor).

Algorithm:

Registers used: A, M, Q, n (counter)

Step 1: Load the initial values for the registers.

$A = 0$ (Accumulator), $Q_{res} = 0$, $M = \text{Divisor}$, $Q = \text{Dividend}$ and n is the count value which equals the number of bits of dividend.

Step 2: Shift left {A,Q}.

Step 3: Perform $A = A - M$.

Step 4: Check the sign bit of A. If 0, goto step 5. If 1, goto step 6.

Step 5: Set LSB of Q as 0. Goto step 7.

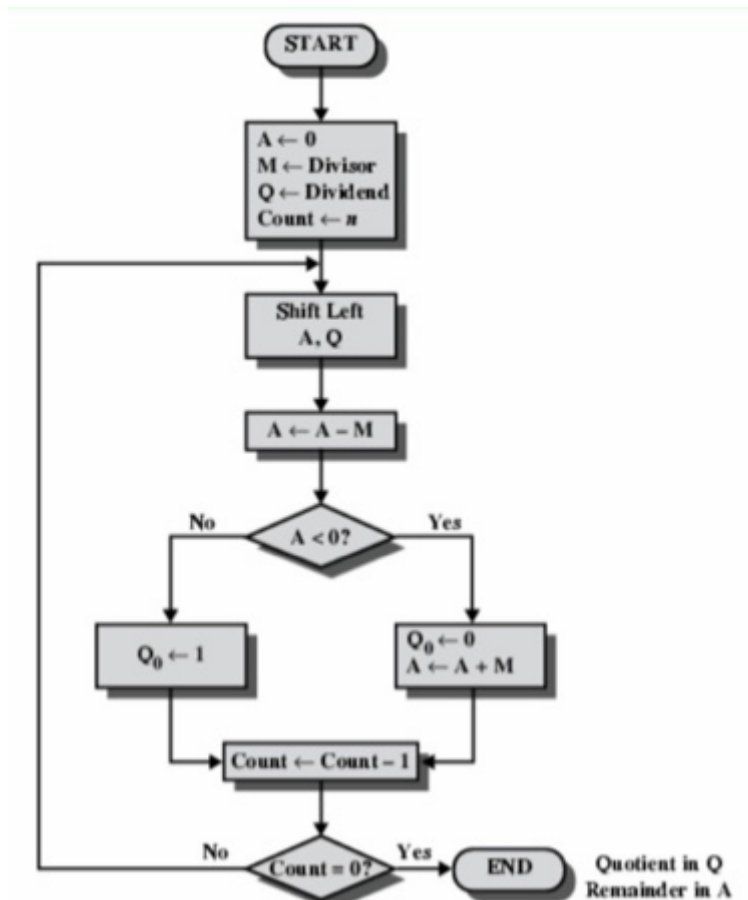
Step 6: Set LSB of Q as 1. Restore the value of A which was present before the subtraction.

Step 7: Decrement count.

Step 8: Check if counter value n is zero. If yes, goto next step. Else, goto step 3.

Step 9: Stop

Flowchart:



Conclusion:

Thus the Program for Restoring Division algorithm has been successfully implemented in C.