Experiment No. 12

Aim:
Simulation of 5 stage or 4 stage or 3 stage pipelining.

Description:
A 4-stage pipeline is a common architectural design used in processors and digital systems to improve preformance by overlapping the execution of multiple instruction or operation. Each stage in the pipeline performs a specific task, and data flows through these stages in a sequential fasion.

Stages of pipeline:

1) Fetch Stage
   → The processor retrives the next instruction or data from the memory.
2) Decode Stage
   → The fetched instruction is decoded to determine the operation to be performed and the operand involved.
3) Execution Stage
   → The decoded instruction is executed, and the operation specified by the instruction is performed on the operands.
4) Writeback Stage
   → The result of the executed operation is written back to the appropriate destination.

A simple diagram illustrating the 4-stage pipeline is shown below:

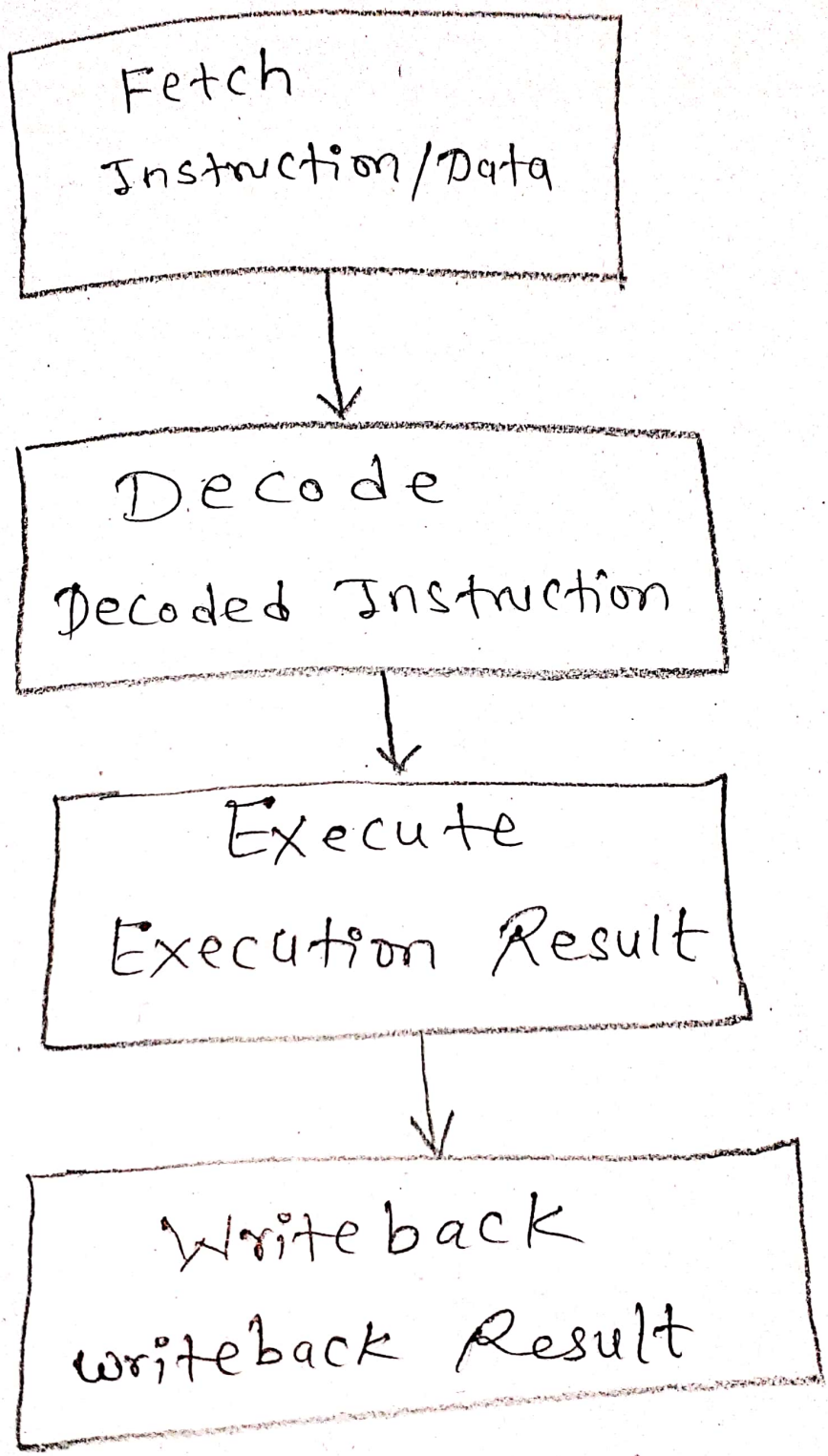

fig: 4-stage pipline

HDL Code

Experiment No. 13

Aim:
To implement Booth addition and subtraction of signed 2's complement data using C.

Description:

The algorithm for adding and subtracting two binary numbers in signed 2's complement representation is shown in flowchart below.

The sum is obtained by adding the contents of AC and BR. The overflow bit 'V' is set to 1 if the XOR of the last two carries is 1, and it is cleared to 0 otherwise.

The subtraction operation is accomplished by adding the contents of AC to the 2's complement of BR. Taking 2's complement of BR has the effect of changing a positive number to negative.

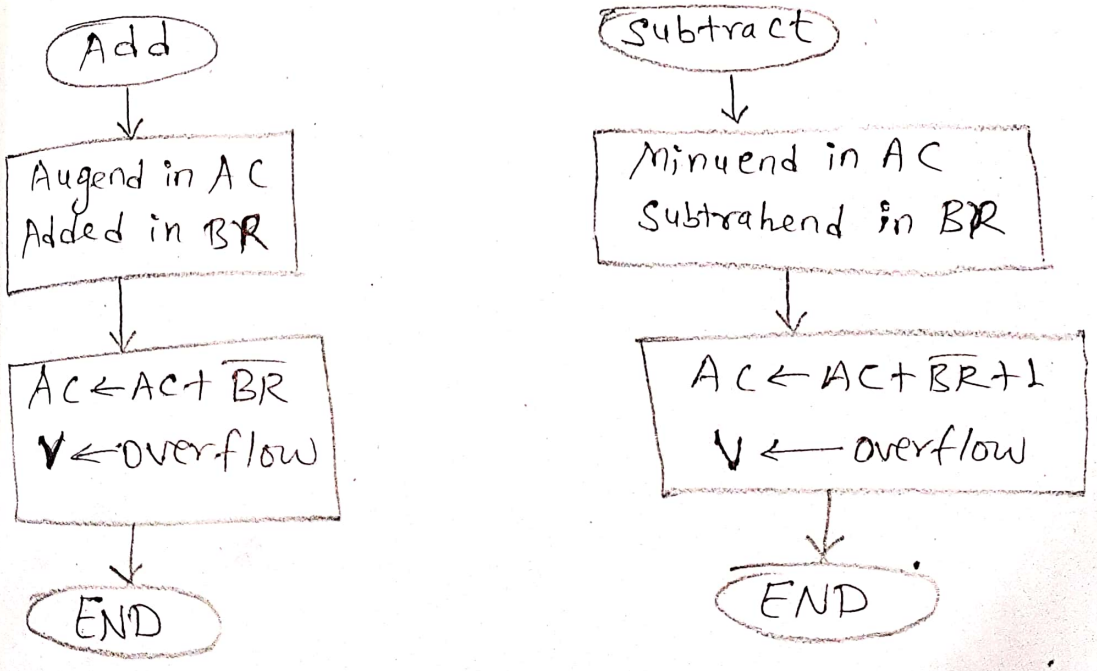

fig: flowchart for adding & subtracting number in 2's complement representation

Experiment No. 14

Aim: Simulation of Boot division algorithm

Description:

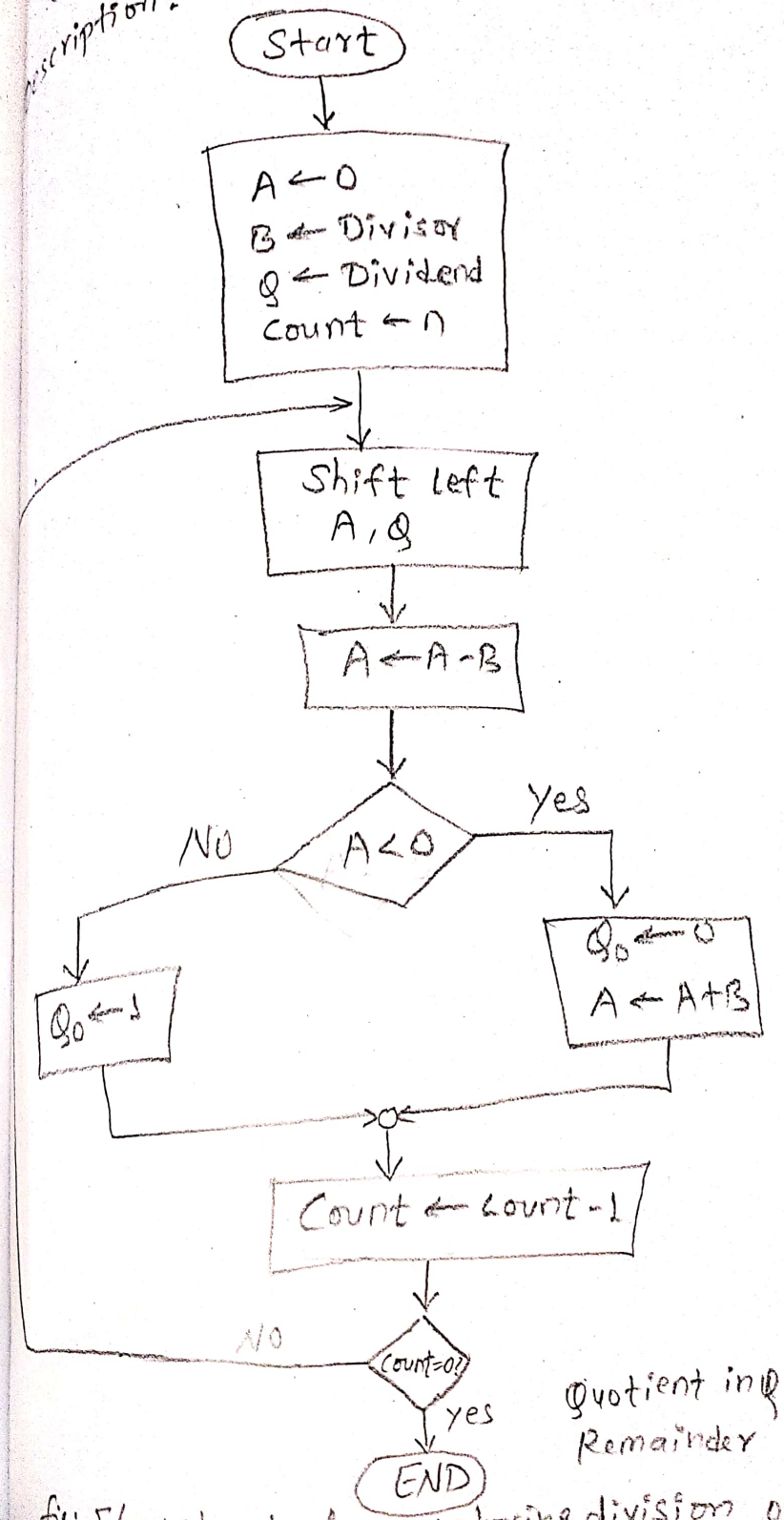

fig: Flowchart for restoring division operation

Aim: To simulate booth Multiplication

Description:

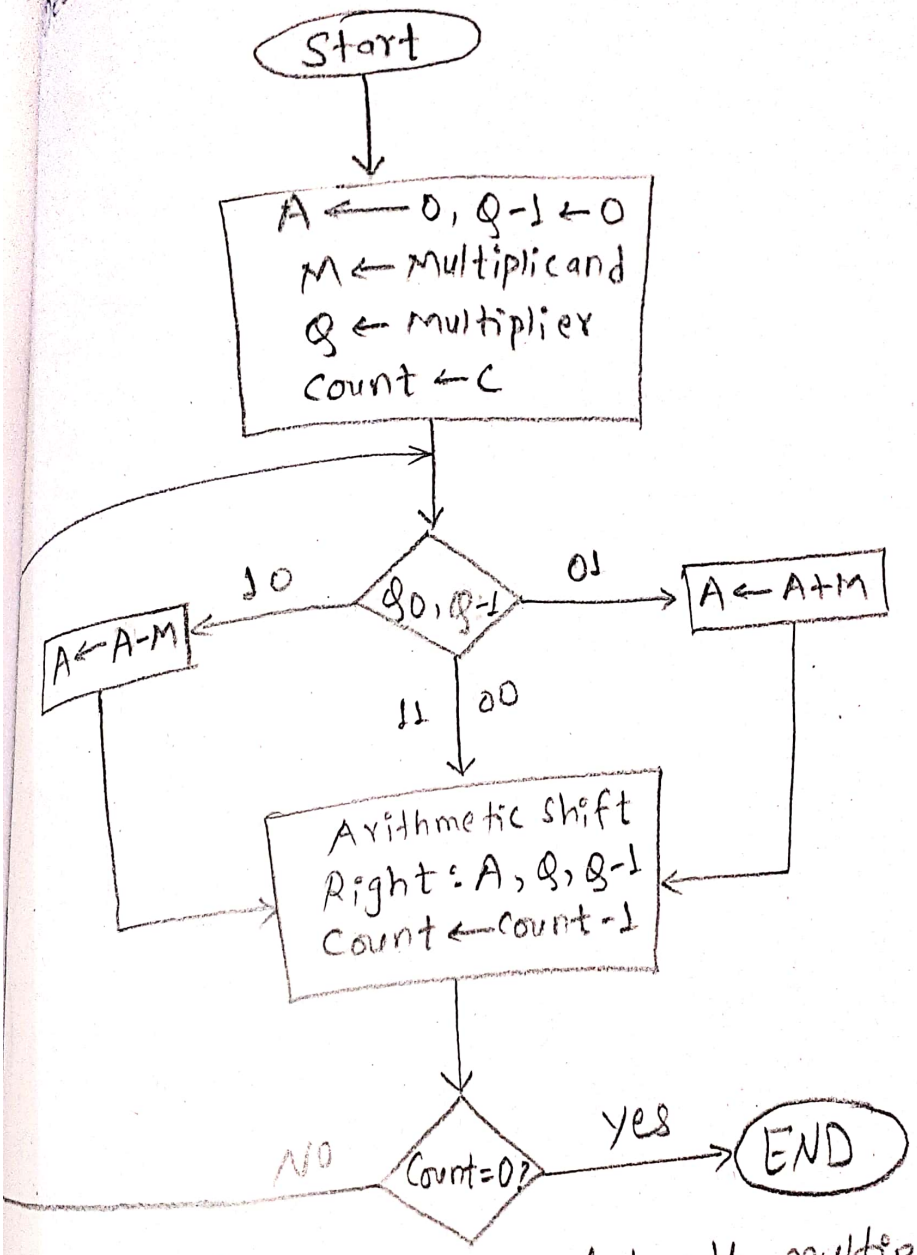

fig: flow chart of booth multiplication.