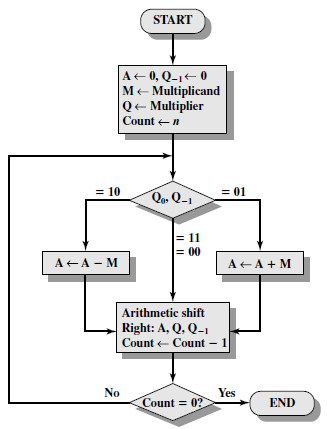
**BOOTH’ S BINARY NUMBER MULTIPLICATION**

**PYTHON IMPLEMENTATION**

**DOCUMENTATION**

**-By Aditya Rana, 2019405**

**Algorithm Description:**



* Take two integer inputs to be multiplied.
* Convert them to binary string format
* Get maximum binary representation length, set number of bits to max\_length+1
* Set the first input as Q
* Set second input as M, calculate -M accordingly. Set M1 as -M and M2 as M
* Enter loop (as in the diagram) checking QLSBqo and implement A-M, A+M, Right Shift according to specified conditions.
* Right Shift :-

Set qo as the LSB of Q. Set Q = ALSB + Q[start to second last ]. Add MSB to A according to previous leading bit and remove LSB.

**Functions’ Description:**

* **main() :**
* input values to be multiplied
* calls booth()
* **booth( int, int) :** 
  + Takes two integer parameters- “a” & ”b”- to be multiplied
  + Calls convert\_To\_binary( ) for each argument ‘a’ and ‘b’
  + Calls balance( ) for every binary number
  + Initiates a while loop ‘n’(number of bits) times, where actual booth implementation occurs
  + Prints Acuumulator, Q, qo values on each iteration
* **convert\_To\_binary( int) :**
* returns binary string to argument
* handles both cases if argument is -ve or +ve, as python’s bin() method returns different binary format for each
* **twos\_comp( int, int) :**
* computes 2’s complement of int value of first argument
* if sign bit is set compute negative value
* return 2’s complement of n-bit
* **add( string, string) :**
* takes two string arguments (binary of two numbers to be added)
* performs binary addition by manipulating the strings
* return n-bit result (binary string)
* **rightShift( string, string, string) :**
* arguments -> binary string of accumulator, binary string of Q, binary string of qo
* reassign them values by splitting as required
* return string of each updated values

**Constraints/Limitations:**

if input 1 is, say, a and input 2 is, say, b, then my implementation of booth algorithm prints faulty values for inputs where a is positive and b is negative.

For example:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Input\_1 | 6 | 2 | 6 | -6 | -6 | 2 | -6 | -2 |
| Input\_2 | 2 | 6 | -2 | 2 | 2 | -6 | -2 | -6 |
| result | ✓ | ✓ | × | ✓ | ✓ | × | ✓ | ✓ |