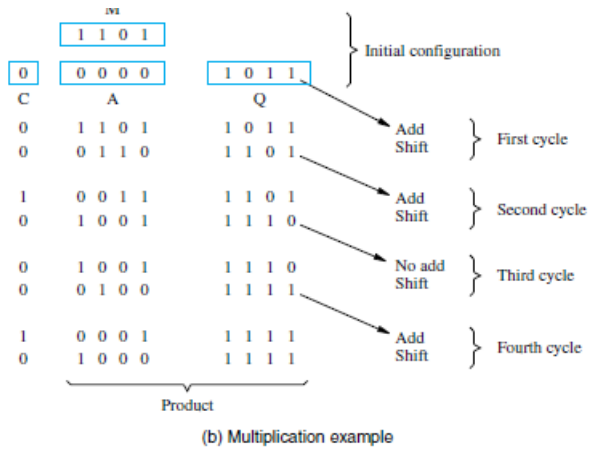
Shift – Add Multiplier

Multiplication is often defined as repeated additions. Thus, to calculate 11 × 23, you would start with 0 and add 11 to it 23 times.



(b) Multiplication example

In this, the 4 bit multiplier is stored in Q register, the 4 bit multiplicand is stored in register B and the register A is initially cleared to zero. The multiplication process starts with checking of the least significant bit of B whether it is 0 or 1.

If the B0 = 1, the number in the multiplicand (B) is added with the least significant bits of the A register and all bits of C, A and Q registers are shifted to the right one bit.

If the bit B0 = 0, the combined C and Q registers are shifted to the right by one bit without performing any addition. This process is repeated for n times for n bit numbers. This method of binary multiplication is called as parallel multiplier.

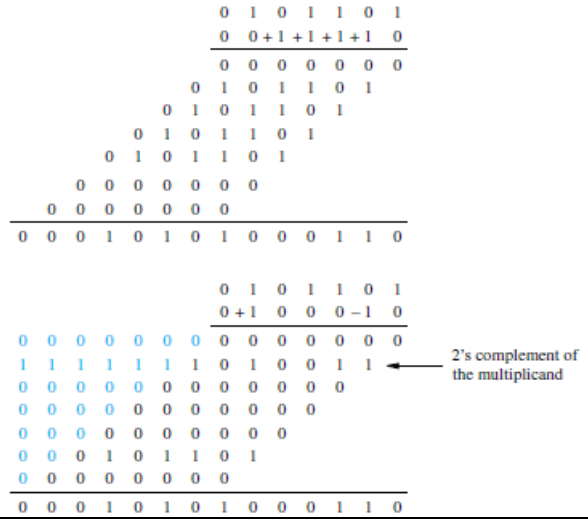
Consider the above figure in which the multiplier and multiplicand values are given as 1011 and 1101 which are loaded into the Q and A registers respectively.

Initially the register C is zero and hence the A register is zero, which stores the carry in addition.

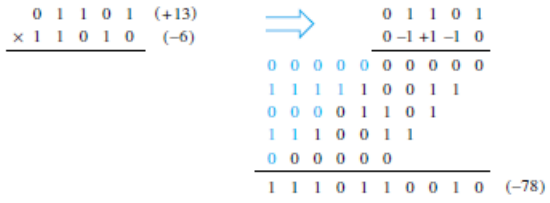
Since the B0 =1, then the number in the B is added to the bits of A and produce the addition result as 1101, and the Q and A register are shifted their values one bit right so the new values during the first cycle are 0110 and 1101 respectively.

This process has to be repeated four times to perform the 4 bit multiplication. The final multiplication result will be available in the A and Q registers as 10001111

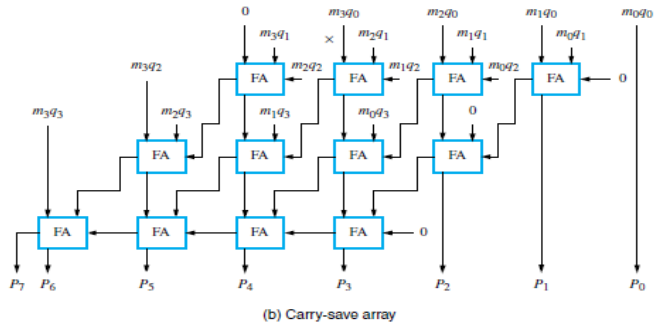
Booth Multiplier



The Booth algorithm generates a 2*n*-bit product and treats both positive and negative 2'scomplement *n*-bit operands uniformly. In general, in the Booth algorithm, −1 times the shifted multiplicand is selected when moving from 0 to 1, and +1 times the shifted multiplicand is selected when moving from 1 to 0, as the multiplier is scanned from right to left.



Carry-Save Addition



(b) Carry-save array

Multiplication requires the addition of several summands. A technique called *carry-save addition* (CSA) can be used to speed up the process.

This structure is in the form of the array in which the first row consists of just the AND gates that produce the four inputs m_3q_0, m_2q_0, m_1q_0, and m_0q_0.

Instead of letting the carries ripple along the rows, they can be "saved" and introduced into the next row, at the correct weighted positions.