PATENT SPECIFICATION (11) 1 547 329

(21) Application No. 40613/77 (22) Filed 29 Sept. 1977
(31) Convention Application No. 728 093 (32) Filed 30 Sept. 1976 in
(33) United States of America (US)
(44) Complete Specification published 13 June 1979
(51) INT. CL.² G06F 7/39
(52) Index at acceptance
G4A 18F 2B2 2BY 8C AU

(72) Inventors MING T. MIU
VIRENDRA S. NEGI

1 547 329

(19)

(54) BINARY MULTIPLIERS

(71) We, HONEYWELL INFORMATION SYSTEMS INC., a Corporation organised and existing under the laws of the State of Delaware, United States of America of 200 Smith Street, Waltham, Massachusetts 02154, United States of America, do hereby declare the invention for which we pray that a patent may be granted to us, and the method by which it is to be performed to be particularly described in and by the following statement:—

The present invention relates to binary multipliers.

Numerous techniques for multiplying two numbers are known, each having certain advantages and disadvantages. For example, some techniques require a rather large amount of logic in order to decrease the time of the multiplication process, and on the other hand some techniques although requiring minimal logic take an extensive period of time to provide the result of the multiplication. Accordingly, it is important in most cases that a balance be achieved between the cost of additional logic circuitry and the time required to perform the multiplication.

An additional consideration in the multiplication process is the sign of the multiplier and the multiplicand. Some multiplication techniques require special treatment for a negative number. For example, in some techniques, the negative number must first be converted to a positive number via for example taking the so-called twos complement of such number. For two negative numbers to be multiplied, then both such numbers must be converted to positive numbers. Obviously, the requirement to initialize the numbers to be multiplied based on their sign requires additional time.

The object of the present invention is to provide a multiplication technique which achieves a good balance between cost and time, particularly with respect to negative numbers.

Accordingly, the present invention provides multiplying apparatus comprising a multiplier register, a multiplicand register, a product register, and means arranged to repeatedly shift the contents of the product register and add into it selectively the contents of the multiplicand register and zero under the control of successive bits of the contents of the multiplier register, the signs of the multiplier and multiplicand being indicated by the topmost bit of each, wherein at each addition following the first 1 bit in the multiplier the sign bit of the multiplicand is inserted into the product register at the bit position corresponding to the position of the topmost bit of the multiplicand on that addition, and the multiplier is converted bit by bit, into complement form as its bits are used to control the additions if its sign is negative.

A multiplication apparatus in accordance with the invention will now be described, by way of example, with reference to the accompanying figures in which:

Figure 1 is a block diagram of the multiplying apparatus;

Figure 2 is a flow diagram of the operation of the apparatus;

Figure 3 shows the format of first, second and third registers in the apparatus;

Figures 4 to 7 are state diagrams of the contents of such three registers during various specific multiplication processes; and

Figure 8 shows gate logic in the apparatus.

By way of an introductory summary, the general principles of binary multiplication as employed in the present system will be briefly discussed. For multiplication of a pair of multibit positive binary numbers, three registers and a parallel binary adder are used; the registers comprise a multiplier register, a multiplicand register, and a (double-length) product register; the lower end of the product register may conveni-

ently be used as the multiplier register, since the bits of the multiplier are used and become "spent" as the multiplication proceeds and the partial product expands from the upper end down into the lower end of its register. Multiplication proceeds by taking each multiplier bit in turn, adding either the multiplicand or 0 into the partial product, and shifting the partial product 1 bit along.

If negative numbers are allowed in the system, matters are somewhat more elaborate. Consider first what happens if "twos complement" notation is used. The top bit of each of the multiplier and multiplicand is now a sign bit. If the multiplier is negative, then the process followed in the present system is to carry out a twos complementing on it as the multiplication proceeds. Thus, starting with the lowest bit, the bits are left unchanged as long as they are 0's, and the first 1 reached is also left unchanged; thereafter, each bit is complemented. The result of the multiplication is the twos complement of the correct product; this may be twos complemented to give the correct product if desired, or may be used in its twos complement form if desired. If the multiplicand is negative, then corrections are required in the additions into the partial product. The rationale behind these corrections can be seen if the multiplicand is imagined to be extended to the left by the inclusion of further bits; extending a positive number in this way is achieved by simply appending 0's, but a negative number has to have 1's appended for such extension. Accordingly, for a negative multiplicand, the whole of the multiplicand (including the top-most, sign, bit) is used for the partial additions; further, a 1 is added into the highest order position at every partial addition following the first 1 bit in the multiplier, regardless of whether the multiplier bit is 0 (partial addition of 000 . . . 0) or 1 (partial addition of the multiplicand). This "topmost position" into which the extra 1 is added is the bit position of the highest order (sign bit of the multiplicand on that addition. Any carry up from this topmost position is ignored. The final result is the correct product of the multiplier (assumed positive) and the (negative) multiplicand, in the same twos complement form as the multiplicand. Multiplication of a negative multiplicand by a negative multiplier is achieved by following both the above procedures, producing the twos complement of the correct product.

It will be realized that this procedure will also work for negative numbers in "ones complement" form, provided that further conventional corrections for addition of negative numbers are employed where the multiplicand is negative. The rationale for these further corrections can be seen by noting that if the final product is to be in 1'c complement form, all additions but one must be of 2's complement quantities. The further corrections therefore consist of adding an extra 1 at the bottom bit of the multiplicand for each 1 in the multiplier *after* the first (end-around carry correcton).

Referring now to Figure 1, there are first (A) and second shift (Y) registers 10 and 12, and a storage (B) register 14, each of 4 bits. Each register has its bit positions labelled as shown, with A4, B4 and Y4 respectively being the most significant bit positions. Bit position Y1 is coupled to bit position A4, so that registers 10 and 12 may be regarded as a single shift register having 8 bit positions. The contents of registers 10 and 12 can be shifted bit by bit from left to right in accordance with the shift output signal provided via shift control logic 16. Shift control logic 16, which is shown in more detail in Figure 8, receives its control signals from the least significant bit position A1 of register 10, from counter 18 and from the sign of the first number via register or flip-flop 20.

A counter 18 is normally preset to the number, 4 of bit locations in register 10 but may be preset to the number of bits in the multiplier. When the counter is decremented to zero, this partially disables logic 16. At this time, the apparatus has performed the entire multiplication sequence and the result thereof will be found in registers 12 and 10.

Add control logic 22, as shown in detail in Figure 9, is coupled to add the contents or a portion thereof of Y register 12 and the contents or a portion thereof of B register 14 via adder 24. Logic 22 is also coupled to simply pass the sign of the multiplicand or second number in B register 14, the sign being in the most significant bit location B4, to the most significant bit location Y4 of register 12. The typical add operation to be performed will add the bits B4, B3, B2 and B1 to bits Y4, Y3, Y2 and Y1 and place the result thereof in locations Y4, Y3, Y2 and Y1 respectively. Shift control logic 16 is coupled with add control logic 22 by several control lines as specifically illustrated in Figures 8 and 9 and as shown by the dotted line of Figure 1.

The operation of the apparatus of Figure 1 is illustrated by the flow diagram of Figure 2. Examples illustrating the operation of such apparatus for the multiplication of two numbers 5 and 6 in the four different sign combinations (the twos complement form being used for negative numbers) are shown in Figures 4-7 which are self-explanatory when taken together with the format illustrated in figure 3. In Figures 4-7, line 2 shows the contents of B register 14, which remain the same throughout the operation.

The contents of registers 10 and 12 change in operation and are shown in the other lines. The magnitude of the result is indicated by a rectangle in the last line of each of these figures, with the sign being the left most bit outside the rectangle. The details of logic 16 and 22 are discussed hereinafter.

Now referring to Figure 2, the multiplication process starts at block 30. Initially, the Y register 12 is reset to include all binary ZEROs as indicated in block 32. Further, initially, the sign of the first number, which is in position A4 of register 10, is clocked into flip-flop 20 as indicated in block 34. Initial conditions are that the register 10 and register 14 respectively have stored therein the multiplier and multiplicand, and following the operation indicated by blocks 32 and 34, the sign of the first number is stored in register 20 and each of the bits of register 12 is 0.

The operation is dependent upon the sign of the multiplier as well as the "current" multiplier bit in position A1 of register 10. (It can be seen that while position A1 must be used initially to indicate this bit, in order to increase the speed of the apparatus position A2 may be utilized thereafter for this purpose). For initial conditions, the apparatus, independent of the sign of the first number in register 10, determines whether a 1 is in position A1. If there is a 0 in position A1, then the contents of registers 10 and 12 are shifted right one position. These shifts, as well as the shifts to be described hereinafter, will cause counter 18 to decrement and accordingly such shifts, the total number of which will be 4, are counted to determine when multiplication is complete. Thus, if initially there are 0's in positions A1 and A2 and 1 in position A3, then there will be two such shifts without any other operation until the 1 is found as shifted from position A3. It will be noted that position A4 includes the sign of the first number; if the bit in position A4 is 1, then the number is negative while if the bit in position A4 is 0 then the number is positive.

Thus block 36 of Figure 2 tests whether or not the bit in position A1 is 1. If the answer is No, block 38 is entered by which the bits in register 10 are shifted right and the 0 in location Y1 of register 12 is shifted into location A4 of register 10. (In practice, the implementation is simplified by simply shifting the entire contents of registers 12 and 10 toward the least significant bit position). In addition, each time there is a shift, counter 18 is decremented. At this point block 36 tests whether the bit in position A1 is a binary ONE. If the answer is Yes, then block 40 is entered; this block is identical to block 38, but is followed by block 42, by which the contents of the B register 14 are loaded into the Y register 12. This is accomplished via control logic 22 and adder 24. The loading of the entire contents of the register 14 into the register 12 may be accomplished without an adder by simply providing gating structure enabled by the in position A1 and in response to the shift operation provided in block 40. Thus at this point, the contents of register 10 have been shifted at least once and the leftmost position thereof includes at least one 0 from register 12 and in addition, the contents of register 14 have been loaded into register 12. It will be realized that a further test (not shown) may be made at the output side of block 38, namely, has the counter 18 counted down to 0? If the answer were No, then block 36 would be reeentered; however, if the answer were Yes, the result of the multiplication would be zero and the process may exit at this point.

Following block 42, block 44 is entered, which tests whether or not the first number A which was in register 10 is a positive number; this is indicated by a 0 in flip-flop 20. Depending upon the answer to block 44, then either one of two sequences is initiated, dependent on the bit in position A1 of register 10, as indicated in blocks 46 and 48.

To summarize the operations following the determinations made in blocks 46 and 48, if the first number A is positive (as indicated by flip-flop 20) then if position A1 of register 10 is 0, the shift operation described with respect to blocks 38 and 40 a complete shift of the contents of register 12, and a loading of the sign of the second number as indicated in position B4 of register 14 into position Y4 of register 12 takes place. If the first number A is a positive number, and the position A1 of register 10 is 1, there will not only be such a shift and load operation, but there will also be an add operation of the contents of the B register 14 with B4 and bit positions Y3, Y2 and Y1 of Y register 12 following which the result is placed in register 12. If the first number A in register 10 was initially a negative number (as indicated by flip-flop 20), then the operation will be the same as just indicated except that the shift and load sign operation will be responsive to 1 in position A1 of register 10 and the shift, load and add operation will be responsive to a 0 in position A1.

Now in more detail, the first example to be taken is that A is a positive number. At this point, block 48 is entered and if the bit position A1 of register 10 is 0, so that the answer to block 48 is No, then block 50 will be entered. A shift operation will be performed, this necessarily being a full shift of the Y and A registers. In addition, the counter 18 is decremented. Further, following the shift operation, position Y4 of register 12 has loaded therein the sign of the

second number B in register 14, as indicated in block 52. At this point, block 59 tests whether or not the count of counter 18 is 0. If the answer is No, then block 44 is re-entered. If the answer is Yes, then block 66 is entered.

If the answer to block 48 was a Yes, i.e., the bit in position A1 is a 1, then block 56 will be entered. Block 56 contains the same operation as provided in block 50. The add control logic 22 assumes that the entire contents of register 12, as well as register 10, have been shifted. Block 57 is then entered, at which time the sign of the second number in position B4 is loaded into position Y4 of register 12, as was the case in block 52. Following this, block 58 is then entered at which time the following operation takes place. The contents of B register 14 are added to the contents of register 12, which of course contains a copy in position Y4 of the bit in position B4 of the B register. Block 60 follows in which the result of the addition is placed in positions Y4, Y3, Y2 and Y1 respectively of register 12. After block 60, block 54 tests whether or not counter 18 has a count of 0. If the answer is No, then block 44 is again entered; if the answer is Yes, then block 66 is entered.

If the first number A is a negative number, as indicated by a No answer in block 44, then block 46 would have been entered. Again block 46 tests whether or not bit position A1 of register 10 has a 0 or a 1. The sequences of blocks 50 and 52 or 56, 57, 58 and 60 are followed as before, except that the Yes and No outputs from block 46 correspond to the No and Yes outputs from block 48. It is evident that a test (not shown) may be made at the output side of block 42 of whether or not the counter has been decremented to 0 by the operations of blocks 38 and 40, so that block 66 may be entered directly without entering block 44. Such a condition may occur for example if a 1000 bit pattern was initially in register 10.

Once the counter 18 is equal to 0, this indicates that the number of total shifts performed is equal to the number of bits in the first number or (if the additional logic required to determine the number of bits in the first number is not provided) is 4 shifts, i.e. the number of bit locations in register 10. As a result of a Yes answer from block 54, block 66 is entered by which the shift operation, as indicated in block 50, is again repeated so as to provide the magnitude of the result in positions Y3 to Y1 and A4 to A1 of registers 12 and 10 respectively. Following this, block 68 is entered by which the sign of the second number is loaded into position Y4 of register 12. Block 70 is then entered and the result of the multiplication is found in registers 12 and 10. The sign is in position Y4 of register 12, with the most significant bit of the magnitude of the result being in position Y3 of register 12 and the least significant bit being in position A1 of register 10. The result needs no further processing if the first number A originally loaded into register 10 was a positive number. If however the first number A loaded into register 10 was a negative number, then the actual result of multiplication will be the so-called twos complement of the contents, including sign and magnitude of the values stored in registers 12 and 10. Accordingly, with the result in registers 12 and 10, the multiplication technique is exited at block 72.

If desired, this number, if in twos complement form, may be converted to normal form by a standard technique, e.g., starting at the right and keeping all trailing 0's and the first 1 reached, and from the next digit on, complementing all bits.

Now referring to Figure 8, the shift contrfol logic 16 is shown. Bit position A1 of register 10 is coupled to the input of a comparator 100 which is enabled by the output of OR gate 102. Under initial conditions, flip-flop 104 is set and will enable comparator 100 via OR gate 102 and will also enable gate 106 and partially enable gate 108. The output of comparator 100 is coupled to several inputs including OR gate 110 which is coupled to generate a signal at its output enabling AND gate 106 whenever comparator 100 is enabled. For either condition (0 or 1 in A1), a shift signal will be provided via OR gate 112 and AND gate 114, which is enabled unless counter 18 has counted down to 0. An extra shift is also provided however, as indicated in block 66 of Figure 2, after the count of counter 18 reaches 0.

At the same time that gate 106 is fully enabled, gate 108 will be fully enabled if in fact a 1 was detected by compartor 100. In response to this, the flip-flop 104 will be reset and accordingly will inhibit any further enabling of gate 106 or gate 108. In addition, the 1 initially detected by comparator 100 will via inverter 120, disable gate 118 from enabling compartor 100. The operation just described will be repeated if a 0 is detected by comparator 100. Further, the output of gate 108 is coupled to add control logic 22 so as to command, by means of the signal on line 80, the loading of the entire contents of register 14 into the Y register 12 as indicated in block 42 of Figure 2. Thus the operation up to block 42 of Figure 2 for the shift control logic 16 has been covered.

For each shift after these initial operations have been made, and so long as counter 18 has not counted down to 0, AND gate 122 is enabled so as to enable via OR

gate 102, comparator 100 to determine whether a 1 or a 0 is in position A1 of register 10. If flip-flop 20 gives a 1 output, indicating that the first number A isnegative, then if there is a 1 in position A1 of register 10, AND gate 124 will be fully enabled so as to cause the shift operation, and in response to the enabling of gate 124, OR gate 126 provides the loading of B4 into Y4 by means of a signal on line 82. This signal is provided to logic 22 so as to provide the operation of blocks 52 and 57. If the first number A is a negative number and the bit in position A1 is 0, then AND gate 128 will be fully enabled causing the shift as indicated in block 56 of Figure 2, and in response thereto via OR gate 126 enabling the loading of B4 and Y4 by means of a signal on line 82. In response to this, AND gate 130 enables the adding of B4-B1 to Y4-Y1 in response to the signals received at OR gate 129. This addition is enabled by a signal on line 82. Also, as indicated in block 60, the add control logic 22 places the result of the addition in the Y register. This covers the operation when the first number is negative. It is noted that lines 80, 81 and 82 are the control lines represented by the dotted line in Figure 1.

When the first number A is a positive number, as indicated by a 0 from flip-flop 20, if comparator 100 indicates a 1 in position A1 of register 10, then AND gate 132 is fully enabled causing the shift signal to occur at the output of AND gate 114. Further, the signal at the output of gate 132 causes via OR gate 126 the load B4 into Y4 operation to take place following which B4-B1 are added to Y4-Y1 by the enabling of AND gate 130. Following this, the result is placed in the contents of the Y register 12. If, on the other hand, when the first number A is a positive number and a 0 is indicated by comparator 100, then AND gate 134 is fully enabled thereby enabling the shift operation, as indicated in block 50 of Figure 2 and further via OR gate 126, the contents of position B4 is loaded into position Y4. This thus covers the operation when the first number is a positive number.

When counter 20 has finally counted down to 0, the shift operation of block 66 is enabled via inverter 120 following which a load operation is provided as indicated in block 68 of Figure 2. This is accomplished since the 1 normally provided by counter 20 changes to 0 when the count equals 0, at which time a 0 is at the input of inverter 136 and a 1 at the output thereof thereby enabling the shift operation via OR gate 112. This also enables the loading of the contents of position B4 into position Y4 via logic 22 in response the signal provided via OR gate 126 on line 82.

Following this operation, the result of the multiplication is in registers 12 and 10. If, in fact, the sign indicated in flip-flop 20 is negative, the result is the twos complement of the contents of registers 12 and 10. The taking of the twos complement is well-known, and the enabling of AND gate 140 causes the twos complementing operation to take place with the result thereof placed either in registers 12 and 10 or in another storage device as may be desired.

WHAT WE CLAIM IS:—

1. Multiplying apparatus comprising a multiplier register, a multiplicand register, a product register, and means arranged to repeatedly shift the contents of the product register and add into it selectively the contents of the multiplicand register and zero under the control of successive bits of the contents of the multiplier register, the signs of the multiplier and multiplicand being indicated by the topmost bit of each, wherein at each addition following the first 1 bit in the multiplier the sign bit of the multiplicand is inserted into the product register at the bit position corresponding to the position of the topmost bit of the multiplicand on that addition, and the multiplier is converted, bit by bit, into complement form as its bits are used to control the additions if its sign is negative.

2. Apparatus according to claim 1 including means which complement the contents of the product register after the additions have been completed if the sign of the multiplier is negative.

3. Apparatus according to either previous claim wherein the multiplier register is part of the product register.

4. Multiplying apparatus substantially as herein described with reference to the accompanying drawings.

M. G. HARMAN,
Chartered Patent Agent.

Printed for Her Majesty's Stationery Office by The Tweeddale Press Ltd., Berwick-upon-Tweed. 1979.
Published at the Patent Office, 25 Southampton Buildings, London, WC2A 1AY, from which copies may be obtained.

gate 102, comparator 100 to determine whether a 1 or a 0 is in position A1 of register 10. If flip-flop 20 gives a 1 output, indicating that the first number A isnegative, then if there is a 1 in position A1 of register 10, AND gate 124 will be fully enabled so as to cause the shift operation, and in response to the enabling of gate 124, OR gate 126 provides the loading of B4 into Y4 by means of a signal on line 82. This signal is provided to logic 22 so as to provide the operation of blocks 52 and 57. If the first number A is a negative number and the bit in position A1 is 0, then AND gate 128 will be fully enabled causing the shift as indicated in block 56 of Figure 2, and in response thereto via OR gate 126 enabling the loading of B4 and Y4 by means of a signal on line 82. In response to this, AND gate 130 enables the adding of B4-B1 to Y4-Y1 in response to the signals received at OR gate 129. This addition is enabled by a signal on line 82. Also, as indicated in block 60, the add control logic 22 places the result of the addition in the Y register. This covers the operation when the first number is negative. It is noted that lines 80, 81 and 82 are the control lines represented by the dotted line in Figure 1.

When the first number A is a positive number, as indicated by a 0 from flip-flop 20, if comparator 100 indicates a 1 in position A1 of register 10, then AND gate 132 is fully enabled causing the shift signal to occur at the output of AND gate 114. Further, the signal at the output of gate 132 causes via OR gate 126 the load B4 into Y4 operation to take place following which B4-B1 are added to Y4-Y1 by the enabling of AND gate 130. Following this, the result is placed in the contents of the Y register 12. If, on the other hand, when the first number A is a positive number and a 0 is indicated by comparator 100, then AND gate 134 is fully enabled thereby enabling the shift operation, as indicated in block 50 of Figure 2 and further via OR gate 126, the contents of position B4 is loaded into position Y4. This thus covers the operation when the first number is a positive number.

When counter 20 has finally counted down to 0, the shift operation of block 66 is enabled via inverter 120 following which a load operation is provided as indicated in block 68 of Figure 2. This is accomplished since the 1 normally provided by counter 20 changes to 0 when the count equals 0, at which time a 0 is at the input of inverter 136 and a 1 at the output thereof thereby enabling the shift operation via OR gate 112. This also enables the loading of the contents of position B4 into position Y4 via logic 22 in response the signal provided via OR gate 126 on line 82.

Following this operation, the result of the multiplication is in registers 12 and 10. If, in fact, the sign indicated in flip-flop 20 is negative, the result is the twos complement of the contents of registers 12 and 10. The taking of the twos complement is well-known, and the enabling of AND gate 140 causes the twos complementing operation to take place with the result thereof placed either in registers 12 and 10 or in another storage device as may be desired.

WHAT WE CLAIM IS:—

1. Multiplying apparatus comprising a multiplier register, a multiplicand register, a product register, and means arranged to repeatedly shift the contents of the product register and add into it selectively the contents of the multiplicand register and zero under the control of successive bits of the contents of the multiplier register, the signs of the multiplier and multiplicand being indicated by the topmost bit of each, wherein at each addition following the first 1 bit in the multiplier the sign bit of the multiplicand is inserted into the product register at the bit position corresponding to the position of the topmost bit of the multiplicand on that addition, and the multiplier is converted, bit by bit, into complement form as its bits are used to control the additions if its sign is negative.

2. Apparatus according to claim 1 including means which complement the contents of the product register after the additions have been completed if the sign of the multiplier is negative.

3. Apparatus according to either previous claim wherein the multiplier register is part of the product register.

4. Multiplying apparatus substantially as herein described with reference to the accompanying drawings.

M. G. HARMAN,
Chartered Patent Agent.

Printed for Her Majesty's Stationery Office by The Tweeddale Press Ltd., Berwick-upon-Tweed, 1979.
Published at the Patent Office, 25 Southampton Buildings, London, WC2A 1AY, from which copies may be obtained.