The multiplier design is based on the multiplication schema shown in Figure 1 for the 4×4 bit case.[6-11] The partial product terms were formed by ANDing each multiplicand bit with each multiplier bit. The partial products X_iY_3 and $X_3Y_i =$ for $i = 0, 1, 2$ contain information concerning the sign of the operands.

$$
\begin{array}{rcccccccc}
 & & & Y_3 & Y_2 & Y_1 & Y_0 & & \\
 & & \times & X_3 & X_2 & X_1 & X_0 & & \\
\hline
 & & 1 & \overline{X_0Y_3} & X_0Y_2 & X_0Y_1 & X_0X_0 & & \\
 & & \overline{X_1Y_3} & X_1Y_2 & X_1Y_1 & X_1Y_0 & & & \\
 & \overline{X_2Y_3} & X_2Y_2 & X_2Y_1 & X_2Y_0 & & & & \\
+1 & X_3Y_3 & \overline{X_3Y_2} & \overline{X_3Y_1} & \overline{X_3Y_0} & & & & \\
\hline
P_7 & P_6 & P_5 & P_4 & P_3 & P_2 & P_1 & P_0 &
\end{array}
$$

Figure 1. 4×4 bit multiplication algorithm

The required subtraction of these product terms is realized by complementing the terms containing sign information and by adding '1's to the fifth and eighth columns ($n + 1$ and $2n$ in the general case). The final product is computed by adding all partial product terms. The following proof demonstrates the validity of the above assertions and illustrates the way that multiplication proceeds.