The 1616 accepts all types of two's-complement and unsigned number systems as input data formats. Included are unsigned integer, unsigned fractional, signed integer, and signed fractional. The device multiplies in all six combinations of these four number systems. But representations of numbers in each system differ in the weight assigned to each bit and the placement of the binary point.

Unsigned integers are represented by

$$X = \sum_{i=0}^{15} x_i 2^i$$

unsigned fractions by

$$X = \sum_{i=0}^{15} x_i 2^{-(16-1)}$$

signed integers by

$$X = -x_{15}\, 2^{15} + \sum_{i=0}^{14} x_i 2^i$$

and signed fractions by

$$X = -x_{15} + \sum_{i=0}^{14} x_i 2^{-(15-1)}$$

where x_0 represents the least significant bit. The binary point is placed in the same location for both signed and unsigned integers, and the weight of individual bits—with the exception of x_{15}—is the same in both systems. The binary point differs by 1 bit position between signed and unsigned fractional numbers. In addition the binary point in signed fractional notation is not at the end of the bit stream. The 1616 prevents a problem that occurs when the sign of the product of two signed fractions is between PR_{30} and PR_{29} or when the sign of the product of a signed fraction multiplied by a signed or unsigned integer lies between PR_{15} and PR_{14}. The multiplier allows the user to shift the product one position to the left and to put a zero in the LSB position.

The 1616 contains circuitry to realign the binary point when the products of multiplication yield ambiguous results. For example, if the inputs to the multiplier are $C000_{16}$ and -2^{14}, or if both are -2^{-1}, the product in integer multiplication is 10000000_{16}. In this case, $C000_{16}$ and -2^{14} are treated as signed integers, and -2^{-1} is a signed fraction. As shown in Table 1, the 1616 accommodates eight different input formats, depending on the op code assigned through OP_2–OP_0. In that way, the chip can remove the ambiguities of identical results for dissimilar products.

If both multiplier inputs are fractional, the output can yield 2 sign bits—a result that makes the mathematics difficult if the product must be added to another fractional number. In this situation, the 1616 shifts the result 1 bit to the left to realign the binary point correctly. Table 2 illustrates a uniform multiplication and the product obtained with various op codes.

Table 2: Multiplication examples

Op Code	X	Y	RND/MN 0/0	RND/MN 0/1	RND/MN 1/0	RND/MN 1/1
0	AAAA	5555	38E31C72	C71CE38E	38E39C72	C71D638E
1	AAAA	5555	38E31C72	C71CE38E	38E39C72	C71D638E
2	AAAA	5555	E38E/1C72	1C71/E38E	E38E9C72	1C72638E
3	AAAA	5555	E38E1C72	1C71E38E	E38E9C72	1C72638E
4	AAAA	5555	C71C38E4	38E3C71C	C71CB8E4	38E4471C
5	AAAA	5555	71C638E4	8E39C71C	71C6B8E4	8E3A471C
6	AAAA	5555	C71C9C72	38E3638E	C71CDC72	38E4238E
7	AAAA	5555	C71C38E4	38E3C71C	C71CB8E4	38E4471C

Table 1: Multiplier input-output format

Op code	O_{p_2}	O_{p_1}	O_{p_0}	Input format	Output format	Round operation
0	L	L	L	X unsigned times Y unsigned	Unsigned output	Add 1 to PR_{15}
1	L	L	H	X unsigned times Y signed	Single signed	Add 1 to PR_{15}
2	L	H	L	X signed times Y unsigned	Single signed	Add 1 to PR_{15}
3	L	H	H	X signed times Y signed	Single signed	Add 1 to PR_{15}
4	H	L	L	X signed times Y signed	Single signed, fractionally adjusted	Add 1 to PR_{14}
5	H	L	H	X unsigned times Y signed	Single signed, fractionally adjusted	Add 1 to PR_{14}
6	H	H	L	X signed times Y signed	Double Signed	Add 1 to PR_{14}
7	H	H	H	X signed times Y unsigned	Single signed, fractionally adjusted	Add 1 to PR_{14}

*The round is applied before the shift occurs. All rounds have the affect of incrementing the most-significant half if the MSB (disregarding double sign bit) is a binary one.