Example #1: Unsigned Multiplication

Show the steps for the multiplication of 10101001b and 00111011b (**unsigned**) using **Hardware Design 3** (http://www.cs.pitt.edu/~childers/CS0447/lectures/numbers3.pdf). Here 10101001b is the multiplicand and 00111011b is the multiplier.

REMEMBER: Do not ignore overflow. Push the overflow bit ('1') in during shift right.

10101001b = 169
00111011b = 59

So the answer should be 169 * 59 = 9971

9971 = 0010 0110 1111 0011b <----- this is the result we expect from multiplication

Now,
M = 1010 1001

Iteration	Step	Product (16 bits)
setup		0000 0000 0011 1011
1	M 1: P = P + M Shift Right	1010 1001 1010 1001 0011 1011 0101 0100 1001 1101
2	M 1: P = P + M Shift Right	1010 1001 1111 1101 1001 1101 0111 1110 1100 1110
3	0: Shift Right	0011 1111 0110 0111
4	M 1: P = P + M Shift Right	1010 1001 1110 1000 0110 0111 0111 0100 0011 0011
5	M 1: P = P + M Shift Right	1010 1001 1 0001 1101 0011 0011 (overflow) 1000 1110 1001 1001 (ov included)
6	M 1: P = P + M Shift Right	1010 1001 1 0011 0111 1001 1001 (overflow) 1001 1011 1100 1100 (ov included)
7	0: Shift Right	0100 1101 1110 0110
8	0: Shift Right	0010 0110 1111 0011

The result is: `0010 0110 1111 0011`

Example #2: Signed Multiplication

Show the steps for the multiplication of 10101001b and 00111011b (**signed**) using **Booth's algorithm** (http://www.cs.pitt.edu/~childers/CS0447/lectures/numbers3.pdf).
Here 10101001b is the multiplicand and 00111011b is the multiplier.

REMEMBER: Ignore overflow, but extend the sign (If the sign is 0, then push 0 during shift right. If the sign is 1 then push 1 during shift right).

10101001b = -87
00111011b = 59

So the answer should be -87 * 59 = -5133

+5133 = 0001 0100 0000 1101b
-5133 = 1110 1011 1111 0011b <----- this is what we expect as a result from Booth's algorithm

Now,
M = 1010 1001
-M = 0101 0111

Iteration	Step	Product (17 bits)
setup		0000 0000 0011 1011 0
1	- M 10: P = P – M Shift Right	0101 0111 0101 0111 0011 1011 0 0010 1011 1001 1101 1 (sign 0 extend)
2	11: Shift Right	0001 0101 1100 1110 1 (sign 0 extend)
3	M 01: P = P + M Shift Right	1010 1001 1011 1110 1100 1110 1 1101 1111 0110 0111 0 (sign 1 extend)
4	-M 10: P = P - M Shift Right	0101 0111 1 0011 0110 0110 0111 0 (ignore ov) 0001 1011 0011 0011 1 (sign 0 extend)
5	11: Shift Right	0000 1101 1001 1001 1 (sign 0 extend)
6	11: Shift Right	0000 0110 1100 1100 1 (sign 0 extend)
7	M 01: P = P + M Shift Right	1010 1001 1010 1111 1100 1100 1 1101 0111 1110 0110 0 (sign 1 extend)
8	00: Shift Right	1110 1011 1111 0011 0 (sign 1 extend)

The result is: 1110 1011 1111 0011