A Low Power Radix-4 Dual Recoded Integer Squaring Implementation For Use in Design of Application Specific Arithmetic Circuits

Jason Moore, Mitchell A. Thornton, David W. Matula
Department of Computer Science and Engineering
Southern Methodist University
Dallas, Texas

Abstract: We introduce an implementation of a radix 4 dual recoding procedure for the squaring operation of an *n*-bit number which reduces the number of bit product terms employed in the previously known squaring methods obtained by either Booth radix-4 recoded multiplication or by radix 2 squaring. Several other squaring algorithms have been developed such as [WSMB99], [YW01], and [SNC01]. Employing the dual recoded radix-4 procedure for design of a squaring circuit introduces a significant reduction in power and area. Architecturally, radix-4 dual recoded squaring uses only the 1's complement representation which allows for a simpler PPG structure as compared to the 2's complement representation required for Booth radix-4 multiplication.

1. Background

The radix 4 recoding procedure utilizes Booth recoding. Let $P = d_{\lfloor \frac{n}{2} \rfloor} d_{\lfloor \frac{n}{2} \rfloor - 1} \cdots d_0$ with $d_i \in \{-2,-1,0,1,2\}$ be the Booth recoded radix 4 representation of $Q = q_{n-1} q_{n-2} \cdots q_0$ [Bo51, Ru75]. It is important to recall how Booth radix-4 digit d_i of P is determined by the three bits $q_{2i+1} q_{2i} q_{2i-1}$ of Q as can be seen from Table 1. Bit q_{-1} is considered a 0.

Binary	Booth Digit
100	$\bar{2}$
110, 101	$\bar{1}$
000, 111	0
001, 010	1
011	2

Table 1: Radix-4 Booth Recodings [Bo51, Ru75]

$Q = 1100\ 0101\ 1000\ 1011_2$	
11	$\bar{1}$
101	$\bar{1}$
001	1
100	$\bar{2}$
011	2
010	1
000	0
110	$\bar{1}$
001	1
$P = (1\,\bar{1}\,0\,1\,2\,\bar{2}\,1\,\bar{1}\,\bar{1})_4$	

Example 1: Shows how $Q = 1100\ 0101\ 1000\ 1011_2$ is recoded to be represented by $P = (1\,\bar{1}\,0\,1\,2\,\bar{2}\,1\,\bar{1}\,\bar{1})_4$

The use of Booth radix-4 recoding for 16-bit integer multiplication for $Q \times P$ requires 88 entries and 9 rows as illustrated in Figure 2 for the *n*-bit product. This is a considerable reduction of the 136 entries and 16 rows for the radix-2 integer partial product array but provides no additional benefit for the squaring

operation. A radix-2 squaring circuit was described in [PBD97] resulting in 72 entries as illustrated in Figure 3.

$1100\,0101\,1000\,1011_2$ $\times\ (\bar{1}\ 0\ \ 1\ 2\ \bar{2}\ 1\ \ \bar{1}\ \ \bar{1})_4$	(selector digits)
0011 1010 0111 0100 1110 1001 1101 00 0101 1000 1011 0011 1010 00 0001 0110 0010 11 0000 00	$\bar{1}$ $\bar{1}$ 1 $\bar{2}$ 2 1 0 $\bar{1}$
0100 0000 1000 0101	2's comp bits
0011 1001 0111 1001	Square

Figure 2: Booth Recoded Radix 4 Multiplication

$(1100\ 0101\ 1000\ 1011)^2$ [1000 1011]	(selector bits)
1000 1011 0001 01o1 0001 0110 0010 o1 0000 0000 00o0 0101 1000 o1 0000 00o0 0000 o0 00o0 o1	1 1 0 1 0 0 0 1
0011 1001 0111 1001	Square

Figure 3: Booth Recoded Radix-2 Squaring

2. Algorithm

The radix-4 dual recoded squaring algorithm determines the Booth digits in a right-to-left manner i.e. starting from the least significant bit and moving toward the most significant bit. Let P_i be the integer formed by shifting the radix 4 digit string right i places deleting the low order i digits obtaining

$$P_i = d_{\lfloor \frac{n}{2} \rfloor} d_{\lfloor \frac{n}{2} \rfloor - 1} \cdots d_i = \sum_{j=i}^{\lfloor \frac{n}{2} \rfloor} d_j 4^{j-i}$$

for $i = 0,1,\cdots \lfloor \frac{n}{2} \rfloor$, with $P_{\lfloor \frac{n}{2} \rfloor + 1} = 0$ and $P_0 = P = Q$. Since then $P_i = 4P_{i+1} + d_i$ and $P_i^2 = 16P_{i+1}^2 + (8P_{i+1} + d_i)d_i$ for $i = 0,1,\cdots, \lfloor \frac{n}{2} \rfloor$, we obtain the following.

Observation 1:

$$Q^2 = \sum_{i=0}^{\lfloor \frac{n}{2} \rfloor - 1} \left(8(Q_{2i+2} + q_{2i+1})d_i + d_i^2\right)16^i$$

Definition 1: Let $Q = q_{n-1}q_{n-2}\cdots q_0$ with $Q_i = \left\lfloor \frac{Q}{2^i} \right\rfloor = q_{n-1}q_{n-2}\cdots q_i$ for $0 \le i \le n-1$ and let $d_i \in \{-2,-1,0,1,2\}$ be the i^{th} digit of the Booth recoded radix 4 representation $Q = P = d_{\lfloor \frac{n}{2} \rfloor} d_{\lfloor \frac{n}{2} \rfloor - 1} \cdots d_0$. Then the i^{th} radix-4 *partial square* is $8(Q_{2i+2} + q_{2i+1})d_i + d_i^2$ where $8(Q_{2i+2} + q_{2i+1})$ is the i^{th} multiplicand factor for radix-4 dual recoded squaring and d_i is the i^{th} *recoded radix 4 select digit*.

Recall that q_{2i+1} is effectively the sign bit of the recoded digit d_i, so we obtain the partial square identity

$$8(Q_{2i+2} + q_{2i+2})d_i + d_i^2 = (-1)^{q_{2i+1}} 8(Q_{2i+2} + q_{2i+2})|d_i| + d_i^2.$$

It is important to observe that the 2's complement of $Q_{2i+2} + q_{2i+1}$ reduces to the sign extended 1's complement of Q_{2i+2}, as formally summarized in the following.

Theorem 1: Let $P = d_{\lfloor \frac{n}{2} \rfloor} d_{\lfloor \frac{n}{2} \rfloor - 1} \cdots d_0$ be the radix-4 dual recoded representation of $Q = q_{n-1}q_{n-2}\cdots q_0$. Let Q^*_{2i+2} be the conditionally 1's complemented sign extended leading bits of $Q_{2i+2} = \left\lfloor \frac{Q}{2^{i+2}} \right\rfloor$ given for $i = 0,1,\cdots \lfloor \frac{n}{2} \rfloor - 1$ by

$$Q^*_{2i+2} = \begin{cases} q_{n-1}q_{n-2}\cdots q_{2i+2}, & \text{for } q_{2i+1} = 0 \\ \cdots 11\bar{q}_{n-1}\bar{q}_{n-2}\cdots\bar{q}_{2i+2}, & \text{for } q_{2i+1} = 1 \end{cases}$$

Then

$$Q^2 = \sum_0^{\lfloor \frac{n}{2} \rfloor} \left(8Q^*_{2i+2}|d_i| + d_i^2\right)16^i.$$

From Theorem 1 we observe that the PPG's for recoded radix 4 squaring are simpler than for a Booth recoded radix-4 multiplier since the complements are simply 1's complements. This means that for the *n*-bit integer radix-4 square: (1) we need to employ only the $\lfloor \frac{n}{4} \rfloor$ low order half of the Booth-4 select digits in forming the radix-4 dual recoded square array, (2) No complement bit row is needed as in Booth-radix 4 multiplication, and (3) no sign extension of the partial squares are needed for the *n*-bit square.

Figure 4 illustrates the radix-4 dual recoded array for computing $Q^2 = (1100\,0101\,1000\,1011)^2 \bmod 2^{16}$. There are only 40 entries in Figure 4 compared to the 72 entries for radix 2 illustrated in Figure 3 and compared to the 88 entries for Booth recoded radix 4 multiplication illustrated in Figure 2.

1100 0101 |10|00| 10|11 [$\bar{2}$ 1 $\bar{1}$ $\bar{1}$]	Booth-4 select digits
0111 0100 1110 10o1 1101 0011 10o1 1011 00o1 01o0	$\bar{1}$ $\bar{1}$ 1 $\bar{2}$
0011 1001 01 11 10 01	Square

Figure 4: Radix-4 Dual Recoded Squaring

3. Implementation

Based on the properties of Theorem 1, an efficient Partial Square Generator (PSG) was designed. Figure 5 is a Synopsys screen shot of an 8-bit PSG. The dataflow through the circuit is as follows:

1. All the bits of the input, *X*, except for the 3 least significant bits are conditionally negated if x_2 is 1,
2. The output from step 1 is left shifted by 1 if x_2 is equal to x_0,

3. If the three least significant bits of *X* are 0, $Y \equiv 0$ else all but the 3 least significant bits of *Y* are the output from step 2 and

$$y_2 = \overline{\overline{(x_0 \oplus x_1)}\,\overline{(x_0 \oplus x_2)}}\,\overline{(x_0 \oplus x_1)}$$

$$y_1 = 0$$

$$y_0 = \overline{\overline{(x_0 \oplus x_1)}\,\overline{(x_0 \oplus x_2)}} + \overline{(x_0 \oplus x_1)}$$

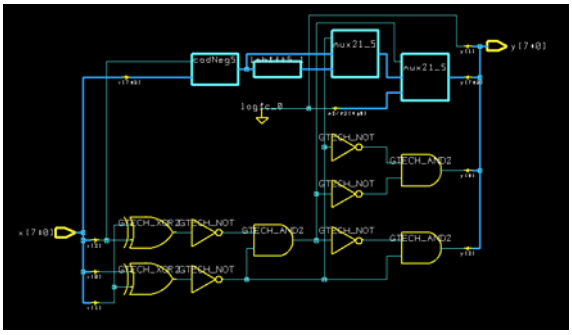

Figure 5: A Synopsys screen shot of an 8 bit PSG

Using the PSG design above, the architecture of the squaring unit can be viewed in at least the two following methods depicted in figures 5. Figure 5 shows the architecture view of building the circuit independent of smaller versions of the circuit. The architecture in Figure 5 is constructed from the following components: $\lceil \frac{n}{2} \rceil$ PSG's of bit sizes *n*, *n*-2, …, 2, and an adder tree of height of $\log_x n$ where *x* is the bit size of the adders used in the adder tree.

4. Results

The radix-4 dual recoded squaring circuit and a general purpose multiplier were both implemented in verilog and mapped to OSU standard cell library [SCWH07]. Both circuits were constrained to run within a 20ns clock-edge and were implemented for 16, 32, and 64 bit-widths. The charts in Figures 7-9 show a substantial gain in power, leakage power, and area for our customized squaring circuit compared to a multiplier circuit.

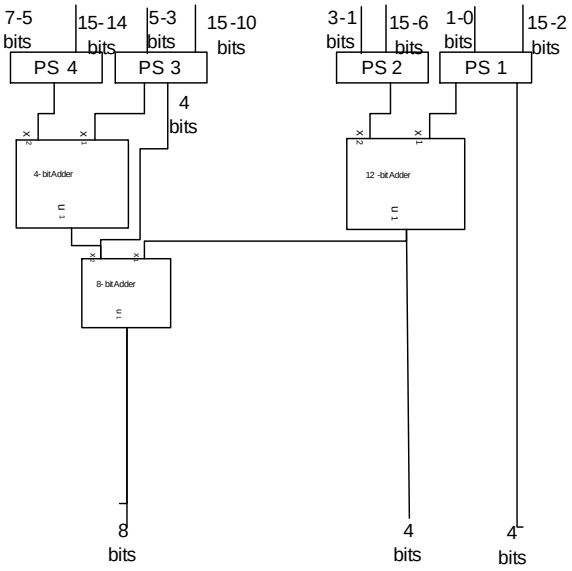

Figure 5: Datapath of 16-bit input/16-bit output Radix-4 Dual Recoded Squaring Circuit

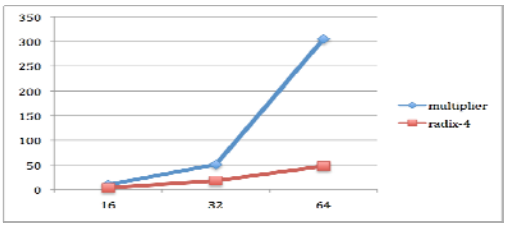

Figure 7: Power (mW) vs Bit-Width chart

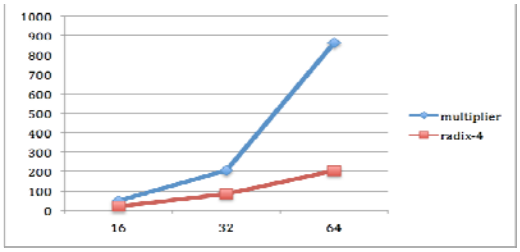

Figure 8: Leakage Power(nW) vs Bit-Width Chart

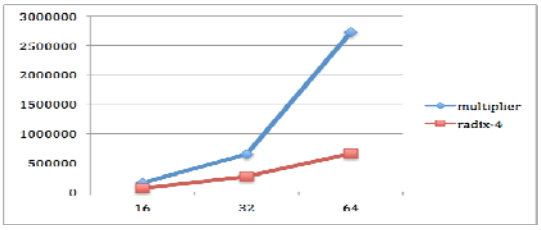

Figure 9: Area vs Bit-Width chart

5. References

[Bo51] A.D. Booth, "A Signed Binary Multiplication Technique," *Quarterly J. Mechanical and Applied Math.*, vol. 4, no. 2, pp. 236-240, 1951.

[Ma07] D.W. Matula, "A Radix-4 dual recoded Squaring Procedure For Use in Design of Application Specific Arithmetic Circuits", SRC Report, 4/12/2007

[PBD97] A. Pirson, J.-M. Bard, and M. Daoudi, Squaring Circuit for Binary Numbers, United States Patent, No. 5,629,885, May 13, 1997.

[Ru75] L.P. Rubinfeld, "A Proof of the Modified Booth's Algorithm for Multiplication," *IEEE Trans. Computers*, vol. 24, no. 10, pp. 1014-1015 Oct. 1975.

[SCWH07] J. Stine, I. Castellanos, M. Wood, J. Henson, F. Love, W.R. Davis, P.D. Franzon, M. Bucher, S. Basavarajaiah, J. Oh, R. Jenkal, FreePDK: An Open-Source Variation-Aware Design Kit. Proceedings of the 2007 IEEE International Conference on Microelectronic Systems Education, 2007.

[SNC01] A. Strollo, E. Napoli, and D. Caro, "New Design of Squarer Circuits Using Booth Encoding and Folding Techniques", Proceedings of the 8th ICECS, pp. 193-196, Sep. 02, 2001.

[WSMB99] K. Wires, W. Schulte, L. Marquette, and P. Balzola, "Combined Unsigned and Two's Complement Squarers", Proceedings of the Thirty Third Asilomar Conference on Signals, Systems, and Computers, pp. 1215-1219., 1999

[YW01] F. Yul and A. Willson, Jr., "Multirate Digital Squarer Architectures", Proceedings of the 8th ICECS, pp. 177-180, Sep. 02, 2001.