Hardware Simplification to the Delta Path in a MASH 111 Delta–Sigma Modulator

Chia-Yu Yao, *Member, IEEE*, and Chih-Chun Hsieh

***Abstract*—A MASH 111 delta–sigma modulator (DSM) is widely used in a fractional-N frequency synthesizer. This brief presents a low-complexity delta-path design in a MASH 111 modulator by simply recoding all carry output signals from accumulators. Compared with the prior approach, the hardware complexity of the proposed delta path is reduced to 53.4% of the prior approach. For frequency synthesizers with limited channels or for the pipelined DSM with accumulators possessing long word lengths, this saving is significant.**

***Index Terms*—Delta–sigma modulator (DSM), fractional-N frequency synthesizer, MASH 111.**

I. Introduction

A MASH 111 delta–sigma modulator (DSM) is widely used in a fractional-N frequency synthesizer [1]–[5] because it is easy to integrate in CMOS and is unconditionally stable [6]. Fig. 1 shows the block diagram of a MASH 111 DSM. We call the upper part in the dashed box as the "delta path" for easy reference. The 1-bit carry output signals from three accumulators, which serve as 1-bit quantized data, are sent to the delta path to calculate the desired 3-bit "pseudorandomized" sequence of the DSM.

In a conventional MASH 111 DSM, three accumulators have exactly the same word length (denoted by N). The output sequence length of a MASH 111 DSM is 2^{N+1} [7].

On the other hand, Ye and Kennedy [8] discovered that if we increase the word length of the first accumulator to $N+1$ and use much shorter word lengths for the second and third accumulators, we can still maintain the output sequence length as 2^{N+1}. Since the word lengths of the second and third accumulators are much shorter than N, the hardware complexity of a MASH 111 DSM is reduced [8].

In this brief, we focus on the delta path of a MASH 111 DSM. In [9], an approach for designing the delta path is given. The hardware complexity of the proposed delta path presented in this brief is about half the design given in [9]. Our idea is given as follows: If we treat the carry output signals o_i, $i = 1, 2, 3$, from the accumulators as -1, rather than 1 as usual, we can use two's complement arithmetic with the shortest bit width in the delta path. Thus, we can simplify the hardware required in the delta path. In this brief, we also investigate the advantages of using the proposed delta path in MASH 111 DSMs. We note that our treatment is independent on the word length that we use in each accumulator. Therefore, the design presented in this brief can be used in any MASH 111 DSM, including the one shown in [8].

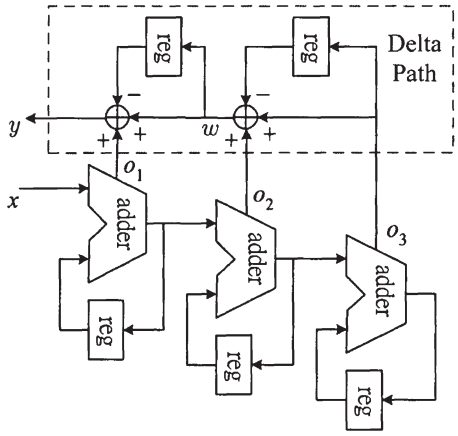

Fig. 1. Block diagram of a MASH 111 modulator.

This brief is organized in the following manner. In Section II, we introduce the hardware design of the proposed method and compare the complexity of the proposed design with that given in [9]. In Section III, a design of the encoder for a pulse-swallow counter connected to the DSM is described. In Section IV, synthesis results using Synopsys's Design Compiler for MASH 111 DSMs with the proposed delta path are shown and are compared with the DSMs employing the conventional delta path. Finally, conclusions of our work are summarized in Section V.

II. Proposed Delta-Path Hardware Design

In Fig. 1, o_i is conventionally 1 or 0, depending on whether overflow occurs in the corresponding accumulator. Therefore, the value set of w is $\{-1, 0, 1, 2\}$, and the value set of y is $\{-3, -2, -1, 0, 1, 2, 3, 4\}$. For two's complement arithmetic, the adder–subtractor units producing w and y should be 3 and 4 bits wide, respectively.

Our basic idea is as follows: If we treat 1 as -1 for o_i when overflow occurs, then the value set of w becomes $\{-2, -1, 0, 1\}$, and the value set of y becomes $\{-4, -3, -2, -1, 0, 1, 2, 3\}$. Thus, the adder–subtractor unit required to produce w becomes 2 bits wide, and the adder–subtractor unit required to produce y becomes 3 bits wide. In addition to the

Manuscript received August 23, 2008; revised November 18, 2008. First published March 10, 2009; current version published April 17, 2009. This work was supported by the National Science Council of Taiwan under Grant NSC 97-2221-E-011-139. This paper was recommended by Associate Editor P. K. Hanumolu.

The authors are with the Department of Electrical Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan (e-mail: chyao@mail.ntust.edu.tw).

Digital Object Identifier 10.1109/TCSII.2009.2015387

TABLE I
TRUTH TABLE OF THE ADDER PRODUCING THE SIGNAL w

o_2	o_3	a	w_1	w_0
0	0	0	0	0
0	0	1	0	1
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	0
1	1	0	1	0
1	1	1	1	1

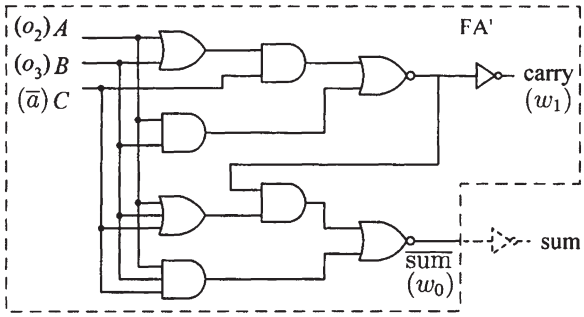

Fig. 2. Full-adder gate-level schematic.

word-length reduction, the corresponding hardware complexity can be reduced even more. This will be explained later.

It is noticed that when we treat 1 as -1 for o_i, the only difference in y is its sign. That is, the output signal y of the proposed delta path is exactly opposite to the original y of the conventional delta path.

Let us observe two adder–subtractor units in the delta path. Conventionally, they are realized as an adder–subtractor tree. First, consider the adder–subtractor unit that produces w, i.e.,

$$w = o_2 + o_3 + o_3 z^{-1} \tag{1}$$

where each input takes only 1 bit. Denoting $o_3 z^{-1}$ as a, we can construct the truth table according to (1). The truth table is shown in Table I. In Table I, we can obtain the Boolean equations for 2-bit outputs as

$$w_1 = o_2 o_3 + (o_2 + o_3)\overline{a} \tag{2}$$

$$\overline{w}_0 = o_2 \oplus o_3 \oplus \overline{a}. \tag{3}$$

The preceding Boolean equations are exactly the same as those of a full adder. However, the sum output is $\overline{w_0}$, rather than the desired w_0. In fact, this would not cost us any extra hardware. Recall that in a conventional full adder (see Fig. 2), the sum signal is generated by inverting an internal $\overline{\text{sum}}$ signal. Therefore, w_0 is already available, and we can remove one inverter, i.e., we even have a simpler full adder than the conventional full adder.

On the other hand, the adder–subtractor unit producing y can be constructed in the following manner: This unit is decomposed into a consecutive subtraction and addition. First, denote $b = wz^{-1}$ and $p = o_1 - b$. Then, $y = p + w$. The value set of p is $\{-2, -1, 0, 1, 2\}$. Thus, it requires 3 bits to represent p. Since two's complement arithmetic is used, sign extension must be carried out. The Boolean equations corresponding to calculating p can be simplified as

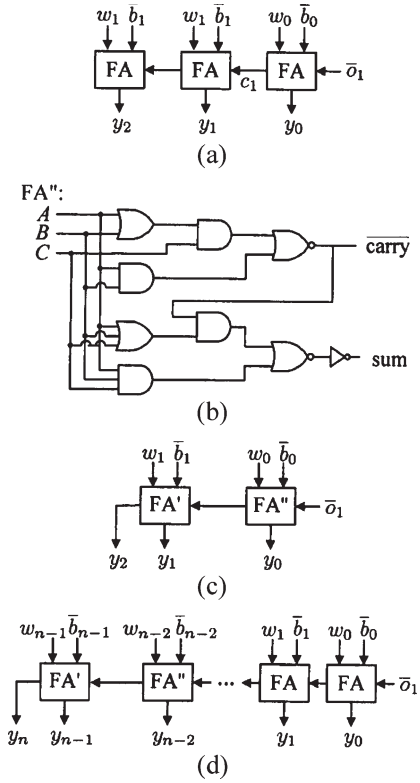

Fig. 3. Three-bit ripple carry adder for computing y.

$$p_0 = \overline{b}_0 \oplus \overline{o}_1 \tag{4}$$

$$p_1 = \overline{b}_1 \oplus \overline{o}_1 \overline{b}_0 \tag{5}$$

$$p_2 = \overline{b}_1 \oplus \overline{o}_1 \overline{b}_0 \overline{b}_1. \tag{6}$$

Hence, the circuit for calculating p is just a half-adder (HA) chain. Therefore, the hardware for calculating y can be as simple as a 3-bit ripple carry adder, as shown in Fig. 3(a). The Boolean expression of y_2 and y_1 can further be simplified as follows:

$$\begin{aligned} y_2 &= \overline{b}_1 \oplus w_1 \oplus \left[(w_1 + c_1)\overline{b}_1 + w_1 c_1\right] \\ &= \left[\overline{c}_1(w_1 + \overline{b}_1) + w_1 \overline{b}_1\right] \end{aligned} \tag{7}$$

$$\begin{aligned} y_1 &= \overline{b}_1 \oplus w_1 \oplus c_1 \\ &= \overline{\overline{b}_1 \oplus w_1 \oplus \overline{c}_1} \end{aligned} \tag{8}$$

where the signal c_1 is the carry output from the least significant bit (LSB) full adder.

Similar to the situation of calculating w, we can use FA$'$ in Fig. 2 to obtain y_2 and y_1. On the other hand, since the carry signal c_1 in (7) and (8) is inverted, it is equivalent to removing the output inverter of the carry path from the LSB full adder, as shown in Fig. 3(b), and is denoted by FA$''$. Therefore, the hardware for calculating y is shown in Fig. 3(c). The same technique can be extended to high-order MASH DSMs. The nth adder–subtractor unit in the delta path can be realized by just a special ripple carry adder shown in Fig. 3(d). For a

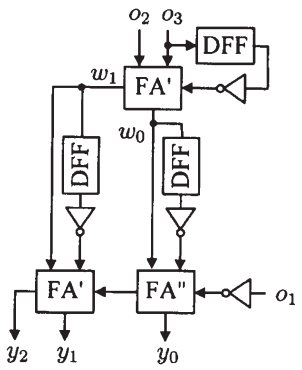

Fig. 4. Gate-level schematic of the proposed delta path in a MASH 111 DSM.

conventional delta path, more logic gates are required to realize an adder–subtractor unit. Moreover, in a conventional delta path, the bit width of each adder–subtractor unit is 1 bit longer than that of the proposed delta path. We note that in a high-order MASH DSM, the structure composed of FAs, FA′s, and FA″s in the delta path is highly regular that it can be reconnected in a carry-save form to reduce the length of the critical path.

The gate-level schematic of the proposed delta path for a MASH 111 DSM is shown in Fig. 4. To verify the correctness of the proposed delta path, we write a Verilog code corresponding to the block diagram shown in Fig. 5(a). Since the proposed delta path and the conventional delta path produce opposite outputs, if the check signal in Fig. 5(a) is zero, the correctness of the proposed delta path is guaranteed. By using Mentor Graphic's ModelSim, we simulate the system corresponding to Fig. 5(a), and we obtain an exact zero of the check signal. Fig. 5(b) shows a small segment of the simulation results. The periodogram of the out signal is shown in Fig. 5(c). We observe that it follows the 60-dB/dec line well.

Next, we discuss the hardware complexity of the proposed delta path. The circuit in Fig. 4 contains three D flip-flops, three full adders in degenerated forms, and four inverters. Compared with the delta-path implementation given in [9], whose gate-level schematic is given in [9, Figs. 7 and 8], the hardware complexity of the proposed circuit is significantly reduced. Without counting the redundant inverters in [9, Fig. 7] and the pipelined registers in [9, Fig. 8], Kenny *et al.* [9] employ four FAs, two inverters, six D flip-flops, and a 4-bit binary-subtract-3 circuit to realize the corresponding delta path. The 4-bit binary-subtract-3 circuit can be realized by one HA, two XNOR gates, one OR gate, and an inverter.

To approximately evaluate the hardware savings of the proposed circuit, we employ the following scoring system: A hardware cost of a logic gate is evaluated by half the number of transistors in a conventional CMOS topology. For example, an inverter costs 1 point. Therefore, a NOR gate costs 2 points, an OR gate costs 3 points, a D flip-flop costs 4 points, an XNOR gate costs 5 points, an HA costs 7 points, an FA′ or an FA″ gate costs 13 points, and an FA gate costs 14 points. With the aforementioned scoring system, the proposed delta path costs 55 points. On the contrary, the delta path in [9] costs 103 points. Therefore, the complexity of the proposed delta path is approximately 53.4% that of the delta path given in [9].

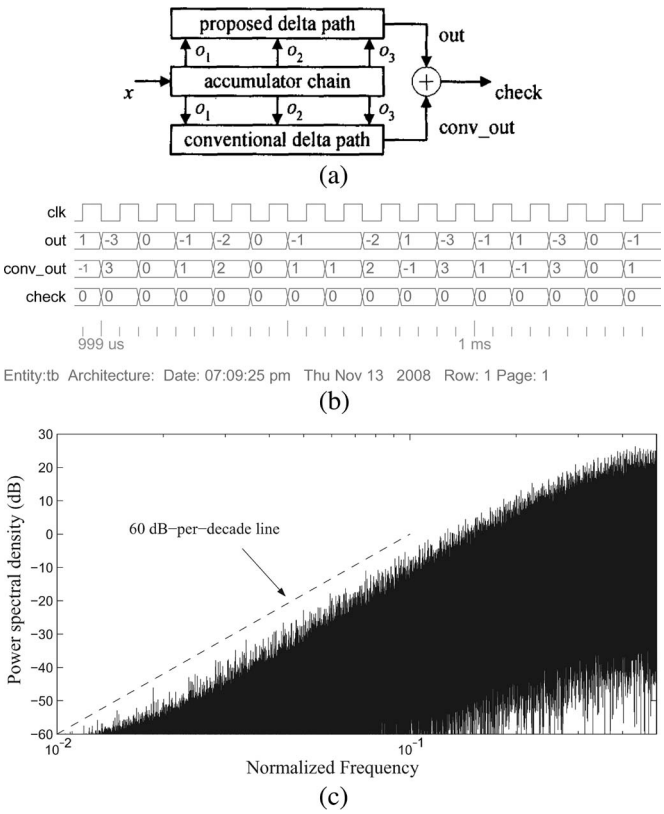

Fig. 5. Verification of the proposed delta path. (a) Block diagram for verification. (b) Simulation waveform by ModelSim. (c) Periodogram of the signal out in (a).

TABLE II
SYNTHESIS RESULTS OF THE PROPOSED DELTA PATH AND THE DELTA PATH IN [9] AT SEVERAL CLOCK FREQUENCIES

clock rate (MHz)	area and power consumption of			
	the proposed delta path		the delta path in [9]	
	area (μm^2)	power (μW)	area (μm^2)	power (μW)
25	2332.4	18.8	4511.0	24.3
50	2332.4	37.5	4511.0	48.6
100	2332.4	75.0	4741.1	97.6
125	2332.4	97.2	5117.2	179.6

Using Synopsys's Design Compiler and the TSMC-Artisan 0.18-μm cell library, we synthesize the proposed delta path and the delta path given in [9] (without pipelined registers). The results are shown in Table II. We observe that at a low clock rate, the area of the proposed delta path is about 51.7% that of the delta path given in [9], which is pretty close to our previous evaluation. We also observe that up to a 125-MHz clock rate, for the proposed delta path, the area remains the same, and power consumption is approximately linearly proportional to the clock rate. However, for the delta path given in [9], the area increases, and power consumption grows faster than the linear proportion. This explains why Kenny *et al.* [9] employ a pipelined structure in their delta path.

III. ENCODER AFTER A MASH 111 DSM

The outputs of a MASH 111 DSM are a sequence of two's complement numbers. They have to be converted into

TABLE III
DSM OUTPUT NUMBERS AND THE CORRESPONDING COUNTER CONTROL WORDS

2's complement numbers by the		binary counter control words in	
conventional delta path	proposed delta path	ascending order	descending order
0100 (4)	100 (−4)	000	111
0011 (3)	101 (−3)	001	110
0010 (2)	110 (−2)	010	101
0001 (1)	111 (−1)	011	100
0000 (0)	000 (0)	100	011
1111 (−1)	001 (1)	101	010
1110 (−2)	010 (2)	110	001
1101 (−3)	011 (3)	111	000

eight consecutive unsigned binary numbers before they can really control a programmable counter. A general solution for the conversion is to employ an adder for a conventional delta path or a subtractor for the proposed delta path.

On the other hand, if the DSM output is connected to a pulse-swallow counter and if the counter control words are binary $\{000, 001, \ldots, 111\}$, then we can use a very simple encoder in which no more than two inverters are required. Table III shows the DSM output numbers and two possible counter control word sets that are in opposite order. One can see that the binary control word is equal to $\overline{y}_2 y_1 y_0$ in ascending order and $y_2 \overline{y}_1 \overline{y}_0$ in descending order. That is, at most, two inverters are required in the encoder.

IV. SYNTHESIS RESULTS AND COMPARISON

Since the delta path is only a part of the DSM, we need to investigate the hardware and power savings for MASH 111 DSMs with the proposed delta path in real designs. Again, we use Synopsys's Design Compiler and the TSMC-Artisan 0.18-μm cell library to estimate the area and power consumption of several MASH 111 DSMs with different input word lengths and clock rates. For each DSM, we try to minimize power consumption during the synthesis process. Two examples are given in this section. The first example deals with DSM's accumulators with short to moderate word lengths, such that nonpipelined DSMs can work in sufficiently high clock rates. The second example deals with DSM's accumulators with long word lengths. In this case, a pipelined DSM architecture [4] must be employed to maintain the clock rate sufficiently high.

Example 1: Table IV shows the synthesis results for 4-, 6-, 8-, and 10-bit DSMs with several clock rates. We note that for the 4- and 6-bit DSMs with the conventional delta path, the clock rate can achieve 100 MHz. However, for the 8- and 10-bit DSMs with the conventional delta path, the clock rate can only achieve 93 and 91.24 MHz, respectively. For the DSMs with the proposed delta path, the maximum clock rate is a little higher than that of the corresponding conventional DSM. In Table IV, we note that as the clock rate increases, the absolute savings of both the area and power consumption for the DSMs with the proposed delta path become significant.

Since the delta path is only a part of the DSM, one may question the percentage savings that we can obtain for using the proposed delta path in the DSMs. There is no doubt that the percentage savings of either area and power are decreased as the word length of the accumulators used in the DSM increases. Consider the frequency synthesizer for wireless communications, in which the number of channels is limited (cf. [1], [3], and [10]). In such cases, we can still observe some percentage savings of the DSM area and power consumption (see Fig. 6). Both the percentage savings of area and power at a high clock rate are greater than those at a low clock rate. For a 4-bit DSM, the percentage savings of area can achieve 23.5% at a high clock rate. For a 10-bit DSM, the area savings are all more than 10.5%. At a high clock rate, the saving can achieve 14.4%. The percentage savings of power consumption range from 2.1% to 14.8%, depending on the bit width and operating clock rate of the DSM.

TABLE IV
SYNTHESIS RESULTS OF MASH 111 DSMS WITH THE PROPOSED DELTA PATH AND THE CONVENTIONAL DELTA PATH AT SEVERAL CLOCK FREQUENCIES

clock rate (MHz)	4-bit MASH 111 DSM with			
	proposed delta path		conventional delta path	
	area (μm^2)	power (μW)	area (μm^2)	power (μW)
12.5	7808.8	34.7	9657.8	36.4
25	7808.8	69.4	9657.8	72.8
50	7808.8	138.8	9657.8	145.5
100	13710.0	480.9	17911.5	564.3
clock rate (MHz)	6-bit MASH 111 DSM with			
	proposed delta path		conventional delta path	
	area (μm^2)	power (μW)	area (μm^2)	power (μW)
12.5	10386.9	48.5	12236.0	50.1
25	10386.9	97.1	12236.0	100.3
50	10386.9	194.9	12426.3	202.1
100	22256.0	839.9	27703.3	952.9
clock rate (MHz)	8-bit MASH 111 DSM with			
	proposed delta path		conventional delta path	
	area (μm^2)	power (μW)	area (μm^2)	power (μW)
12.5	12965.0	62.5	14814.1	64.2
25	12965.0	125.1	14814.1	128.4
50	12931.8	255.0	14987.9	260.8
93	28709.9	1001.5	35713.2	1168.9
clock rate (MHz)	10-bit MASH 111 DSM with			
	proposed delta path		conventional delta path	
	area (μm^2)	power (μW)	area (μm^2)	power (μW)
12.5	15543.2	76.4	17392.2	78.0
25	15543.2	152.7	17392.2	155.9
50	15460	318.2	17305.8	326.4
91.24	36942.9	1264.3	44515.7	1450.5

Example 2: In this example, we use the same word length (25 bits) as [9] for the accumulators in the DSM. Since the word length is long, we use the pipelined DSM structure [4], as shown in Fig. 7. Table V summarizes the synthesis results by Design Compiler. We observe that power consumption of DSMs with the proposed delta path is about the same as those with the conventional delta path. However, they show a significant difference in the area when the clock rate is high. Fig. 8 shows the area saving percentage of the DSMs with the proposed delta path over the DSMs with the conventional delta path. At a 100-MHz clock rate, the area saving is greater than 15%.

Therefore, not only the proposed delta path can save area in the small-to-moderate word-length DSMs, but it is also helpful

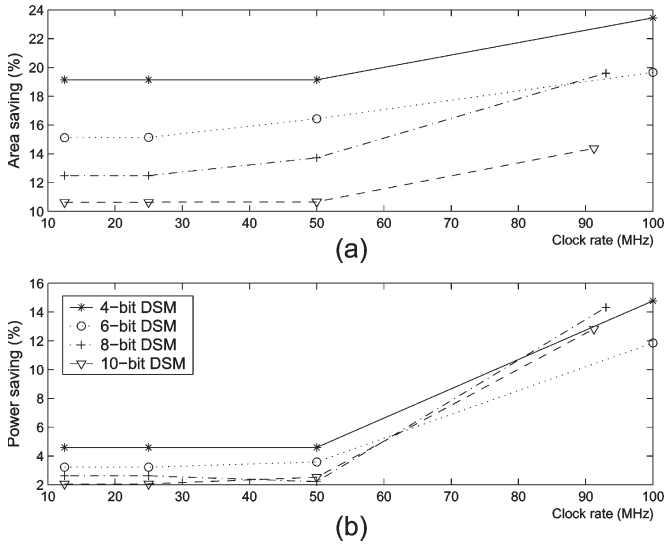

Fig. 6. Area and power saving percentages of MASH 111 DSMs with the proposed delta path over those with the conventional delta path. (a) Area saving percentage. (b) Power saving percentage.

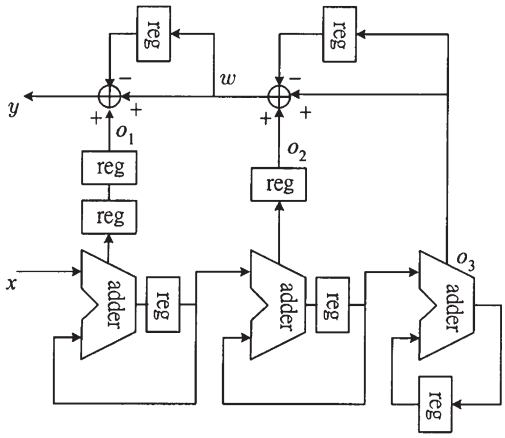

Fig. 7. Pipelined DSM structure for long-word-length accumulators.

TABLE V
SYNTHESIS RESULTS OF THE 25-BIT PIPELINED MASH 111 DSM WITH THE PROPOSED DELTA PATH AND THE CONVENTIONAL DELTA PATH

clock rate (MHz)	25-bit pipelined MASH 111 DSM with			
	proposed delta path		conventional delta path	
	area (μm^2)	power (μW)	area (μm^2)	power (μW)
25	35612.0	274.4	36181.8	278.6
50	37482.1	603.6	38468.6	618.9
75	62482.9	1023.4	67845.7	1051.3
100	70742.8	1551.7	84225.9	1589.7

to reduce the area for the pipelined DSM with long word lengths, particularly at a high clock rate.

V. CONCLUSION

In this brief, we have proposed a simplified delta-path design in a MASH 111 DSM by recoding the 1-bit carry output signal from the accumulator as -1's. The gate-level schematic shows a significant reduction of the hardware complexity compared with that of a prior approach.

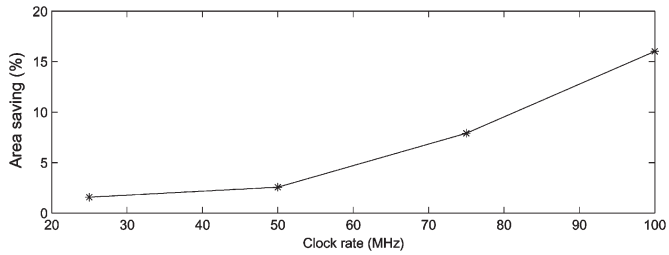

Fig. 8. Area saving percentage of the pipelined MASH 111 DSMs with the proposed delta path over those with the conventional delta path. The word length of the accumulator is 25 bits.

If a pulse-swallow counter is used after the DSM and if the counter control words are binary $\{000, 001, \ldots, 111\}$, then the encoder for converting output data from the proposed delta path requires no more than two inverters.

For frequency synthesizers used in wireless communications, in which the number of channels is limited, the savings of the DSM area and power using the proposed delta path are significant, particularly at a high clock frequency.

For the DSMs with long word lengths, a pipelined structure is employed to maintain a sufficiently high clock rate. In this case, the proposed delta path is also helpful to reduce the area of the DSM at a high clock rate.

ACKNOWLEDGMENT

The authors would like to thank the Chip Implementation Center (CIC), Taiwan, for providing the TSMC-Artisan 0.18-μm cell library in this project.

REFERENCES

[1] R. Ahola and K. Halonen, "A 2 GHz $\Delta\Sigma$ fractional-N frequency synthesizer in 0.35 μm CMOS," in *Proc. 26th Eur. Solid-State Circuits Conf.*, Sep. 2000, pp. 472–475.

[2] W. Rhee, B.-S. Song, and A. Ali, "A 1.1-GHz CMOS fractional-N frequency synthesizer with a 3-b third-order $\Delta\Sigma$ modulator," *IEEE J. Solid-State Circuits*, vol. 35, no. 10, pp. 1453–1460, Oct. 2000.

[3] R. Ahola and K. Halonen, "A 4 GHz fractional-N frequency synthesizer," in *Proc. Int. Conf. Electron., Circuits, Syst.*, Dec. 2000, vol. 1, pp. 239–242.

[4] S.-M. Wu, R.-Y. Liu, and W.-L. Chen, "A 5.8-GHz delta–sigma fractional-N frequency synthesizer for IEEE 802.11a applications," in *Proc. 5th Int. Conf. ASIC*, Oct. 2003, vol. 2, pp. 1074–1077.

[5] W.-T. Chen, J.-C. Hsu, H.-W. Lune, and C.-C. Su, "A spread spectrum clock generator for SATA-II," in *Proc. Int. Symp. Circuits Syst.*, May 2005, vol. 3, pp. 2643–2646.

[6] B. D. Muer and M. S. J. Steyaert, "On the analysis of $\Delta\Sigma$ fractional-N frequency synthesizers for high-spectral purity," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 50, no. 11, pp. 784–793, Nov. 2003.

[7] K. Hosseini and M. P. Kennedy, "Mathematical analysis of digital MASH delta–sigma modulators for fractional-N frequency synthesizer," in *Proc. Ph.D. Res. Micro-Electron. Electron.*, Jun. 2006, pp. 309–312.

[8] Z. Ye and M. P. Kennedy, "Reduced complexity MASH delta–sigma modulator," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 8, pp. 725–729, Aug. 2007.

[9] T. P. Kenny, T. A. D. Riley, N. M. Filiol, and M. A. Copeland, "Design and realization of a digital $\Delta\Sigma$ modulator for fractional-N frequency synthesis," *IEEE Trans. Veh. Technol.*, vol. 48, no. 2, pp. 510–521, Mar. 1999.

[10] H. R. Rategh, H. Samavati, and T. H. Lee, "A CMOS frequency synthesizer with an injection-locked frequency divider for a 5-GHz wireless LAN receiver," *IEEE J. Solid-State Circuits*, vol. 35, no. 5, pp. 780–787, May 2000.