

固定位宽乘法器的量化误差补偿方法及电路实现

马晓龙, 陈贵灿

(西安交通大学电子与信息工程学院, 710049, 西安)

摘要: 为了减小乘法器量化噪声对认知无线电信道检测性能的影响并节省芯片面积, 提出一种高精度的固定位宽基-4 Booth(FBB-4B)乘法器结构. 该乘法器的截断部分被分为保留、自适应补偿和常数补偿 3 部分. 常数补偿部分的量化误差补偿值合并到自适应补偿部分, 根据自适应补偿部分进位状态的编码产生自适应量化误差补偿值, 并设计了补偿进位生成电路. 相较于截断部分全部采用自适应补偿的乘法器, FBB-4B 乘法器的自适应补偿部分所包含的部分积位数较少, 使得自适应补偿部分的量化误差减小, 从而提高了该乘法器的精度. 仿真实验表明, FBB-4B 乘法器的精度比其他同类乘法器的精度提高了约 13%, 比理想基-4 Booth 乘法器的面积减少了 30% 左右.

关键词: Booth 乘法器; 固定位宽乘法器; 截断误差; 量化误差补偿

中图分类号: TN495 **文献标志码:** A **文章编号:** 0253-987X(2011)12-0075-07

A Compensation Method of Quantizing Error and Its Circuits Implementation for Fixed-Width Multiplier

MA Xiaolong, CHEN Guican

(School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: A high accuracy fixed-width base-4 Booth(FBB-4B) multiplier is proposed to reduce the effect stemmed from the quantizing noise of multipliers on the channels detecting of Cognitive Radios and to save the area of a chip. The truncated partial-product in the FBB-4B multiplier is divided into three sections: reserved section, adaptive compensation section and constant compensation section. The compensation value of quantizing error (VQER) derived from the constant compensation section is incorporated into the adaptive compensation section, and the VQER of the adaptive compensation section is generated by the encoding of carry states in adaptive compensation section. The circuits for generating the compensation carry bits are then designed. Compared with the multipliers which all truncated partial-product bits are used for adaptive compensation, the adaptive partial-product bits in the FBB-4B multiplier are reduced, so that the error of the adaptive compensation is decreased, and the accuracy of the fixed-width multiplier is improved. Simulation results show that the quantizing accuracy of the FBB-4B multiplier is increased about 13% compared with the existing methods, and it saves up to 30% area compared with the ideal base-4 Booth multipliers.

Keywords: Booth multiplier; fixed-width multiplier; truncated error; quantization error compensation

固定位宽乘法器因其较小的面积和功耗, 在多媒体信号处理和数字信号处理(DSP)的专用集成电

路(ASIC)设计中得到广泛应用^[1-4]. 在认知无线电信道检测专用芯片中, 为了减小芯片面积, 其中的乘

法器采用固定位宽基-4 Booth 乘法器.然而,固定位宽乘法器的量化噪声会影响信道检测的准确性,为此本文提出一种新的截断误差补偿方法以减小固定位宽基-4 Booth 乘法器的量化噪声.

通常有 3 种减小乘法器截断误差的方法:一种是常数补偿方法^[5],它是将截断部分的进位用一个常数来代替,这种方法误差较大;另一种方法是自适应误差补偿方法^[6],它根据截断的部分积产生可变的进位补偿而减小了截断误差;第三种是文献[7]提出的结合常数补偿和自适应误差补偿的混合补偿方法,与文献[5-6]相比,这种补偿方法所获得平均误差和最大绝对误差更小.文献[8-9]分别提出了不同的自适应误差补偿方法并应用于 Baugh-Wooley 乘法器的设计,但这些方法皆不能直接应用于 Booth 编码的乘法器.文献[10]提出了一种应用二进制阈值算法的补偿方法,虽然其误差均值得以减小,但其均方误差较之文献[1]的结果更大.文献[2]首先研究了应用统计线性回归分析产生量化误差补偿的方法,并将其扩展到基 4-Booth 乘法器的设计中,其量化误差较之文献[1]有明显下降.文献[11]则重点改进了文献[2]的不足,提出将截断部分分为主截断部分和次截断部分,并将与高位相邻的主截断部分予以保留,而次截断部分的截断误差则通过自适应补偿方法实现,从而获得了良好的性能.文献[12]在文献[11]的基础上提出一种新的进位补偿方法与结构,虽然其量化精度与[11]相当,但其量化误差分布则更加对称.

文献[11-12]中补偿进位的状态由全部次截断的部分积统计产生,量化误差较大.本文所提出的固定位宽基-4 Booth(FBB-4B)乘法器的次截断部分采用自适应误差补偿与常数补偿相结合的方法,提高了量化误差精度,并且这种方法可以极大地减小其他文献中的自适应进位统计产生时所需的耗时的计算量.仿真实验表明,与其他文献中的类似结构乘法器相比,在硬件代价相当的条件下,本文所提出的乘法器的精度可以提高约 13%.

1 改进的基-4 Booth 乘法器简介

Booth 编码方法能使乘法器的部分积数量明显减少,因而在乘法器硬件设计中得到了广泛应用^[10-12].常用的 Booth 编码方法是文献[13]所提出的改进的基-4 Booth 编码方法,表 1 列出了其编码逻辑.

表 1 改进的基-4 Booth 编码

$y_{2i+1}y_{2i}y_{2i-1}$	W_i	A_i	B_i	M_i	n_i
000	0	0	0	0	0
001	$\times 1$	0	1	0	1
010	$\times 1$	0	1	0	1
011	$\times 2$	1	0	0	1
100	$\times (-2)$	1	0	1	1
101	$\times (-1)$	0	1	1	1
110	$\times (-1)$	0	1	1	1
111	0	0	0	1	0

注: $y_{2i+1}y_{2i}y_{2i-1}$ 表示乘数的连续 3 位数; W_i 为编码权值; A_i 和 B_i 分别为 $|W_i|$ 的二进制表示的高位和低位; M_i 为 W_i 的符号状态; n_i 为 W_i 的非 0 状态.

16×16 位改进的基-4 Booth 乘法器的部分积阵列如图 1 所示,本文以此为例讨论固定位宽乘法器的量化误差补偿方法.

2 高精度进位补偿方法及实现

图 1 中乘法器的乘积可以表示为

$$\begin{aligned} P &= \text{sum}(P_W) + \text{sum}(P_T) = \\ &\text{sum}(P_W) + \text{sum}(P_M) + \text{sum}(P_A) + \text{sum}(P_C) \end{aligned} \tag{1}$$

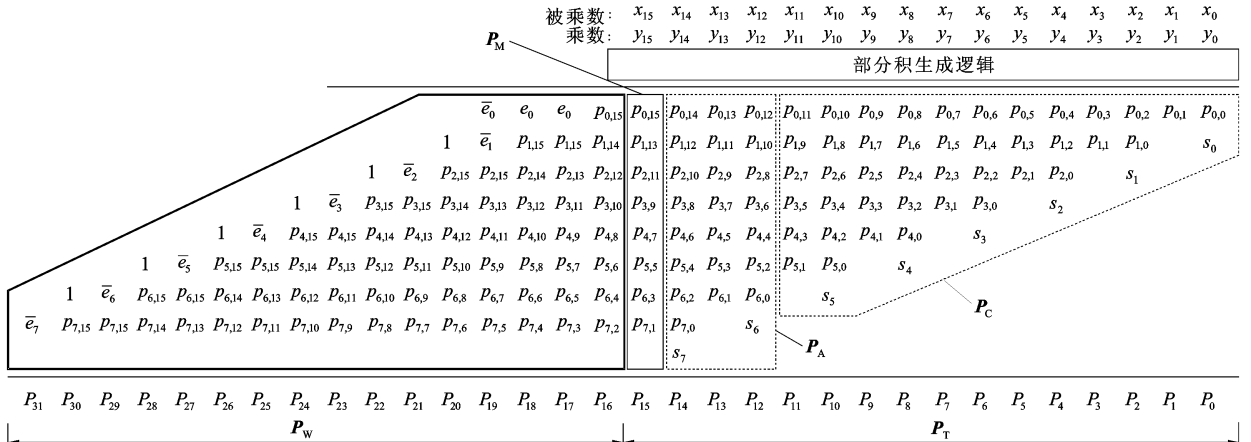
式中: $\text{sum}(\cdot)$ 表示求和; P_W 为部分积阵列的固定位宽积; P_T 为部分积阵列的截断积; P_M 为主截断积; P_A 为自适应补偿积; P_C 为常数补偿积.若假定小数点位于 P_W 与 P_T 之间,这对结果分析并无影响,则有

$$\begin{aligned} \text{sum}(P_M) &= 2^{-1}(p_{0,15} + p_{1,13} + \\ &p_{2,11} + p_{3,9} + p_{4,7} + p_{5,5} + p_{6,3} + p_{7,1}) \tag{2} \\ \text{sum}(P_A) &= 2^{-2}(p_{0,14} + p_{1,12} + \\ &p_{2,10} + p_{3,8} + p_{4,6} + p_{5,4} + p_{6,2} + p_{7,0} + s_7) + \\ &2^{-3}(p_{0,13} + p_{1,11} + p_{2,9} + p_{3,7} + p_{4,5} + p_{5,3} + p_{6,1}) + \\ &2^{-4}(p_{0,12} + p_{1,10} + p_{2,8} + p_{3,6} + p_{4,4} + p_{5,2} + p_{6,0} + s_6) \end{aligned} \tag{3}$$

$$\begin{aligned} \text{sum}(P_C) &= \\ &2^{-5}(p_{0,11} + p_{1,9} + p_{2,7} + p_{3,5} + p_{4,3} + p_{5,1}) + \\ &2^{-6}(p_{0,10} + p_{1,8} + p_{2,6} + p_{3,4} + p_{4,2} + p_{5,0} + \\ &s_5) + \cdots + 2^{-16}(p_{0,0} + s_0) \end{aligned} \tag{4}$$

式中: $p_{i,j}$ 为乘数的第 i 个 Booth 编码与第 j 位被乘数所生成的部分积.

在 FBB-4B 乘法器中, P_M 保留下来用以提高量化误差精度^[11].因为 P_A 和 P_C 距离 P_W 较远,对乘积的



P_W 表示部分积阵列的固定位宽部分; P_T 表示部分积阵列的截断部分; $p_{i,j}$ 为乘数的第 i 个 Booth 编码与第 j 位被乘数所生成的部分积; P_M 为主截断部分; P_A 为自适应补偿部分; P_C 为常数补偿部分; e_i 为符号扩展位^[14]; s_i 为符号位, 且 $s_i = M_i$

图 1 16×16 位改进的基-4 Booth 补码乘法器部分积阵列

影响较小,故这两部分的进位可以通过统计的方法产生. 相对于 P_C , P_A 对乘积影响更大,所以这一部分的进位通过自适应补偿方法来生成. 可以根据到 P_M 的最大进位来决定 P_A 包含的列数,一般为 3~5 列. P_C 对乘积的影响相对更小,这一部分的进位则通过常数补偿方法来实现.

表 1 中 n_i 表示编码权值 W_i 的非 0 状态,它与 $|W_i|$ 二进制表示的高位 A_i 和低位 B_i 的关系如下

$$n_i = A_i \vee B_i \tag{5}$$

式中: $\vee$ 表示逻辑“或”,即如果 Booth 编码不为 0, 则 $n_i=1$,而当 $n_i=1$ 时, $p_{i,j}$ 为 0 或 1 的概率相等. 假定 $p_{i,j}$ 和 s_i 为 0 或 1 的概率皆为 1/2,则有

$$E[p_{i,j}] = E[s_i] = \frac{1}{2} \tag{6}$$

式中: $E[\cdot]$ 表示求均值. 这样,将式(6)代入式(4), 可得 P_C 部分的均值为

$$E[\text{sum}(P_C)] = 2^{-5} \times \left(\frac{1}{2} \times 6\right) + 2^{-6} \times \left(\frac{1}{2} \times 7\right) + \dots + 2^{-16} \times \left(\frac{1}{2} \times 2\right) = \frac{3}{16} \tag{7}$$

那么,常数 3/16 将作为 P_C 的量化误差补偿值.

另外, s_7 的均值为 1/2,考虑到应用舍入处理, 将其用常数 1 来代替. 根据上面小数点位置的规定, s_7 的等效十进制值为 1/4. 综上, P_A 和 P_C 两部分等效为图 2 左侧图所示结构,右侧图为其等效部分积位的阵列形式,记为 P'_A .

图 2 中 $c_i (i=1,2,\dots,6)$ 表示 P'_A 中各行相加的进位状态,并通过仿真来统计各 $c_i=1$ 的次数与 n_i 之间的关系. 接下来这样考虑:如果乘数的 Booth 编码 $n_7 n_6 n_5 n_4 n_3 n_2 n_1 n_0$ 中只有一位是 1,例如

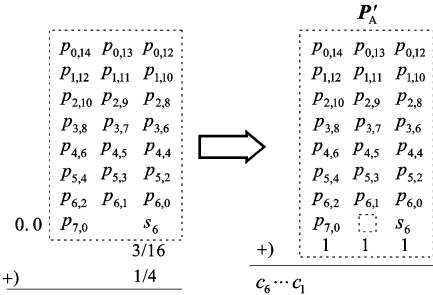


图 2 FBB-4B 乘法器次截断部分的等效部分积阵列

$n_7 n_6 n_5 n_4 n_3 n_2 n_1 n_0 = 00000001$ 等,那么 P'_A 中除最下面一行外只可能有一行非全 0,将其称为第 1 类. 根据 $n_7 n_6 n_5 n_4 n_3 n_2 n_1 n_0$ 中 1 的个数共可分为 9 类,其中第 0 类表示为 $n_7 n_6 n_5 n_4 n_3 n_2 n_1 n_0 = 00000000$. 因为 P'_A 有 3 列,所以第 1 类非全 0 的那一行共有 8 种不同状态,即 P'_A 有 8 种状态,然后对所有状态计算 P'_A 的和并统计各 $c_i = 1$ 的次数. 再如 $n_7 n_6 n_5 n_4 n_3 n_2 n_1 n_0 = 00000011$ 等为第 2 类,其 P'_A 中只有两行有可能非全 0, P'_A 则有 64 种不同状态,然后同样对所有状态计算求和并统计各 $c_i = 1$ 的次数. 因为部分积 $p_{i,j}$ 为 0 或 1 的概率相等,且各 $s_i = 1$ 的概率也相等,所以仿真时并不必关心 $n_7 n_6 n_5 n_4 n_3 n_2 n_1 n_0$ 中若干个 1 出现的位置,而只需对每类中的一种情况进行仿真,其结果就代表了同类的所有其他情况. 与文献 [11-12] 等相比,本文的仿真方法大大减少了统计仿真时的计算量. 每一类情况仿真的进位状态 $c_i = 1$ 的总数列于表 2 中. 注意,图 2 中 P'_A 的倒数第 2 行的中间一位为 0,但其对统计结果影响非常小,所以在仿真时将其视作一般部分积对待.

下面,先对上文所提到的每一类的状态进行再编码,即对 $n_7n_6n_5n_4n_3n_2n_1n_0$ 中 1 的个数进行计数,称其为类型编码. 对第 0 类,其编码 $z_3z_2z_1z_0=0000$,第 1 类编码为 $z_3z_2z_1z_0=0001$ 等. 对 c_i 作如

下规定:如果表 2 中某 $c_i=1$ 的总数大于此类仿真总次数的一半,则记 $c_i=1$,否则记 $c_i=0$. 这样,从表 2 得出 P'_A 的补偿进位 c_i 与 $z_3z_2z_1z_0$ 之间的关系如表 3 所示.

表 2 P'_A 进位状态仿真统计结果

类别	进位状态 $c_i=1$ 的总数							仿真总次数
	c_0	c_1	c_2	c_3	c_4	c_5	c_6	
第 0 类	0	0	0	0	0	0	0	0
第 1 类	7		0	0	0	0	0	8
第 2 类	63	21	0	0	0	0	0	8^2
第 3 类	511	350	35	0	0	0	0	8^3
第 4 类	4 095	3 605	1 225	35	0	0	0	8^4
第 5 类	32 767	31 486	18 844	2 898	21	0	0	8^5
第 6 类	262 143	259 147	205 534	71 022	4 963	7	0	8^6
第 7 类	2 097 151	2 090 724	1 897 019	1 048 576	200 133	6 428	1	8^7
第 8 类	16 777 215	16 764 354	16 144 677	11 782 380	4 146 393	438 834	6 435	8^8

表 3 类型编码与补偿进位间的关系

$z_3z_2z_1z_0$	c_0	c_1	c_2	c_3	c_4	c_5	c_6
0000	0	0	0	0	0	0	0
0001	1	0	0	0	0	0	0
0010	1	0	0	0	0	0	0
0011	1	1	0	0	0	0	0
0100	1	1	0	0	0	0	0
0101	1	1	1	0	0	0	0
0110	1	1	1	0	0	0	0
0111	1	1	1	1	0	0	0
1000	1	1	1	1	0	0	0

从表 3 看到,对 16×16 乘法器仅需 4 个进位状态寄存器 $c_0\sim c_3$ 就可以满足对 P'_A 进位状态寄存的要求.

图 3 是对 $n_7n_6n_5n_4n_3n_2n_1n_0$ 中 1 的个数进行计数产生类型编码 $z_3z_2z_1z_0$ 的组合逻辑电路,图中 2Bs-Adder 和 3Bs-Adder 分别表示 2 bit 和 3 bit 加法器.

根据表 3,补偿进位 $c_0\sim c_3$ 与编码 $z_3z_2z_1z_0$ 间的逻辑关系如下

$$\left. \begin{aligned} c_0 &= z_3z_2z_1z_0 + \overline{z_3z_2z_1z_0} \\ c_1 &= z_3z_2z_1z_0 + \overline{z_3z_2} + \overline{z_3}z_1z_0 \\ c_2 &= z_3z_2z_1z_0 + \overline{z_3}z_2(z_1 + z_0) \\ c_3 &= z_3z_2z_1z_0 + \overline{z_3}z_2z_1z_0 \end{aligned} \right\} \quad (8)$$

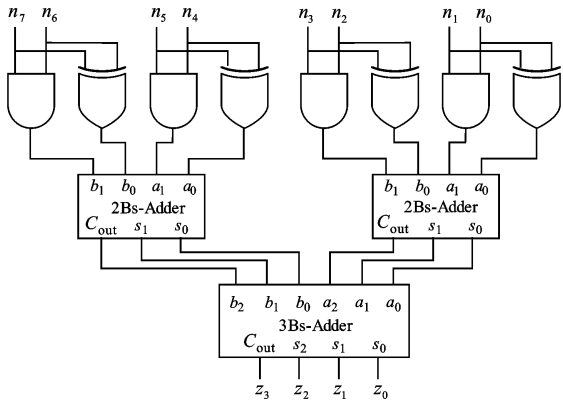


图 3 类型编码电路

综上,由主截断部分积阵列 P_M 与量化误差补偿电路构成的 FBB-4 Booth 乘法器结构如图 4 所示. 为了进行舍入处理,在图 4 中 $p_{7.1}$ 下面加入常数 1.

3 乘法器性能分析比较

3.1 乘法器性能仿真

在乘法器位宽分别为 8、10、12 和 16 bit 的情况下,每次仿真时随机产生 60 000 组数据进行相乘,计算固定位宽乘法器的输出乘积 $P_{fixed-width}$ 与理想乘积值 P_{ideal} 之间的误差 $\epsilon=P_{ideal}-P_{fixed-width}$. 对结果进行分析,获得量化误差的最大误差值 ϵ_{max} 、均值 ϵ_{mean} 和均方差 ϵ_{var} 并列于表 4. 从表 4 的结果可以看出,本文设计的 FBB-4B 乘法器的最大误差与其他文献相当,其均值 ϵ_{mean} 小于文献[10-11]的结果. 另外,我

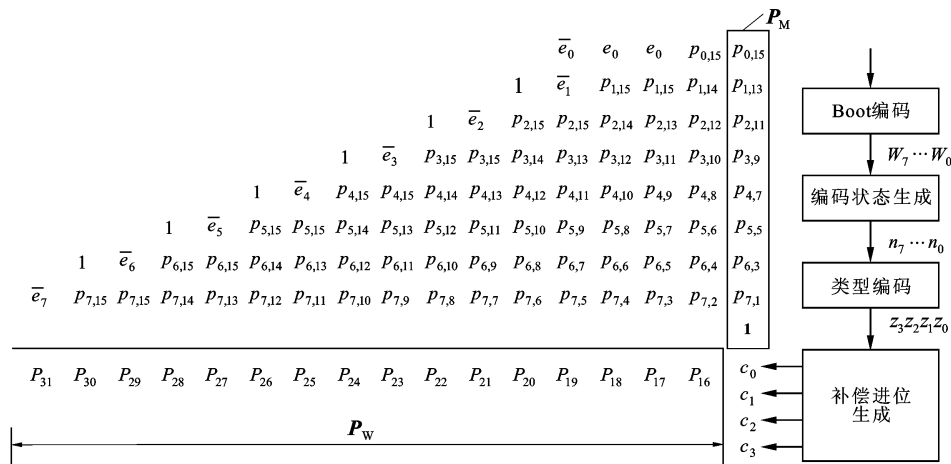


图 4 本文所提出的 16×16 FBB-4 Booth 乘法器结构

表 4 不同乘法器误差统计与比较

位宽/bit	乘法器类型	$\epsilon_{\max}$	ϵ_{mean}	ϵ_{var}	量化误差/%		
					$ \Delta <1$	$1\leq \Delta <2$	$2\leq \Delta <3$
8	理想舍入	0.500 0	0.000 0	0.083 3	100		
	文献[11]	1.500 0	0.132 8	0.166 4	77.2	22.797	0.003 1
	文献[10]	1.164 1	0.175 8	0.145 9	79.9	21.100	0.000 0
	文献[12]	1.168 0	0.007 8	0.136 7	81.5	18.500	0.000 0
	FBB-4B	1.430 8	0.071 7	0.166 9	90.3	9.698	0.001 6
10	文献[11]	1.500 0	0.121 1	0.171 3			
	文献[10]	1.365 2	0.202 8	0.171 3			
	文献[12]	1.500 0	−0.003 9	0.154 2			
	FBB-4B	1.522 5	0.068 0	0.190 5	89.91	10.08	0.008 4
12	文献[11]	2.000 0	0.127 0	0.195 0	73.33	26.65	0.018
	文献[10]	1.564 9	0.220 3	0.196 0	72.5	27.50	0.000
	文献[12]	1.666 7	0.002 0	0.167 1	76.83	23.17	0.000
	FBB-4B	2.030 0	0.065 5	0.198 4	89.70	10.287	0.013
16	文献[11]	2.500 0	0.125 5	0.223 5	70.10	29.83	0.07
	文献[10]	1.965 0	0.238 4	0.240 9	67.71	32.29	0.005
	文献[12]	2.166 7	0.000 5	0.196 1	73.13	26.85	0.02
	FBB-4B	2.640 3	0.060 4	0.216 9	87.60	12.37	0.03

们对量化误差的分布情况进行了统计,从表 4 第 6 ~8 列可以看出,对于不同位宽,FBB-4B 乘法器的归一化误差 $|\Delta|<1$ 的比例高于其他文献达 13% 以上,而较大误差 $2\leq|\Delta|<3$ 的比例相对较低. 文献 [12] 所提出的误差补偿方法,使截断误差的正、负分

布基本对称,因而其误差均值优于其他文献,但是其量化误差补偿结构与文献 [11] 的基本相同,从仿真结果看,其归一化误差 $|\Delta|<1$ 的比例并没有得到提高,所以其量化误差精度并没有得到明显改善. 结果分析表明,FBB-4B 乘法器的量化精度要高于表

中其他文献的量化精度.

应用 TSMC 0.18 μm CMOS 工艺标准单元库对位宽分别为 8、10、12 和 16 bit 的几种乘法器进行综合,结果列于表 5. 从表 5 可看出,乘法器位宽较小时,例如位宽为 8、10 bit,FBB-4B 乘法器的面积、功耗较大. 乘法器位宽较大时,由于 FBB-4B 乘法器的补偿电路产生的进位数要少于其他文献,使部分积累加部分的硬件开销增加不大,从而抵消了其相对其他文献稍大的补偿进位生成电路,因而其面积、功耗与其他文献相当. 通过对等效面积的计算表明,所提出的乘法器相对理想乘法器可节约面积 30%左右.

表 5 不同乘法器性能比较

位宽/bit	乘法器类型	等效面积/ μm^2	延时/ns	功耗/mW
8	理想	3 392	5.82	1.030
	文献[11]	2 238	5.81	0.639
	文献[10]	2 396	6.14	0.667
	文献[12]	2 251	5.85	0.628
	FBB-4B	2 356	6.31	0.659
10	理想	5 154	6.60	1.293
	文献[11]	3 479	6.26	0.789
	文献[10]	3 758	6.97	0.833
	文献[12]	3 395	6.62	0.777
	FBB-4B	3 612	6.94	0.798
12	理想	7 455	7.24	1.773
	文献[11]	5 144	6.67	1.064
	文献[10]	5 286	7.61	1.137
	文献[12]	5 161	7.43	1.099
	FBB-4B	5 239	7.66	1.049
16	理想	13 554	8.06	2.562
	文献[11]	9 692	7.84	1.486
	文献[10]	10 058	8.41	1.563
	文献[12]	9 508	8.08	1.476
	FBB-4B	9 641	8.72	1.461

FBB-4B 乘法器的类型编码电路中的加法器树采用了行波进位加法器,使得补偿进位生成电路延时较大,进而使整个乘法器关键路径延时增大,其值大于所列文献约 8%~18%.

3.2 乘法器应用实验仿真

下面通过一个 20 阶的低通 FIR 滤波器来验证本文设计的 FBB-4B 乘法器的性能. FIR 滤波器采

用直接结构,输入、输出及滤波器系数皆为 16 bit. 测试输入信号为图 5a 所示的幅值为 1 V、频率为 5 MHz 的正弦信号. 仿真时,激励文件包含 2 000 个输入信号样本. FIR 滤波器中的乘法器应用 FBB-4B 乘法器、文献[11-12]的乘法器分别进行例化,不同例化情况下滤波器的输出如图 5b 所示(图中示出 10 个输出样本). 从表 6 输出幅值量化误差的均值和方差结果比较可以看出:FBB-4B 乘法器样本误差均值小于文献[11],高于文献[12];样本误差方差分别低于文献[11]和文献[12]约 13.5%和 8.5%,其结果与 3.1 节对乘法器的统计仿真结果吻合.

图 5c 示出 3 种例化情况下 FIR 滤波器输出信号的谱分析结果. 从图 5c 的局部放大部分和列于表 6 最后一列的频谱最大幅度相对误差的计算结果可以看出:应用 FBB-4B 乘法器的 FIR 滤波器的输出信号频谱幅度最大值更接近理想值,说明 FBB-4B 乘法器的量化误差对频谱检测的影响最小.

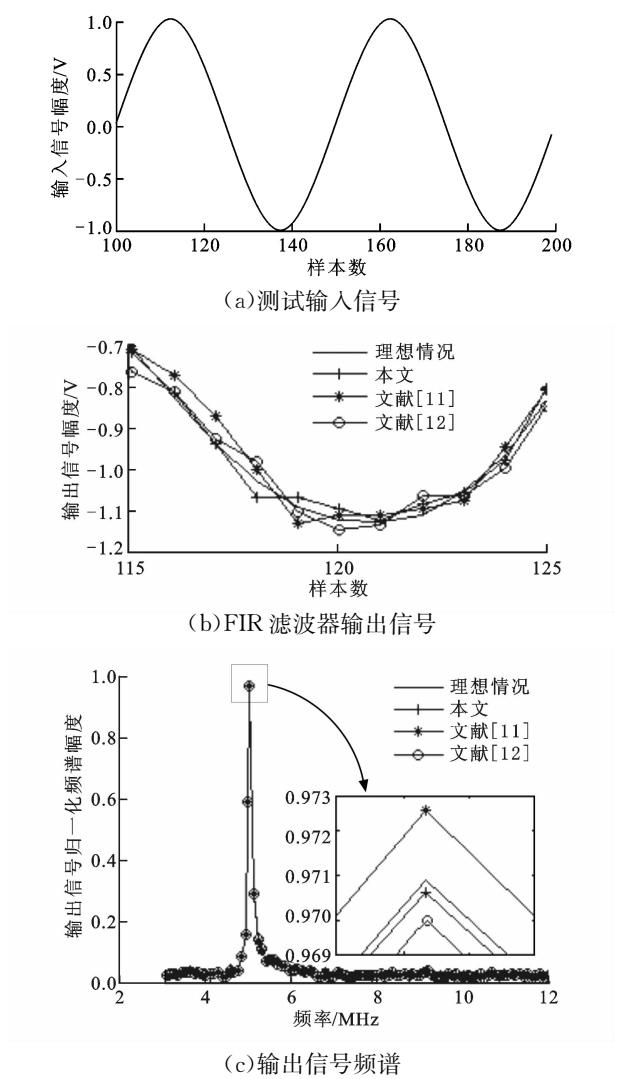


图 5 不同乘法器例化时的 FIR 滤波器性能比较

表 6 滤波器数值实验结果

乘法器 类型	输出幅度量化 误差均值/V	输出幅度量化 误差方差/V	频谱最大幅度 相对误差/%
文献[11]	0.135	0.327	0.015 5
文献[12]	0.015	0.315	0.008 2
FBB-4B	0.061	0.288	0.002 0

4 结 论

采用本文所提出的自适应量化误差补偿方法所设计的 FBB-4B 乘法器,可以大幅提高固定位宽乘法器的量化精度. 仿真实验表明,本文所提出的乘法器在面积、功耗等方面与其他文献中的类似乘法器相当,不足之处是其关键路径延时稍大. 最后,通过应用实例验证了 FBB-4B 乘法器的精度高于类似结构乘法器,可应用于有较多乘法器且数据精度要求较高的 ASIC 设计中.

参考文献:

[1] JOU J M, KUANG S R, CHEN R D. Design of low-error fixed-width multipliers for DSP applications [J]. IEEE Transactions on Circuits and Systems; II Analog and Digital Signal Processing, 1999, 46(6): 836-842.

[2] JON S J, WANG Hui-Hsuan. Fixed-width multiplier for DSP application [C]//Proceedings of International Conference on Computer Design. Los Alamitos, CA, USA; IEEE Computer Society, 2000: 318-322.

[3] 孙凌,杨明武. 定宽截断式并行乘法器的实现研究 [J]. 中国集成电路, 2007, 16(12): 67-70.

SUN Ling, YANG Mingwu. Design of low-error two's-complement fixed-width parallel multipliers [J]. China Integrated Circuit, 2007, 16(12): 67-70.

[4] CHEN Yuan-Ho, CHANG Tsin-Yuan, JOU Ruei-Yuan. A statistical error-compensated booth multipliers and its DCT applications [C] // Proceedings of IEEE Region 10 Annual International Conference. Piscataway, NJ, USA; IEEE, 2010: 1146-1149.

[5] SCHULTE M J, SWARTZLANDER E E. Truncated

multiplication with correction constant [C]//Proceedings of IEEE Workshop on VLSI Signal Processing. Piscataway, NJ, USA; IEEE, 1993: 388-396.

[6] KING E J, SWARTZLANDER E E. Data-dependent truncated scheme for parallel multiplication [C]//Proceedings of the Thirty-First Asilomar Conference on Signals, Systems & Computers. Piscataway, NJ, USA; IEEE, 1997: 1178-1182.

[7] STINE J E, DUVERNE O M. Variations on truncated multiplication [C] // Proceedings of the Euromicro Symposium on Digital System Design. Los Alamitos, USA; IEEE Computer Society, 2003: 112-119.

[8] VAN Lan-Da, YANG Chih-Chyau. Generalized low-error area-efficient fixed-width multipliers [J]. IEEE Transactions on Circuits and Systems; I Regular Papers, 2005, 52(8): 1608-1619.

[9] NICOLA P, DAVIDE D C, VALERIA G, et al. Truncated binary multipliers with variable correction and minimum mean square error [J]. IEEE Transactions on Circuits and Systems; I Regular Papers, 2010, 57(6): 1312-1325.

[10] SONG Min-An, VAN Lan-Da, KUO Sy-Yen. Adaptive low-error fixed-width booth multipliers [J]. IEICE Trans Fundamentals, 2007, E90-A(6): 1180-1187.

[11] CHO K J, LEE K C, CHUNG J G, et al. Design of low-error fixed-width modified booth multiplier [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2004, 12(5): 522-531.

[12] WANG Jiun-Ping, KUANG S R, LIANG S C. High-accuracy fixed-width modified booth multipliers for lossy applications [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2010, 19(1): 52-60.

[13] MACSORLEY O L. High-speed arithmetic in binary computers [J]. Proceedings of the IRE, 1961, 49(1): 67-91.

[14] BEWICK G W. Fast multiplication: algorithms and implementation [D]. Stanford, CA, USA; Stanford University, 1994.

(编辑 刘杨)