The New Architecture of Chinese Abacus Multiplier

[1]Chien-Hung Lin,[1]Yun-Fu Huang, [1]Der-Her Lee, [1]Pao-Hua Liao and [2]Chih-Wei Hsu
[1]Department of Electrical Engineering,
LEE-Ming Institute of Technology
[2]General Education Center
LEE-Ming Institute of Technology
No 2-2,Lijuan Rd.,Liming,Taishan,Taipei
Taiwan, R.O.C
lkhwin2004@yahoo.com.tw

Abstract—This study demonstrated a 4x4 bits multiplier that was based on the Chinese abacus. Comparing the simulation results of this work with the speed and power consumption of the 4x4 bits Braun array multiplier, this 4x4 bits abacus multiplier showed a 19.7% and 10.6% delay improvement in 0.35μm and 0.18μm technology respectively than that of the 4x4 bits Braun array multiplier, while power consumption of the 4x4 bits abacus multiplier was 8.7% and 18% lower respectively.The performance**:** power-consumption*delay of the abacus multiplier is respectively,less about 23.2% and 23.5% also.

Key Words—Performance,Tree-based multipliers, Braun array multiplier,Function table, Chinese abacus multiplier, Delays, Thermometric

1 Introduction

Multiplication is one of the most critical operations for many computational systems. Among the various multiplier techniques available, the most well-known ones are array-based multipliers [1][2] and tree-based multipliers [3]. They are commonly used in VLSI design for implementing fast multipliers [4]-[9]. The aim of this study is to devise a multiplier based on the ancient Chinese abacus algorithm, for a more efficient operation of high speed and low power consumption..

The Chinese abacus is an old invention that has been widely used as a tool for performing arithmetic functions in China and other Asian countries for centuries. The basic architecture of the Chinese abacus is depicted in Fig.1, which represents the number of one hundred and sixty-eight. Each column on the Chinese abacus consists of upper and lower beads, in which each upper bead denotes an amount of 5 and the lower one an amount of 1. One key feature of the Chinese abacus is the use of one upper bead in each column, which minimizes operations for users. The earliest multiplier and adder employing the technique of the Chinese abacus were proposed by Gang et al. [10]–[12].

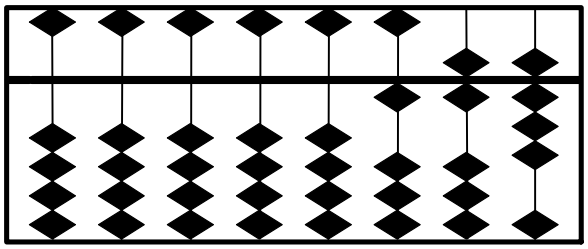


Fig.1 Basic architecture of Chinese abacus showing the number of 168.

In this article, a novel multiplier based on the Chinese abacus was put forward as shown in Fig.2. This multiplier is made up of three segments, each with three rows of beads. Each up bead represents sixteen, while each of the middle and bottom bead represents 4 and 1 respectively. Fig.2 depicts the number of $39(2*4^2+1*4^1+3*4^0)$ represented in the proposed Chinese abacus.

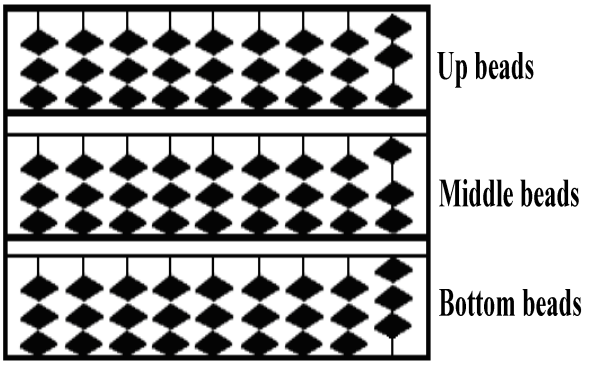


Fig.2 The configuration of proposed novel Chinese abacus multiplier

.

In the proposed 4x4 bits Chinese abacus multiplier, instead of making one straight forward multiplication as shown in Fig. 3, it was done in three steps:

(1) Binary product to abacus
(2) Parallel addition
(3) Thermometric to Binary

The proposed Chinese abacus multiplier is based on an abacus adder in [15] that each column element, e.g. $(H_2H_1H_0|M_2M_1M_0|L_2L_1L_0)_{abacus}$, consists of beads of three different weighting. Below shows how a number is represented:

$(H_2H_1H_0|\ M_2M_1M_0|L_2L_1L_0)_{abacus}$
$= (H_2 + H_1 + H_0)*16 + (M_2 + M_1 + M_0)*4 + (L_2 + L_1 + L_0)$

Fig.3 shows a multiplication operation where B = $(b_3b_2b_1b_0)_2 = (1101)_2 = 13$ is the multiplicand and A = $(a_3a_2a_1a_0)_2 = (1110)_2 = 14$ is the multiplier. This multiplication first begins with two partial products, i.e., $(a_1a_0)_2 * (1101)_2 = (001|011|011)_{abacus}$ and $(a_3a_2)_2 * (1101)_2 = (011|001|111)_{abacus}$,, followed by the binary products of these two partial products , which is denoted by a binary product to abacus (BPA), $(011|111|001|011)_{abacus}$,, as shown in Fig. 3. This binary product represents a number of $(011|111|001|011)_{abacus} = 2\times4^3 + 3\times4^2 + 1\times4^1 + 2\times4^0 = 182$.

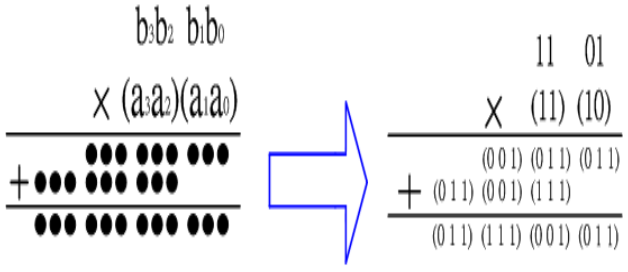


Fig.3 Example of the multiplication based on the proposed algorithm.

Figure 4 is the block diagram of the proposed multiplier. The 4x4 bits abacus multiplier is divided into three modules. The first one is the BPA (binary product to abacus) module. The second one is the PA (parallel addition) module [15]. The third one is the TB (Thermometric to Binary) [15]. These three modules are discussed in the following sections.

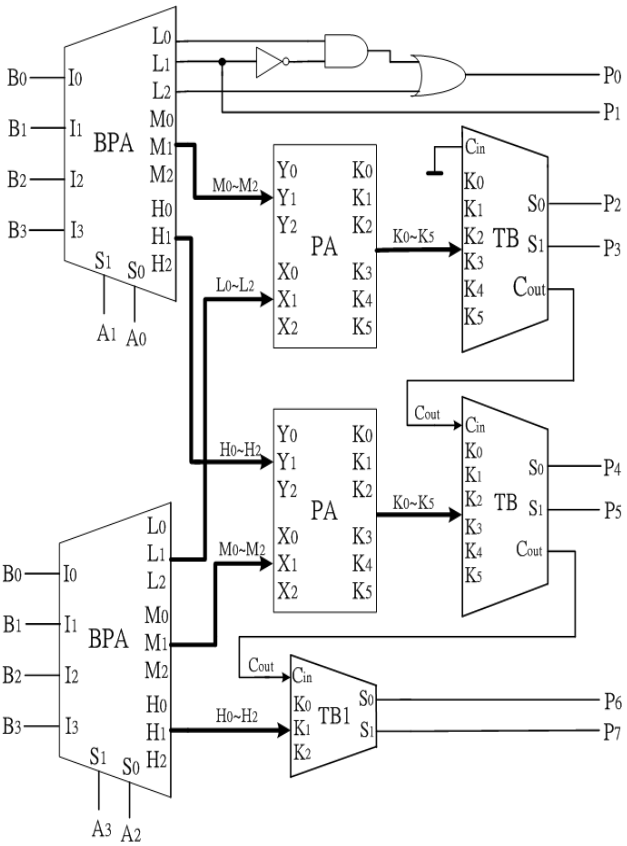


Fig.4 Block diagram of the 4x4 bits abacus multiplier.

2 The Design of the Proposed Multiplier

2.1 The BPA Module

Fig. 5 shows the block diagram of the BPA module. In this module, each 4x2 binary number, $(b_3b_2b_1b_0)_2*(a_1a_0)_2$ and $(b_3b_2b_1b_0)_2*(a_3a_2)_2$, were converted into abacus representation. $(H_2H_1H_0|M_2M_1M_0|L_2L_1L_0)_{abacus}$. $(H_2H_1H_0)$ represented three higher beads of sixteen (4^2). $(M_2M_1M_0)$ represented the three middle beads of four (4^1), and $(L_2L_1L_0)$ represented three lower beads of one (4^0).

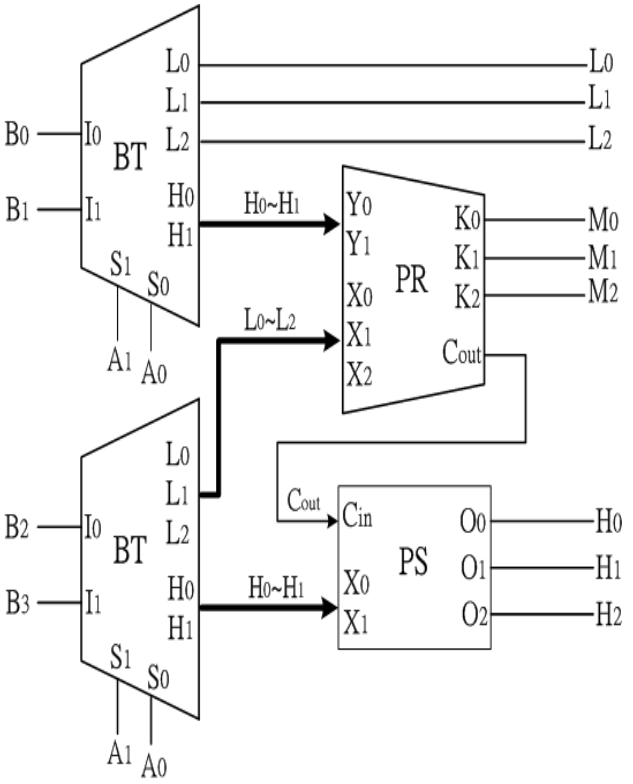


Fig.5 The block diagram of BPA module.

From Fig 5, one can also see that BPA module is made up of three different sub-modules, and equations (1) - (5) model he behavior of the BT module:

$$L_0 = (I_1 + I_0)(\overline{S_1} \cdot S_0) + (I_0)(S_1 \cdot \overline{S_0}) + (I_1 + I_0)(S_1 \cdot S_0) \quad (1)$$

$$L_1 = (I_1)(\overline{S_1} \cdot S_0) + (I_0)(S_1 \cdot \overline{S_0}) + (I_1 \oplus I_0)(S_1 \cdot S_0) \quad (2)$$

$$L_2 = (I_1 \cdot I_0)(\overline{S_1} \cdot S_0) + (\overline{I_1} \cdot I_0)(S_1 \cdot S_0) \quad (3)$$

$$H_0 = I_1 \cdot S_1 \quad (4)$$

$$H_1 = (I_1 \cdot I_0)(S_1 \cdot S_0) \quad (5)$$

The PR module adds the beads with the same weight and then transforms the beads into middle beads $(K_2K_1K_0)$. The behavior of the PR module can be modeled in equations (6)~(11). The function table shown in Fig.6 is used to explained equation (8)~(11). Take Fig.6(a) in conjunction with equation (8) as an example, in order to produce various combinations in 0 and 1 for $C_{out:}$ when C_{out} is 0, $(0)\overline{X_0}$ covers the three combination of $(Y_1\ Y_0)$ $(X_2\ X_1\ X_0)$ equal to (0 0)(0 0 0), (0 1)(0 0 0), (1 1)(0 0 0) respectively; when C_{out} is 0, $(0)f_2$ covers the three combinations of $(Y_1\ Y_0)$ $(X_2\ X_1\ X_0)$ equal to (0 0)(0 0 1), (0 1)(0 0 1), (1 1)(0 0 1) respectively; when C_{out} is 0, Y_1f_1 covers the three combinations of $(Y_1\ Y_0)$ $(X_2\ X_1\ X_0)$ equal to (0 0)(0 1 1), (0 1)(0 1 1) respectively and $(Y_1\ Y_0)$ $(X_2\ X_1\ X_0)$ equal to (1 1)(0 1 1) when $C_{out:}$ is 1; Y_0X_2 covers the three combinations of $(Y_1\ Y_0)$ $(X_2\ X_1\ X_0)$ equal to (0 0)(1 1 1) when $C_{out:}$ is 0, and $(Y_1\ Y_0)$ $(X_2\ X_1\ X_0)$ equal to (0 1)(1 1 1), (1 1)(1 1 1) respectively when $C_{out:}$ is 1. Others such as Fig.6(b)**:**K_0, Fig.6(c)**:**K_1 and Fig.6(d)**:**K_2 can be deduced similarly.

$$f_1 = \overline{X_2} \cdot X_1 \quad (6)$$

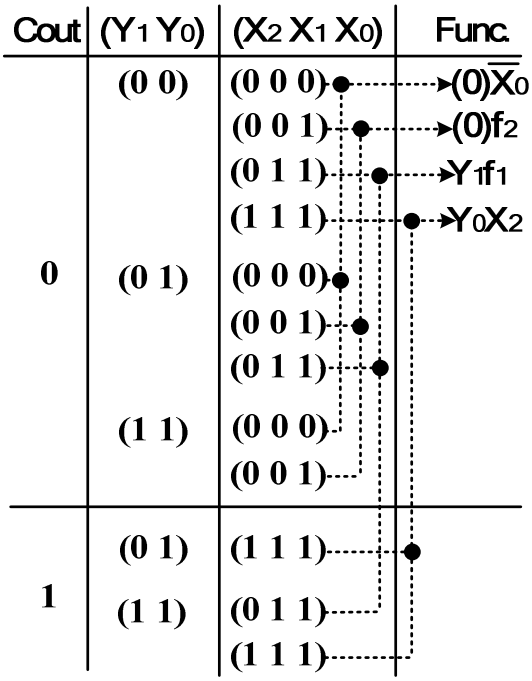
$$f_2 = \overline{X_1} \cdot X_0 \quad (7)$$

$$C_{out} = (Y_1)\, f_1 + (0)\, f_2 + (0)\,\overline{X_0} + (Y_0)\, X_2 \quad (8)$$

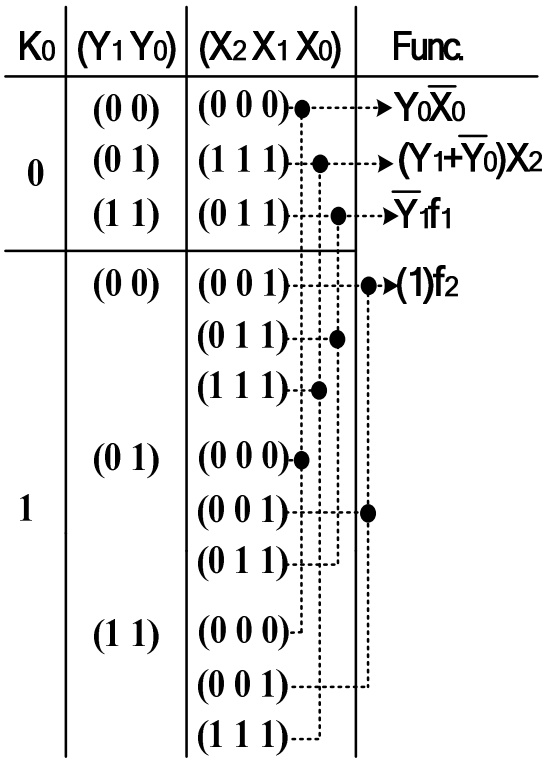
$$K_0 = (\overline{Y_1})\, f_1 + (1)\, f_2 + (Y_0)\,\overline{X_0} + (Y_1 + \overline{Y_0})\, X_2 \quad (9)$$

$$K_1 = (\overline{Y_1})\, f_1 + (Y_0)\, f_2 + (Y_1)\,\overline{X_0} + (Y_0)\, X_2 \quad (10)$$

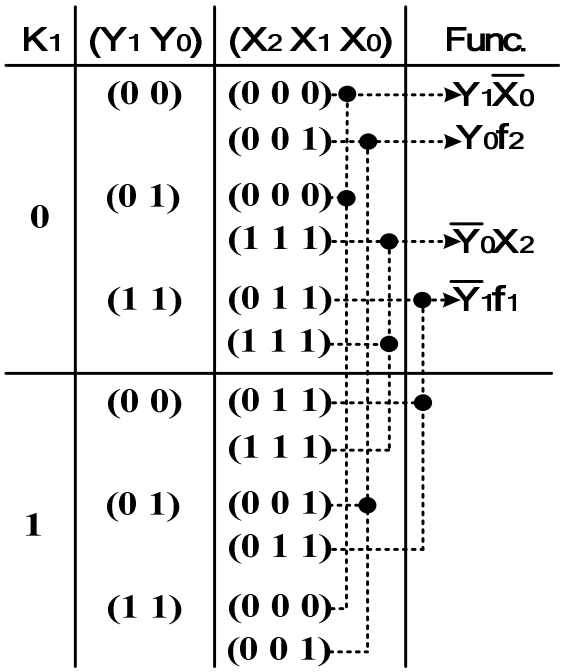
$$K_2 = (\overline{Y_1} \cdot Y_0)\, f_1 + (Y_1)\, f_2 + (0)\,\overline{X_0} + (\overline{Y_0})\, X_2 \quad (11)$$



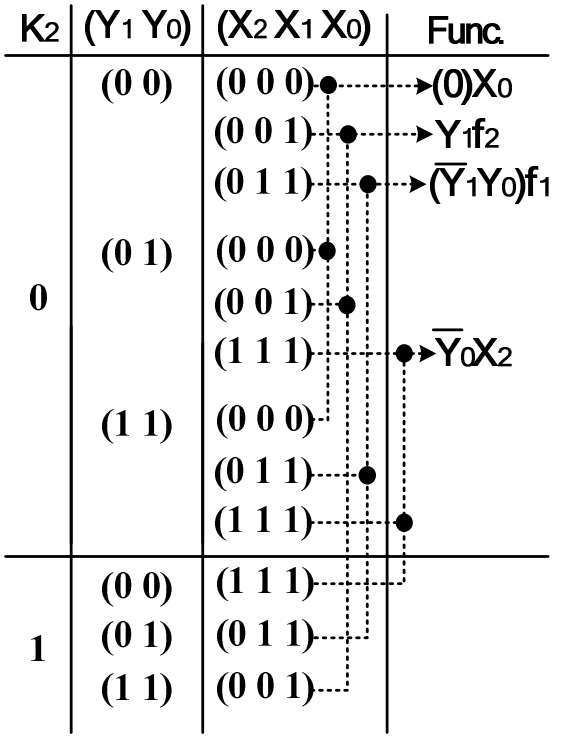
(a)



(b)



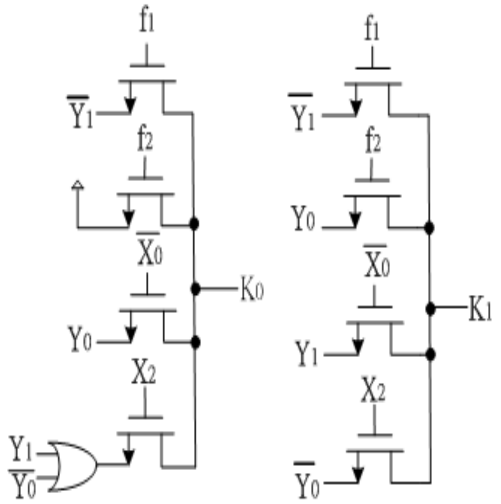
(c)



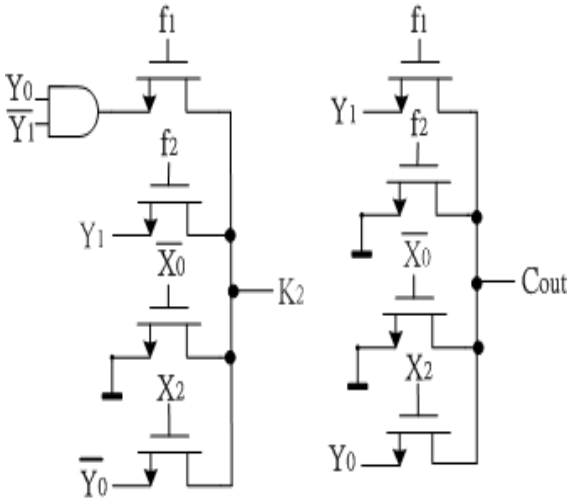
(d)

Fig.6 The function table of K_0,K_1,K_2,C_{out}

Fig. 7 depicts the circuit of PR module in detail. What the PS module does is transferring the previous stage into higher beads, and equation (12) - (14) model the behavior of PS module.



(a)



(b)

Fig.7 Detailed circuit of the PR module.

$$O_0 = X_0 + C_{in} \quad (12)$$

$$O_1 = X_1 + C_{in} X_0 \quad (13)$$

$$O_2 = 0 \quad (14)$$

Below is an example that demonstrates the BPA algorithm: $(B_3B_2B_1B_0)_2 = (1101)_2 = 13$ and $(A_1A_0)_2 = (10)_2 = 2$. $(1101)_2$ * $(10)_2$ = $(001|011|011)_{abacus}$ = (0+0+1)*16 + (0+1+1)*4+ (0+1+1)*1 = 26.

2.2 The PA (Parallel Addition) Module:

Similar to multiplexers, this module can count two column elements of the same value simultaneously and convert the sum into a thermometric representation K_0~K_5, in which $0 \leqq K_i \leqq K_j \leqq 1$ for i > j.

From Fig. 4, one can clearly see that the numbe $(X_2X_1X_0)$ is the input signal of the multiplexer, while $(Y_2Y_1Y_0)$ is the selector used to modify the number $(X_2X_1X_0)$. Together they produce the thermometric sum $(K_5K_4K_3K_2K_1K_0)$. Note that there are only four configurations for each number $(X_2X_1X_0)$ or $(Y_2Y_1Y_0)$, i.e., 000, 001, 011, and 111.

Equations (15) – (22) model the behavior of PA module. The function table shown in Fig.8 is used to explained equation (17) ~ (22). Take Fig.8(a) in conjunction with equation (17) as an example, in order to produce various combinations in 0 and 1 for K_0 **:** covers the four combinations of $(Y_2Y_1Y_0)$ $(X_2X_1X_0)$ equals to (0 0 0)(0 0 0) when K_0 is 0, and $(Y_2Y_1Y_0)$ $(X_2X_1X_0)$ equal to (0 0 1)(0 0 0), (0 1 1)(0 0 0), (0 0 1)(0 0 0) respectively when K_0 is 1; (1)$\mathbf{f_2}$ covers the four combinations of $(Y_2Y_1Y_0)$ $(X_2X_1X_0)$ equal to (0 0 0)(0 0 1), (0 0 1)(0 0 1), (0 1 1)(0 0 1), (1 1 1)(0 0 1) respectively when K_0 is 1; (1)f1 covers the four combinations of $(Y_2Y_1Y_0)$ $(X_2X_1X_0)$ equal to (0 0 0)(0 1 1), (0 0 1)(0 1 1), (0 1 1)(0 1 1), (1 1 1)(0 1 1) respectively when K_0 is 1; (1)X_2 covers the four combinations of $(Y_2Y_1Y_0)$ $(X_2X_1X_0)$ equal to (0 0 0)(1 1 1), (0 0 1)(1 1 1), (0 1 1)(1 1 1), (1 1 1)(1 1 1) when K_0 is 1. Others such as Fig.8(b):K_1, Fig.8(c):K_2, Figure 8(d):K3, Fig.8(e):K_4, Fig.8(f):K_5 can be deduced similarly.

$$f_1 = \overline{X_2} \cdot X_1 \quad (15)$$

$$f_2 = \overline{X_1} \cdot \ X_0 \quad (16)$$

$$K_0 = (1)\, f_1 + (1)\, f_2 + (Y_0)\, \overline{X_0} + (1)\, X_2 \quad (17)$$

$$K_1 = (1)\, f_1 + (Y_0)\, f_2 + (Y_1)\, \overline{X_0} + (1)\, X_2 \quad (18)$$

$$K_2 = (Y_0)\, f_1 + (Y_1)\, f_2 + (Y_2)\, \overline{X_0} + (1)\, X_2 \quad (19)$$

$$K_3 = (Y_1)\, f_1 + (Y_2)\, f_2 + (0)\, \overline{X_0} + (Y_0)\, X_2 \quad (20)$$

$$K_4 = (Y_2)\, f_1 + (0)\, f_2 + (0)\, \overline{X_0} + (Y_1)\, X_2 \quad (21)$$

$$K_5 = (0)\, f_1 + (0)\, f_2 + (0)\, \overline{X_0} + (Y_2)\, X_2 \quad (22)$$

K_0	$(Y_2Y_1Y_0)$	$(X_2X_1X_0)$	Func.
0	(0 0 0)	(0 0 0)	$Y_0\overline{X_0}$
	(0 0 0)	(0 0 1)	(1)f_2
		(0 1 1)	(1)f_1
		(1 1 1)	(1)X_2
	(0 0 1)	(0 0 0)	
		(0 0 1)	
		(0 1 1)	
		(1 1 1)	
1	(0 1 1)	(0 0 0)	
		(0 0 1)	
		(0 1 1)	
		(1 1 1)	
	(1 1 1)	(0 0 0)	
		(0 0 1)	
		(0 1 1)	
		(1 1 1)	

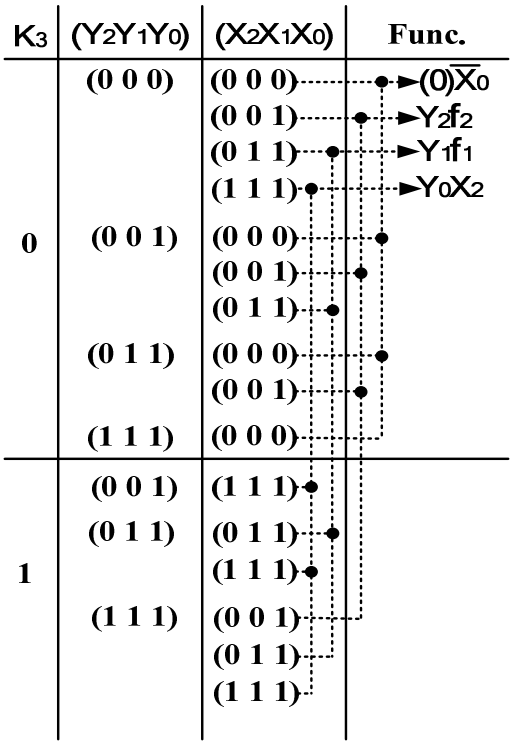
(a)

K_1	$(Y_2Y_1Y_0)$	$(X_2X_1X_0)$	Func.
0	(0 0 0)	(0 0 0)	$Y_1\overline{X_0}$
		(0 0 1)	Y_0f_2
	(0 0 1)	(0 0 0)	
	(0 0 0)	(0 1 1)	(1)f_1
		(1 1 1)	(1)X_2
	(0 0 1)	(0 0 1)	
		(0 1 1)	
		(1 1 1)	
1	(0 1 1)	(0 0 0)	
		(0 0 1)	
		(0 1 1)	
		(1 1 1)	
	(1 1 1)	(0 0 0)	
		(0 0 1)	
		(0 1 1)	
		(1 1 1)	

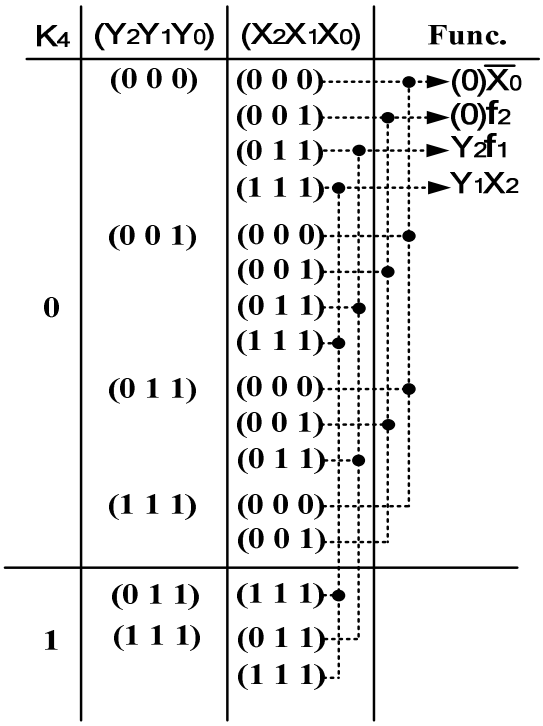
(b)

K_2	$(Y_2Y_1Y_0)$	$(X_2X_1X_0)$	Func.
0	(0 0 0)	(0 0 0)	$Y_2\overline{X_0}$
		(0 0 1)	Y_1f_2
		(0 1 1)	Y_0f_1
	(0 0 1)	(0 0 0)	
		(0 0 1)	
	(0 1 1)	(0 0 0)	
	(0 0 0)	(1 1 1)	(1)X_2
	(0 0 1)	(0 1 1)	
		(1 1 1)	
1	(0 1 1)	(0 0 1)	
		(0 1 1)	
		(1 1 1)	
	(1 1 1)	(0 0 0)	
		(0 0 1)	
		(0 1 1)	
		(1 1 1)	

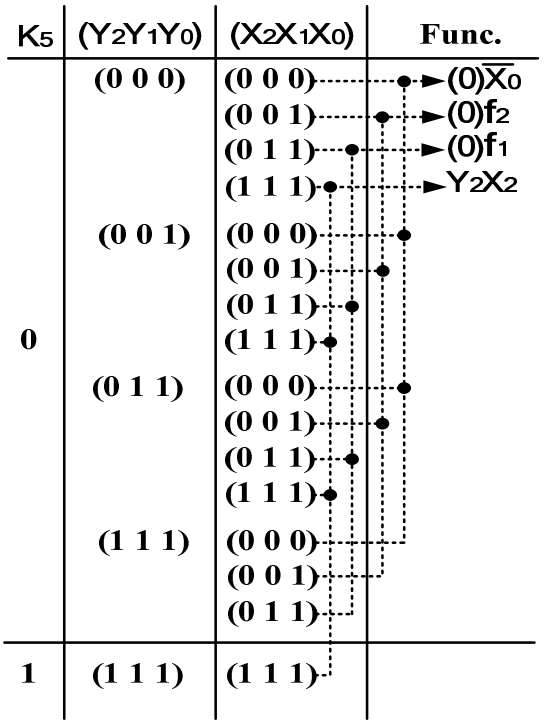
(c)



(d)



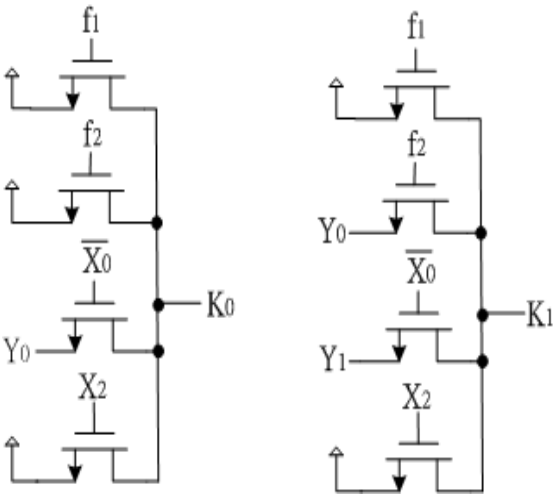
(e)



(f)

Fig.8 The function table of K_0,K_1,K_2, K_3,K_4,K_5

Fig. 9 shows the detailed circuits. It is evident that the PA module can count all the beads simultaneously.



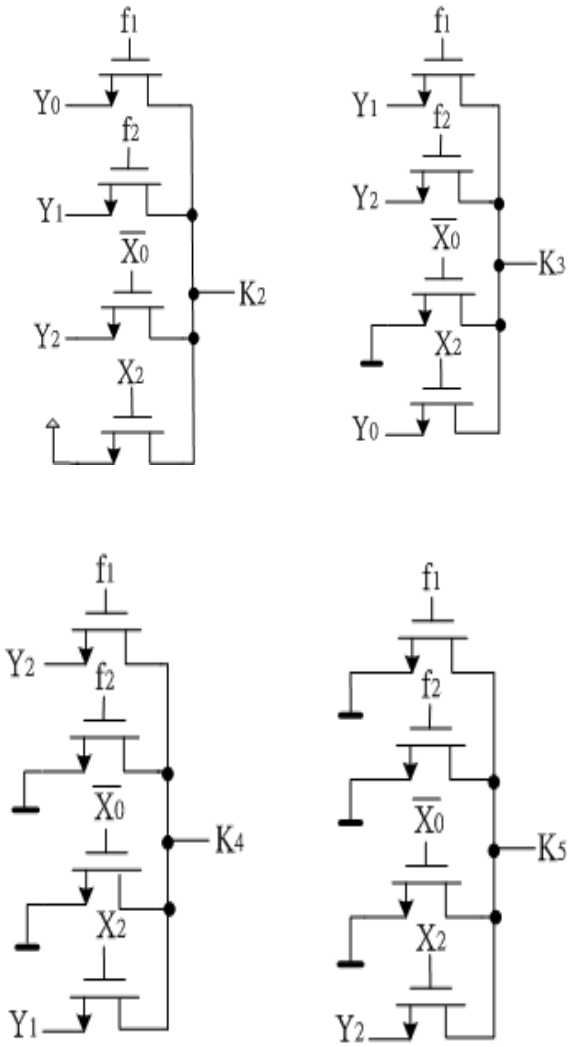


Fig.9 The circuit of ($K_0K_1K_2K_3$ K_4K_5) in PA module.

2.3 The TB (Thermometric to Binary) Transformation Module

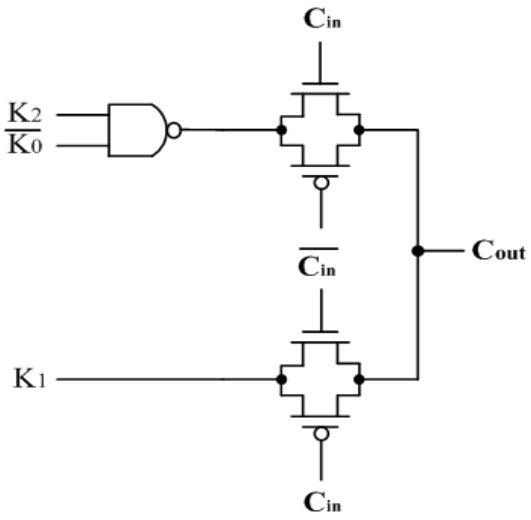
The function of this module is to transform thermometric representation to binary numbers, where either the higher or lower part numbers of K_5 - K_0 are converted into binary numbers as shown in Fig.4. The following equations can help to determine the output signals S_1, S_0 and C_{out} are determined using the following equations:

$$C_{out} = C_{in}K_2 + K_3 \quad (23)$$

$$S_0 = \overline{C_{in}}\ \overline{K_1}\ K_0 + \overline{C_{in}}\ \overline{K_3}\ K_2 + \overline{C_{in}}\ \overline{K_5}\ K_4 + C_{in}\overline{K_0} + C_{in}\overline{K_2}K_1 + C_{in}\overline{K_4}K_3 + C_{in}K_5 \quad (24)$$

$$S_1 = \overline{C_{in}}\ \overline{K_3}\ K_1 + C_{in}\overline{K_2}K_0 + C_{in}K_4 + K_5 \quad (25)$$

The detailed circuits of the TB module are depicted in Fig.10.



(a)

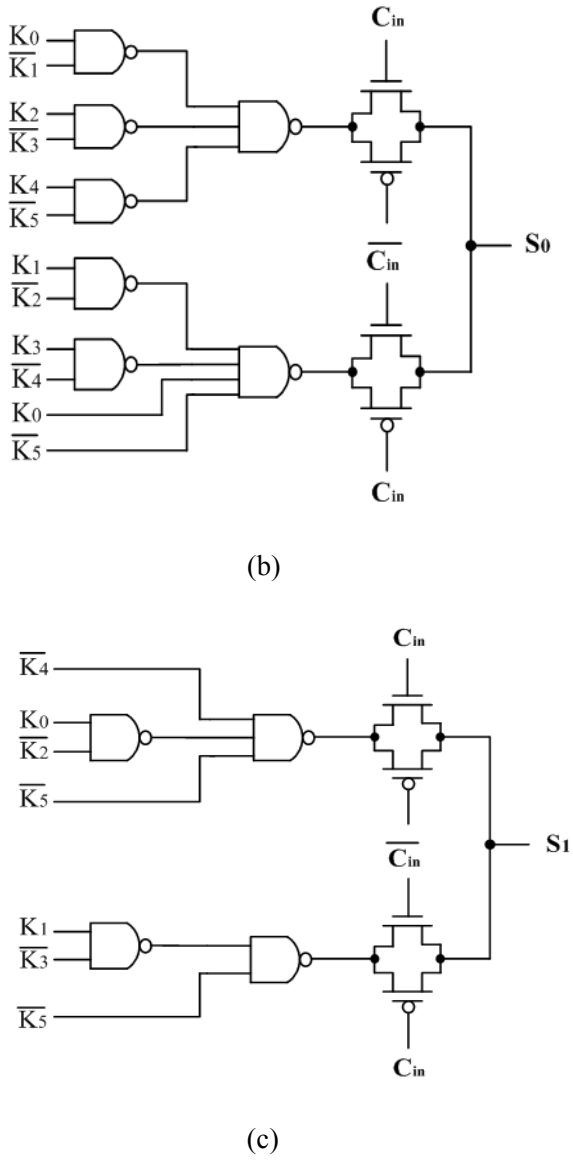
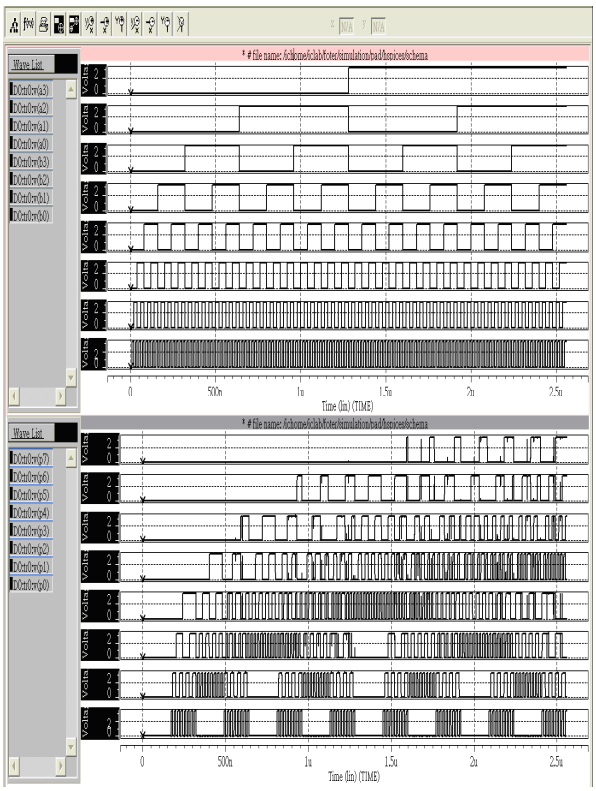


Fig.10 (a) The circuit of C_{out} in TB block, (b) The circuit of S_0 in TB block, (c)The circuit of S_1 in TB block.

3 Simulations and Comparisons

In the previous sections, we looked at the Chinese abacus and constructed a prototype multiplier based on its principles. In order to simulate the circuits of this prototype multiplier, HSPICE as well as the 0.35μm and 0.18μm TSMC CMOS technology were employed, while the length and the width of the transistors were kept as small as this technology would allow to simplify the process. For the PMOS transistors, the width was two and a half times longer than the NMOS transistors. For each output in the simulation, this article used CMOS inverters as the loads. Finally, the simulation ran on the 0.35μm CMOS technology at a frequency of 50 MHz, and the output curve, input curve, and power consumption curve were plotted as shown in Fig. 11 and 12 respectively.



.

Fig.11 Simulation output and input waves of the 4-bit Chinese abacus multiplier

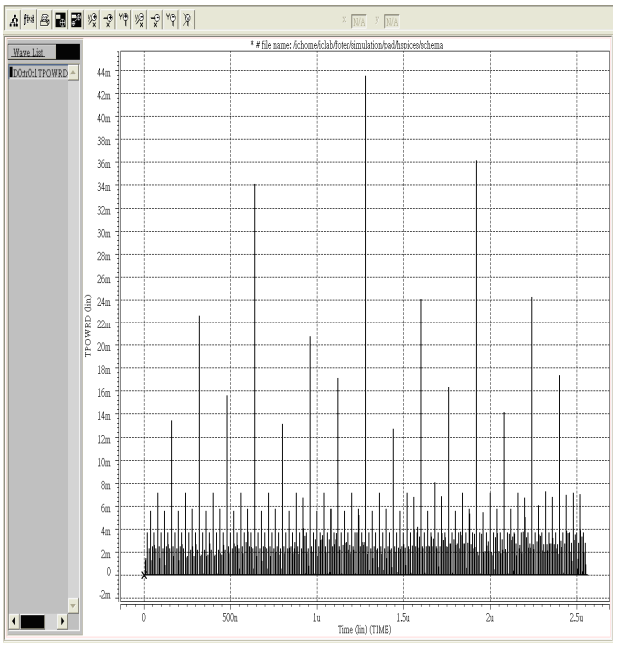


Fig.12 Simulation power consumption of the 4-bit Chinese abacus multiplier.

The simulation results are compared with those of the Braun array multiplier, as listed in Table 1.The delay is defined as the longest signal time from input to output with all input patterns. The 4x4 abacus multiplier results a delay of 2.83ns and 1.432ns for 0.35μm and 0.18μm TSMC CMOS technologies, respectively. These data are 19.7% and 10.6% less than those of the Braun array multiplier for the same 0.35μm and 0.18μm technologies, respectively.

The power consumption levels of various multipliers using different methodologies is also listed in Table 1.The average results of the power consumption are different for all input patterns. The simulation was performed at a frequency of 50MHz. For the 4-bit abacus multiplier power consumptions were 8.7 % and 18% less than those of the Braun array multiplier with 0.35μm and 0.18μm technologies,respectively.From these results, we conclude that the abacus multiplier still has competitive with the Braun array multiplier. The chip layout of the 4x4 abacus multiplier is shown in Fig.13.

TABLE 1. Simulation results of the 4x4 array multiplier and abacus multiplier

	Tech.	[14]	Braun	Abacus	Reduction % (compare to Braun)
Delay (ns)	0.35	-	3.39	2.83	19.7%
	0.18	3.97	1.584	1.432	10.6%
Power (μw)	0.35	-	313	288	8.7%
	0.18	145.5	45.2	38.3	18%
P*D (μw*ns)	0.35	-	1061	815	23.2%
	0.18	577.6	71.6	54.8	23.5%

4 Conclusion

This study put forward a multiplier based on algorithm of the Chinese abacus and simulated all results using both the 0.18μm and 0.35μm TSMC CMOS technology. The simulation results showed that delay level of the 4x4 bits abacus multiplier is 19.7% and 10.6% better respectively in 0.35μm and 0.18μm technology than those of Braun array multiplier; furthermore, power consumption of the 4x4 bits abacus multiplier showed an 8.7 % and 18% improvement respectively in 0.35μm and 0.18μm technology than those of Braun array multiplier. Therefore, it is obvious that the abacus multiplier proposed by this study has a competitive advantage over the conventional fast multipliers, as it can significantly reduce delay and power consumption.

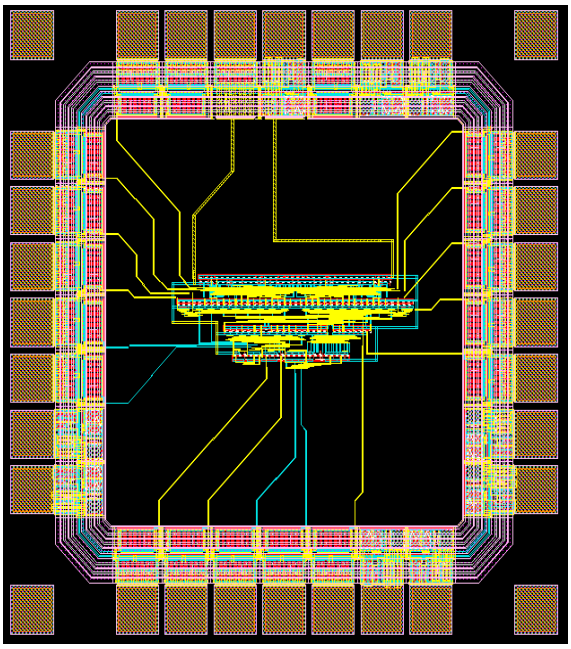


Fig.13 The chip layout of 4x4 abacus multiplier in 0.35μm TSMC CMOS technology.

References:

[1] S. D. Pezaris, A 40-ns 17-bit by 17-bit array multipliers, *IEEE Transactions on Computers*, Vol. 20, April 1971, pp. 442-447.

[2] K.Z. Pekmestzi, Multiplexer-based array multipliers, *IEEE Transactions on Computers*, Vol. 48, No. 1, Jan. 1999, pp. 15-23.

[3] C. Wallace, A suggestion for a fast multiplier, *IEEE Transactions on Electronic Computers*, Vol. 13, 1964, pp. 14-17.

[4] Hwang-Cherng Chow, I-Chyn Wey and Hsing-Chung Liang, High Speed Pipelined Booth Multiplier, *WSEAS Transactions on Circuits and Systems,* Issue 5, Vol. 4, May 2005,pp.495-505

[5] S.Anipana, P. A. Janakiraman, Serial to Parallel Conversion of Pulse-rate Signals using Binary Rate Multiplier, *Principle,India Annual conference proceedings of the IEEE INDICION*, 20-22 Dec. 2004,pp.359-362.

[6] Kenny Johansson, Oscar Gustafsson, and Lars Wanhammar, Trade-Offs in Multiplier Block Algorithms for Low Power Digit-Serial FIR Filters, *Proceedings of the 10th WSEAS International Conference on CIRCUITS,* Vouliagmeni, Athens, Greece, July 10-12, 2006,pp324-329

[7] Yang Ai-Min, Zhang Wen-Xiang, A Fast Multiplier Design over Composite Fields, *WSEAS Transactions on Systems*, Issue 3, Vol. 6, Mar. 2007,pp.444-448

[8] Jenn-Shyong Horng,I-Chang Jou1 and Chiou-Yng Lee, On complexity of normal basis multiplier using modified Booth's algorithm, *Proceedings of the 7th WSEAS International Conference on Applied Informatics and Communications*, Athens, Greece, Aug. 24-26, 2007,pp12-17

[9] Guoping Wang, A Reconfigurable Unsigned/Signed Binary Multiplier, *WSEAS Transactions on Circuits and Systems*, Issue 5, Vol. 3, July 2004,pp.1079-1084

[10] Franco Maloberti and Chen Gang, The Chinese Abacus method: can we use it for digital arithmetic, *Proceedings of the 8th Great Lakes Symposium on VLSI*, 19-21, Feb. 1998 pp. 192 – 195.

[11] Franco Maloberti and Chen Gang, Use of the Chinese Abacus method for digital arithmetic functions, *Proceedings of the 1998 IEEE International Symposium on Circuits and Systems*, Vol. 5, 31 May - 3 June 1998 pp. 213 – 216.

[12] Franco Maloberti and Chen Gang, Performing Arithmetic Functions with the Chinese Abacus Approach, *IEEE Transaction on circuits and systems-II: Analog and digital signal processing*, Vol. 46,No. 12, Dec. 1999, pp. 1512 – 1515.

[13] Shu-Chung Yi., Kun-Tse Lee., Jin-Jia Chen., Chien-Hung Lin., Chuen-Ching Wang., Chin-Fa Hsieh and Chih-Yung Lu., The new architecture of radix-4 Chinese abacus adder, *IEEE International Symposium on Multiple-Valued Logic*, 17-20 May. 2006, pp. 12-15.

[14] Vasefi, F., and Abid, Z, Low power n-bit adders and multiplier using lowest-number-of-transistor 1-bit adders, *Canadian Conference on Electrical and Computer Engineering*, May. 2005, pp. 1731 – 1734.

[15] Shu-Chung Yi, Zi-Yi Zhao, Chien-Hung Lin, Yu-Zhi Xie, Yen-Ju Chen, Yi-Jie Lin, The novel Chinese abacus adder, *IEEE International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*, Ambassador Hotel, Hsinchu, Taiwan, April 25-27, 2007, pp. 270–273.