Low-Error and Area-Efficient Fixed-Width Multiplier by Using Dual Group Minor Input Correction Vector and Recovering Input Correction Vector Over-Compensation

魏一勤　　　王俊傑
I-Chyn Wey,　Chun-Chien Wang

Department of Electrical Engineering, Graduate Institute of Electrical Engineering, and Green Technology Research Center, Chang Gung University, Taiwan, ROC.
Email: icwey@mail.cgu.edu.tw

摘要-本篇論文利用分群的次高權重部分乘積項，協助截斷電路中權重最高的部分乘積項共同進行誤差補償，以增進錯誤修正能力；進一步限制截斷電路中最高權重部分乘積項的補償值，降低過度補償誤差；進而透過邏輯化簡，將補償電路複雜度降低。在16×16位元固定寬度乘法器上，相較於截斷乘法器能修正87%的誤差，且只需要完整乘法器53%的電路面積。

I.　　介紹

在許多高速數位訊號處理與多媒體應用中，乘法運算一直扮演著極重要的角色，因為它主導著晶片面積大小及運算速度快慢。然而在高效能乘法器(high-performance multiplier)的架構中，為了增加其運算速度與降低硬體成本，並避免乘法運算位元寬度無限增長，就必須減少部份乘積的個數以及縮短部份乘積累加時所需的運算時間。此時，通常會截斷乘法運算中低權重位元(less significant bit，LSB)乘積的輸出。然而，由於進位的改變會發生大量的截斷誤差(truncation error)，進而降低實際的應用。

許多已發表的文獻對於截斷誤差提出不同的分析與補償。他們的目的都是一致的，以少量的硬體來設計誤差補償電路，降低截斷誤差。觀察這些文獻的補償方式，大致可分為兩大類：固定常數校正值(fixed constant correction value)[1]-[3]與變動校正值(variable correction value)[4]-[7]二種補償方式。雖然固定常數校正值補償有比變動校正值補償簡單的誤差補償電路的結構，但其固定的誤差補償方式無法像變動校正值補償般隨著誤差補償輸入的不同而作彈性的變化，因此變動校正補償方式的錯誤修正效果較好。

在已發表的變動校正值補償方式中，文獻[4]提出以乘積輸出結果為一的項次所發生的平均機率作為逼近理想誤差補償的函式，然後設計出一個漣波架構(ripple architecture)的誤差補償電路，可以降低固定寬度乘法器的截斷誤差。

文獻[5]使用statistical analysis and linear regression analysis，發現截斷掉的低位元部份乘積中權重最高的部分乘積總和直接當作誤差補償，可以比文獻[4]更逼近理想值。且其截斷掉的低位元部份乘積中權重最高的部分乘積沒有經過邏輯閘，而是直接輸入固定寬度乘法器作補償，因此誤差補償電路面積可較文獻[4]更小。

文獻[7]也是以多重輸入誤差補償結構為基礎，且首度以截斷掉的低位元部份乘積中權重最高的部分乘積權重作為補償的根據，低權重的部份乘積以樹狀加法器結構進行補償，高權重的部份乘積以modified addition進行補償。由於考慮的補償因素比文獻[5]多，在對理想補償的逼近上也比文獻[5]好。

但以上的文獻卻都未考慮截斷掉的低位元部份乘積中次高權重的部分乘積對誤差補

償的影響，會使原本以最高權重的部份乘積為輸入的誤差補償電路發生多補償或少補償的錯誤補償，導致其無法逼近理想誤差補償值。

在本篇論文中，對於無號數固定寬度乘法器提出一個新的誤差補償電路。首先，以文獻[7]的誤差補償電路為基礎，探討其補償失敗的項次，然後以截斷掉的低位元部份乘積中次高的部分乘積進行誤差補償，可以有效的降低截斷誤差。最後再對新的誤差補償電路做邏輯化簡以節省電路面積。

II. 固定寬度乘法器設計基礎

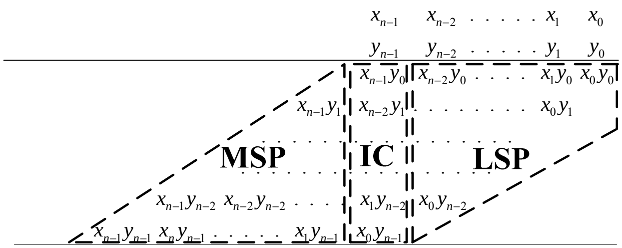
完整的無號數 Baugh-Wooley 陣列乘法器由 X 和 Y 為兩個無號數 n 位元的數字如(1)所示：

$$X=\sum_{i=0}^{n-1} x_i \cdot 2^i \quad Y=\sum_{i=0}^{n-1} y_i \cdot 2^i \tag{1}$$

一個乘法計算 P=X・Y 當作部分乘積 x_iy_j 的權重總和如(2)所示：

$$P=\sum_{k=0}^{2n-1} p_k \cdot 2^k=\sum_{j=0}^{n-1}\sum_{i=0}^{n-1} x_i y_j \cdot 2^{i \cdot j} \tag{2}$$

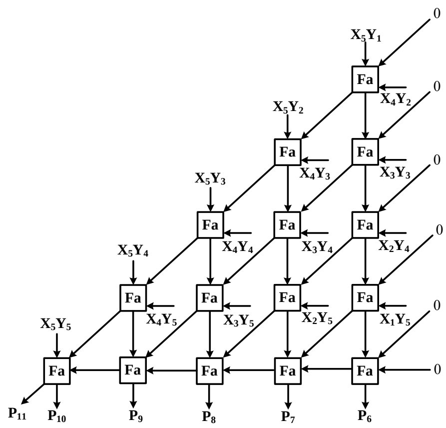
由式子(2)可以表示出一個完整的無號數n×n Baugh-Wooley部分乘積陣列顯示在圖一。



圖一 完整無號數n×n Baugh-Wooley乘法器部分乘積陣列

為了簡化討論與分析，我們將完整的無號數n×n Baugh-Wooley部分乘積陣列區分成三個子集MSP(most significant part)，IC(input correction vector)和LSP(less significant part)如圖一所示。

在DSP的應用和節省硬體成本上，通常只需要一個完整乘法器P_{2n-1}，…，P_n輸出的精準度作簡易地分析如文獻[4],[7]。因此刪除IC和LSP的部分乘積來簡化完整乘法器，形成一個固定寬度(Fixed-Width)乘法器如圖二所示。相較於一個完整乘法器，固定寬度乘法器的硬體複雜度大約是完整乘法器的一半，在面積和功率上都佔有優勢，而且有較短的最長路徑(Critical Path)，使延遲縮短。但是由於進位的改變，使固定寬度乘法器的輸出準確度降低，造成大量的截斷誤差。



圖二 六位元固定寬度乘法器架構

A. 定義精準度

觀察固定寬度乘法器的精確度是否可以因誤差補償電路而提升。我們定義2n位元完整乘法器與固定寬度乘法器間的誤差為(3)。

$$\varepsilon=P-P_t \tag{3}$$

其中P是完整乘法器的輸出，P_t是固定寬度乘法器的輸出定義為(4)。

$$\begin{aligned}P_t&=\sum_{j=1}^{n-1} y_j 2^{-j}\sum_{i=1}^{n-j} x_i 2^{-i}\\&\quad+f\left(x_1y_n,x_2y_{n-1},\ldots,x_{n-1}y_2,x_ny_1\right)\\&=\sum_{j=1}^{n-1} y_j 2^{-j}\sum_{i=1}^{n-j} x_i 2^{-i}+f\left(IC\right)\end{aligned} \tag{4}$$

其中f(IC)稱為誤差補償函式。我們考慮絕對平均誤差(absolute mean error, ε_{am})、均方誤差(mean square error, ε_{ms})和誤差變異數(variance of error, ε_v)定義為式子(5)，(6)和(7)。

$$\varepsilon_{am} = E\left\{|\varepsilon|\right\}/2^n \tag{5}$$

$$\varepsilon_{ms} = E\left\{\varepsilon^2\right\}/2^n \tag{6}$$

$$\varepsilon_{v} = E\left\{\left(\varepsilon - \varepsilon_{am}\right)^2\right\}/2^n \tag{7}$$

其中E{}表示平均運算子。根據(2)和(4)，ε可以寫成(8)。

$$\begin{aligned}\varepsilon &= P - P_t \\ &= S\left(x_1,\ldots,x_n;y_1,\ldots,y_n\right) - f\left(IC\right)\end{aligned} \tag{8}$$

其中$S(x_1,\ldots,x_n;y_1,\ldots,y_n)$=S(X;Y)是IC和LSB部份乘積的總和。

從式子(8)可以看出當 S(X;Y)和 f(IC)之間的差距越小，則精確度越高，表示誤差補償電路的實用性與效能越好。為了使精確度提高，所以我們令兩個理想誤差補償函式 f_{am} 和 f_{ms} 可以降低誤差 ε_{am} 和 ε_{ms}，然後注意到校正向量 IC 的每一個值 IC_0 可以從不同的 X 和 Y 獲得。以六位元為例，IC_0=(0,1,1,1,1,1)時可以獲得三個不同的(X;Y)值：(X;Y)=(011111_2;111111_2)，(X;Y)=(011111_2;111110_2) 和 (X;Y)=(111111_2;111110_2)，他們屬於在 IC=IC_0 下，X 和 Y 值的集合 $\Omega(IC_0)$元素，然後以 $N(IC_0)$=3 表示在集合 $\Omega(IC_0)$中的元素數量為三。利用 $N(IC_0)$和 S(X;Y)求得平均值，即可分析出 IC_0 權重，進而得到 IC_0 中部份乘積代表的權重。式子如(9)。

$$S_{avg}\left(IC_0\right) = \frac{1}{N\left(IC_0\right)} \sum_{(X,Y)\in\Omega(IC_0)} S\left(X,Y\right) \tag{9}$$

為了獲得 f_{ms} 函式，我們考慮當 X 和 Y 輸入屬於 $\Omega(IC_0)$的絕對平均誤差 ε_{am} 和均方誤差 ε_{ms}，式子如(10)和(11)。

$$\varepsilon_{am}\left(IC_0\right) = \frac{1}{N\left(IC_0\right)} \sum_{(X;Y)\in\Omega(IC_0)} \left(S\left(X;Y\right) - f_{am}\left(IC_0\right)\right)^2 \tag{10}$$

$$\varepsilon_{ms}\left(IC_0\right) = \frac{1}{N\left(IC_0\right)} \sum_{(X;Y)\in\Omega(IC_0)} \left(S\left(X;Y\right) - f_{ms}\left(IC_0\right)\right)^2 \tag{11}$$

全部的 ε_{am} 和 ε_{ms} 可以由 $\varepsilon_{am}(IC_0)$和 $\varepsilon_{ms}(IC_0)$ 乘上在 $\Omega(IC_0)$中元素的個數，然後加總獲得，式子如(12)和(13)。

$$\varepsilon_{am} = \sum_{IC_0} \varepsilon_{am}\left(IC_0\right) \frac{N\left(IC_0\right)}{2^{2n}} \tag{12}$$

$$\varepsilon_{ms} = \sum_{IC_0} \varepsilon_{ms}\left(IC_0\right) \frac{N\left(IC_0\right)}{2^{2n}} \tag{13}$$

從(10)和(11)式可以看出，選擇適當的 $f_{am}(IC_0)$和 $f_{ms}(IC_0)$來降低 $\varepsilon_{am}(IC_0)$和 $\varepsilon_{ms}(IC_0)$，就可以得到較小的誤差 ε_{am} 和 ε_{ms}。

藉由簡單的代數運算，式子(10)和(11)可以改寫成(14)與(15)。

$$\varepsilon_{am}\left(IC_0\right) = S_{avg}\left(IC_0\right) - f_{am}\left(IC_0\right) \tag{14}$$

$$\begin{aligned}\varepsilon_{ms}\left(IC_0\right) &= \left[S_{avg}\left(IC_0\right) - f_{ms}\left(IC_0\right)\right]^2 \\ &\quad + \frac{1}{N\left(IC_0\right)} \sum_{(A,B)\in\Omega(IC_0)} \left[S\left(x;y\right) - S_{avg}\left(IC_0\right)\right]^2\end{aligned} \tag{15}$$

透過(14)與(15)可以得知如果 $f_{am}(IC_0)$，$f_{ms}(IC_0)$ 與 $S_{avg}(IC_0)$有相同的數值，將可以使誤差 ε_{am} 和 ε_{ms} 縮至最小。為了便於觀察數據與準確性的分析，我們將 $S_{avg}(IC_0)$作正規化與捨位，$f_{am}(IC_0)$和 $f_{ms}(IC_0)$就會計算成式子(16)。

$$f_{ms}\left(IC_0\right) = Round_n\left[S_{avg}\left(IC_0\right)\right] \tag{16}$$

B. IC 權重分析

文獻[7]根據式子(9)和(16)觀察 S_{avg}(IC)和 f_{ms}(IC)整理出表一，得到一個誤差補償函式可以近似成一個輸入修正向量部份乘積的權重總和。以六位元固定寬度乘法器為例，當輸入修正向量的所有位元全為 0 時，S_{avg}(IC)為 0.224，因為小於 LSB/2=0.5，所以經過捨位後會使 f_{ms}(IC)得到 0。當輸入修正向量變成 IC=(0,0,0,0,0,1)，唯一部份乘積 x_6y_1 等於 1 時，S_{avg}(IC)會向上增加到 0.814，經過捨位 f_{ms}(IC)得到 1。由 IC=(0,0,0,0,0,1)和 IC=(0,0,0,0,0,0)的 S_{avg}(IC)相減可得 x_6y_1 的有效權重為 0.590。

假如我們考慮第三行，當只有部分乘積 x_4y_3 為 1 而其餘 IC 部份乘積為 0 時，與 IC=(0,0,0,0,0)的 S_{avg}(IC)相減可得 x_4y_3 的有效權重為 0.680 比 x_6y_1 的有效權重 0.590 高。

表一 不同 IC 所對應的 S_{avg} 和 f_{ms}

ROW	IC	S_{Avg}(IC)	f_{ms}(IC)
1	(000000)	0.224	0
2	(000001)	0.814	1
3	(000100)	0.904	1
4	(001100)	1.699	2

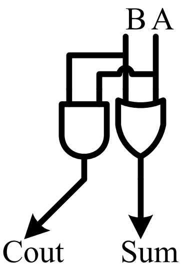
根據上述的討論，文獻[7]認為 S_{avg}(IC)不同權重是由於每個輸入修正向量元素的有效權重造成的。而且內部部分乘積(以六位元為例，如 x_4y_3 和 x_3y_4)比外部部分乘積(以六位元為例，如 x_6y_1 和 x_1y_6)有較大的權重。比起文獻[4]-[6]只使用 S_{IC} 的和去計算補償係數並且視 IC 中每個部份乘積有相同的權重，文獻[7]顯然可以更接近理想誤差補償。

C. IC 補償電路基本架構

針對以上對於 IC 中部份乘積的權重分析與討論，文獻[7]將誤差補償向量分成兩個子集並且使用兩個加法樹作誤差補償。

第一個加法樹，對於低權重部份乘積給定一個以標準全加器(FA)和半加器(HA)組合而成的一個標準計數器如文獻[7]。因此頭尾兩項部份乘積的權重為 2^{n-1}。

第二個加法樹，對於高權重部份乘積使用修正型半加器(mHA)，他是由一個或閘和及閘組成，所以稱為 AO 閘如圖二所示。其真值表如表二所示和標準半加器的真值表比較。可以發現唯一的不同就是當 A 和 B 輸入是 1 時，Sum=1 且 C_{out}=1。此外，從修正型半加器的真值表也可發現，當輸入總和有多少 1，輸出總和就有多少 1，因此高權重部份乘積的權重為 2^n。就好比輸入就是輸出，可省略 AO 單元。

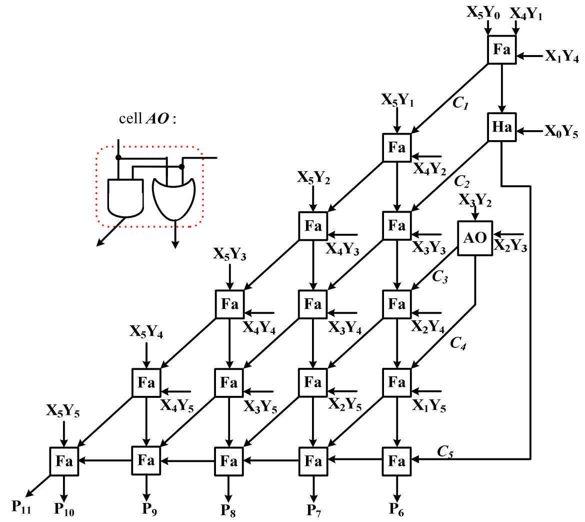


圖三 AO 電路圖(修正型半加器)

表二 修正型半加器真值表

Input		Output	
A	B	Cout	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

結合以上兩種加法樹所組成的誤差補償電路就稱之為 Dual-tree 結構的誤差補償電路，表示在圖四。



圖四 文獻[7]提出的 6×6 固定寬度乘法器

III. 所提出新的誤差補償電路設計

我們根據文獻[7]的補償結果做進一步的分析，並且以 β 表示 IC 中權重較低的頭尾各兩項部分乘積總和，如式子(17)

$$\beta = x_{n-1}y_0 + x_{n-2}y_1 + x_1y_{n-2} + x_0y_{n-1} \quad (17)$$

然後透過加法器電路得到輸出進位輸出的總

和 Carry，定義在(18)

$$Carry = C_1 + C_2 + C_{n-1} \quad (18)$$

最後比較 β 和 Carry 的關係得到 Carry_relation。以六位元為例，可以得到文獻[7] β 和 Carry 關係比較表，如表三所示。

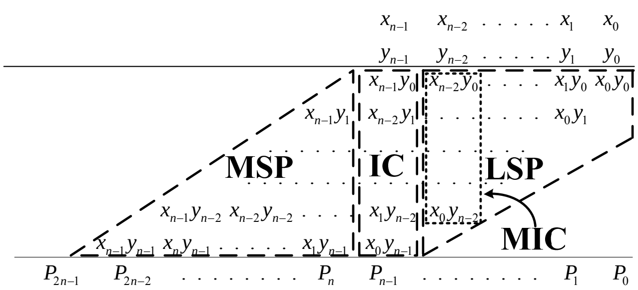
表三　文獻[7] β 和 Carry 關係比較表

β	Carry	Carry_relation
0	0	β
1	1	β
2	1	β-1
3	2	β-1
4	2	β-2

A. 誤差的選擇與補償

然而，文獻[7]的誤差補償結果，仍然有誤差補償錯誤發生。其中誤差補償錯誤可以分成兩類，第一種誤差補償錯誤，是由於誤差補償電路補償不足造成，會使輸出結果小於理想值。第二種誤差補償錯誤，是由於誤差補償電路補償過多造成的，會使結果大於理想值。本篇論文，主要針對 $|\varepsilon|>2^{n-1}$ 的誤差進行補償。因為 IC 誤差補償電路輸入補償的權重為 2^n，所以當誤差 $\varepsilon>2^{n-1}$ 時，少補償一位元相當於減少 2^n，就可以將文獻[7]的誤差縮小成 $|\varepsilon|<2^{n-1}$。同理，當誤差 $\varepsilon<-2^{n-1}$ 時，多補償一位元，誤差也可以縮小至 2^{n-1} 以下。

在本篇論文中，有別於其他文獻，僅以權重最高的 IC 部份乘積作為誤差補償。而是另外參考次高權重的部份乘積，在此定義為 minor input correction vector (MIC)如圖五。



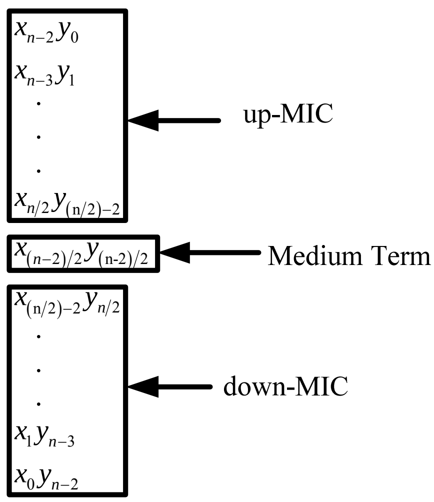
圖五 MIC 部份乘積陣列

其中MIC=($x_{n-2}y_0$，$x_{n-3}y_1$，…，x_1y_{n-3}，x_0y_{n-2})。根據文獻[7]對於IC部份乘積權重的分析，同理MIC中的部份乘積，也可以得到權重最高的中間項 (Medium Term) 部份乘積為 $x_{(n-2)/2}y_{(n-2)/2}$。以此部分乘積為中心向上向下分成 up-MIC($x_{n-2}y_0$，$x_{n-3}y_1$，…，$x_{n/2}y_{(n/2)-2}$) 和 down-MIC($x_{(n/2)-2}y_{n/2}$，…，x_1y_{n-3}，x_0y_{n-2})兩個部分乘積子集如圖六。然後，考慮MIC的進位補償權重為 2^n。由於產生 $x_{(n-2)/2}y_{(n-2)/2}$ 的進位路徑是MIC部分乘積中最長的，所以 $x_{(n-2)/2}y_{(n-2)/2}$ 為1機率也就是MIC部分乘積中最大的。但並不表示其餘MIC的部份乘積只要再有一個為1就可進位，畢竟 $x_{(n-2)/2}y_{(n-2)/2}$ 仍然有為0的可能。故為了確保MIC的部份乘積總和可以做有效進位補償，我們選擇up-MIC和down-MIC的總和為 $S_{up\text{-}MIC}$ 和 $S_{down\text{-}MIC}$ 表示在下面兩式。

$$S_{up\text{-}MIC} = x_{n-2}y_0 + x_{n-3}y_1 + \ldots + x_{n/2}y_{(n/2)-2} \quad (19)$$

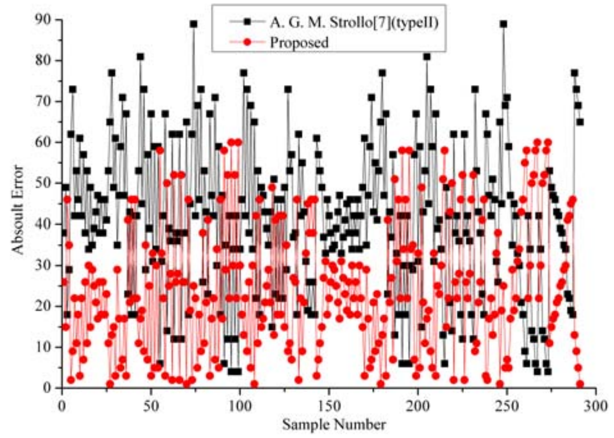
$$S_{down\text{-}MIC} = x_{(n/2)-2}y_{n/2} + \ldots + x_1y_{n-3} + x_0y_{n-2} \quad (20)$$

當(19)和(20)式都不為 0 時，進位才會成立。相反的，為了確保 MIC 的部份乘積總和不會進位，必須讓 $S_{up\text{-}MIC}$ 和 $S_{down\text{-}MIC}$ 皆為 0。

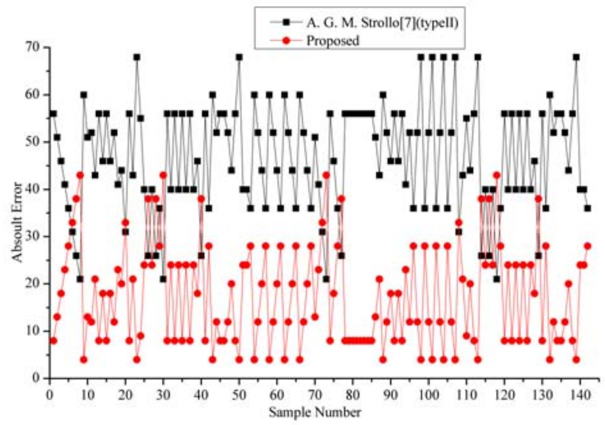


圖六 MIC 的兩個子集 up-MIC 和 down-MIC

當 β=2 或 β=4 時，根據表三的誤差補償方式會有許多補償不足的情形發生，透過我們所設計的 MIC 進位補償方式可以有效補足誤差。如圖七所示，大致上較大誤差的峰值都可以有效降低。



圖七 β=2 或 β=4 六位元固定寬度乘法器的絕對誤差與修正後絕對誤差比較

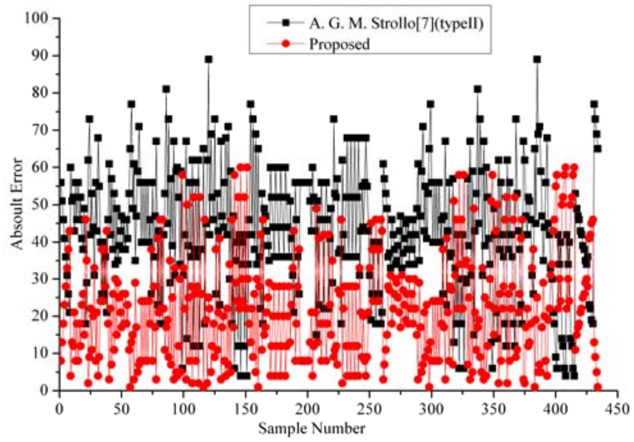


圖八 β=1 六位元固定寬度成法器的絕對誤差與修正後絕對誤差比較

而在 β=1 下 IC 其餘權重較重部份乘積($x_{n-3}y_2$，$x_{n-4}y_3$，⋯，x_3y_{n-4}，x_2y_{n-3})總和 S_{ICh} 如式子(21)

$$S_{ICh} = x_{n-3}y_2 + x_{n-4}y_3 + ... + x_3y_{n-4} + x_2y_{n-3} \quad (21)$$

當式子(21)不為 0 時，表三的誤差補償方式會導致補償過多以致產生額外的補償誤差。我們利用 MIC 沒有進位來判斷，限制讓 IC 部份乘積總和減少一位元作誤差補償，可以有效對於過度補償的情形進行修正。如圖八所示，經過補償值的限制設計後，過度補償的情形得以改善。結合以上兩種補償方式有助於降低文獻[7]中較大補償誤差項以及過度補償項的誤差，因此整體誤差補償效果得以提升，如圖九所示。



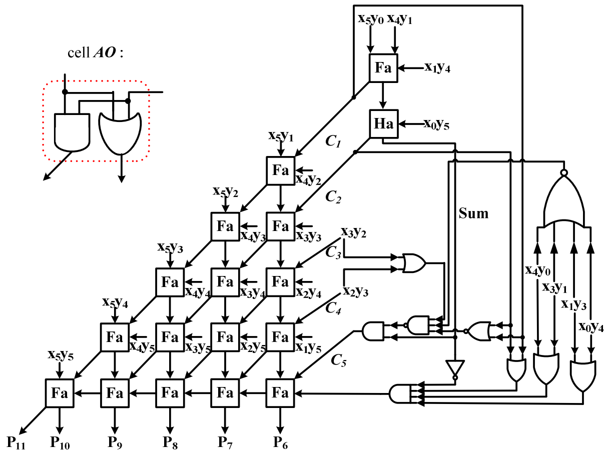
圖九 六位元固定寬度成法器的絕對誤差與修正後絕對誤差比較

B. 新的誤差補償電路設計架構

首先，我們保留文獻[7]的誤差補償電路架構如圖四。接著，將 C_1 和 C_2 作為 OR 閘的輸入，然後 OR 閘的輸出會和經過 inveter 反向的 Sum 合併輸入一個 4 輸入 AND 閘作為 β=2 或 β=4 判別的依據。兩個子集 up-MIC 部分乘積和 down-MIC 部分乘積各以一個 OR 閘連接，判別是否一個子集至少有一個部份乘積為 1。而這兩個 OR 閘的輸出也將併入一個 4 輸入的 AND，以搭配 β=2 或 β=4 的判別，進而輸出 1 對補償不足的誤差加以補償。

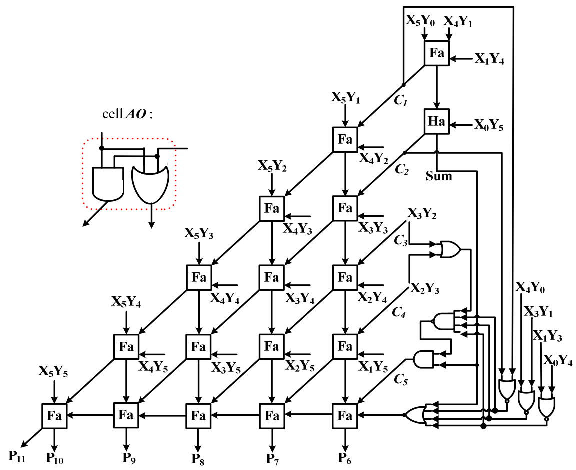
另外，為了取得 β=1，C_1 和 C_2 則當做 NOR 閘的輸入，因為 β=1 時不會有進位的產生。此時，up-MIC 部分乘積和 down-MIC 部分乘積會同時輸入一個 4 輸入 NOR 閘，限制部分乘積為 1 的產生，避免發生進位。而 IC 其餘權重較重部份乘積以 OR 連接，判別是否至少有一項部份乘積為 1。將此 OR 閘輸出和之前兩個 NOR 閘輸出合併輸入一個 3 輸入 NAND 閘。當以上條件成立時會輸出 0 給 AND 閘，當 AND 閘另外一個輸入 Sum 為 1 時，將不進

行誤差補償，以減少過度補償的誤差。最後，新的誤差補償電路顯示在圖十。



圖十　本文所提出新的六位元固定寬度乘法器誤差補償電路架構

為了縮小電路面積，降低誤差補償路成本。我們將三個輸進一個 4 輸入 AND 閘的 OR 閘轉換成三個 NOR 閘和三個 inveter。接下來，把這三個 inveter 和將 Sum 反向的 inveter 閘及一個 4 輸入 AND 閘化簡成一個 4 輸入 NOR 閘。而原本輸入一個 3 輸入 NAND 閘的兩個 NOR 閘，則改由此三個 NOR 閘的輸出作為 NAND 閘輸入，由於多加一個輸入，所以 3 輸入 NAND 閘將改成 4 輸入。最後，可以獲得一個低電路複雜度的誤差補償電路如圖十一所示。



圖十一　經過化簡的誤差補償電路

IV.　補償電路誤差修正效能比較

在此，我們探討新的誤差補償電路與文獻[7]對於直接截斷(Direct-truncated)完整乘法器的誤差補償效果分別從 6 位元，8 位元，12 位元，到 16 位元作分析與比較，定義出絕對平均誤差(absolute mean error, ε_{am})、均方誤差(mean square error, ε_{ms})和誤差變異數(variance of error, ε_v)對於截斷(Truncated)完整乘法器的誤差百分比。如以下式子所示。

$$\varepsilon_{am,\%}=\frac{\varepsilon_{am}}{\varepsilon_{am(Truncated)}}\times 100\% \tag{22}$$

$$\varepsilon_{ms,\%}=\frac{\varepsilon_{ms}}{\varepsilon_{ms(Truncated)}}\times 100\% \tag{23}$$

$$\varepsilon_{v,\%}=\frac{\varepsilon_{v}}{\varepsilon_{v(Truncated)}}\times 100\% \tag{24}$$

由以上式子可以看出位於分子的 ε_{am}、ε_{ms} 和 ε_v 的值越小，表示補償電路的修正誤差效果越佳，並且能有效提升固定寬度乘法器的精確度，使其更能運用在實際 DSP 應用中。因此，從表四到表六可以發現我們所提出的誤差補償電路能有效改善截斷誤差，無論是相對絕對平均誤差(absolute mean error, $\varepsilon_{am,\%}$)比較結果，相對均方誤差(mean square error, $\varepsilon_{ms,\%}$) 比較結果，或是相對誤差變異數(variance of error, $\varepsilon_{v,\%}$)比較結果，我們所設計的固定寬度乘法器的精確度表現都是其中最佳的。至於表七則是計算各文獻所提出的誤差補償電路所需要總電晶體個數佔完整乘法器總電晶體個數的百分比比較，定義成式子(25)。

$$R_{\%}=\frac{N_{Truncated}}{N_{Standard}}\times 100\% \tag{25}$$

結合表四和表七，將我們所提出的設計和誤差補償較精準的文獻[5], [6], [7]進行電路面積的比較，如圖十二和圖十三。從圖十二中，能看出在 8 位元固定寬度乘法器，本篇論文有最低的誤差，但電路面積稍大。然而，在圖十三中，16 位元固定寬度乘法器，本篇論文依舊保有最低的誤差，但電路面積已比大多數文

獻小。這顯示本篇論文所提出的固定寬度乘法器電路設計在輸入位元越大時，其誤差和面積優勢越顯著。

當電晶體個數消耗越多時，表示所需要的電路硬體成本就越高。圖十四中，我們將我們的設計和相近的設計文獻[4], [6], [7]比較，可以發現本篇所提出的誤差補償電路雖然在 6 位元固定寬度乘法器的電晶體消耗上比其他文獻高，然而到 16 位元固定寬度乘法器時，所需要的電晶體個數反而比起其他文獻來的低。顯然地，本篇論文所提出的誤差補償電路在電路面積的增長上相較於其他文獻所提出的誤差補償電路來的緩和，這主要是因為我們所設計的補償電路大部份都是固定式的，硬體複雜度並不會隨位元數增加而上升，如此的優勢，在越來越複雜的 DSP 應用上將更顯示出其有效節省電路成本的優勢。最後，我們透過電路合成，將我們所設計的低面積、高精確度 16 位元固定寬度乘法器實現於晶片中。圖十五為 16 位元固定寬度乘法器的晶片製程佈局圖，晶片面積為 18401.64μm^2。

表四　相對絕對平均誤差(absolute mean error, $\varepsilon_{am,\%}$)比較結果

Multiplier	n=6	n=8	n=12	n=16
D-Truncated structure	100% (80.25)	100% (448.25)	100% (11264.25)	100% (245760.25)
J. M. Jou[4]	46.44%	38.03%	27.47%	21.39%
S. J. Jou[5]	30.04%	23.64%	17.25%	14.06%
F. Curticapean[6]	27.22%	22.27%	16.92%	13.98%
A. G. M. Strollo[7](typeI)	26.62%	22.35%	17.38%	14.45%
A. G. M. Strollo[7](typeII)	26.46%	21.38%	16.03%	13.27%
Proposed paper	23.71%	19.59%	15.36%	13.06%

表五　相對均方誤差(mean square error, $\varepsilon_{ms,\%}$)比較結果

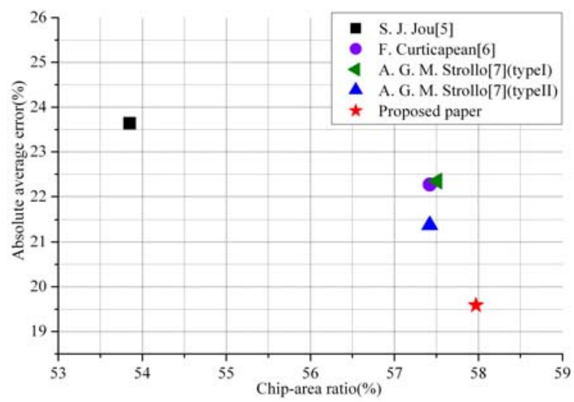
Multiplier	n=6	n=8	n=12	n=16
D-Truncated structure	100% (2.236)	100% (4.018)	100% (9.098)	100% (16.181)
J. M. Jou[4]	21.04%	14.89%	8.58%	5.59%
S. J. Jou[5]	9.85%	6.56%	3.82%	2.66%
F. Curticapean[6]	8.09%	5.82%	3.66%	2.62%
A. G. M. Strollo[7](typeI)	7.73%	5.86%	3.85%	2.78%
A. G. M. Strollo[7](typeII)	7.67%	5.38%	3.30%	2.37%
Proposed paper	6.02%	4.46%	3.04%	2.30%

表六　相對誤差變異數(variance of error, $\varepsilon_{v,\%}$)比較結果

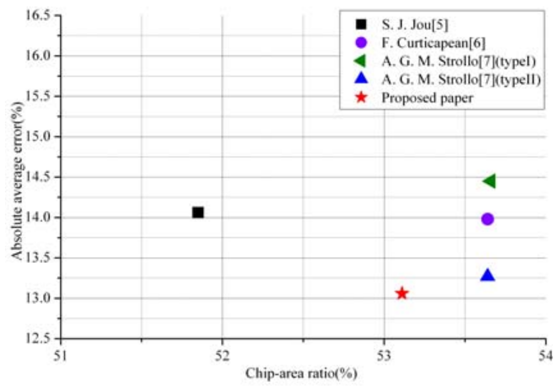
Multiplier	n=6	n=8	n=12	n=16
D-Truncated structure	100% (2318)	100% (58494)	100% (25532558)	100% (9088701326)
J. M. Jou[4]	19.51%	16.40%	13.72%	12.35%
S. J. Jou[5]	12.30%	9.94%	8.00%	7.21%
F. Curticapean[6]	10.08%	8.82%	7.64%	7.08%
A. G. M. Strollo[7](typeI)	9.67%	8.88%	7.97%	7.45%
A. G. M. Strollo[7](typeII)	9.66%	8.19%	6.94%	6.44%
Proposed paper	7.14%	6.79%	6.43%	6.27%

表七　電晶體個數比較結果

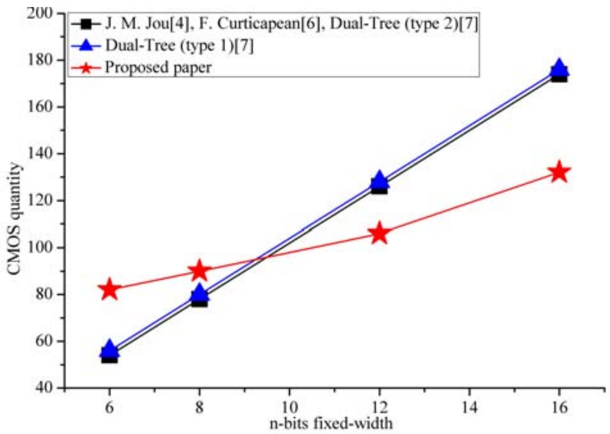
Multiplier	n=6	n=8	n=12	n=16
Standard	100% (1140)	100% (2184)	100% (5280)	100% (9720)
MSP	55.26%	53.85%	52.50%	51.85%
J. M. Jou[4]	60.00%	57.42%	54.89%	53.64%
S. J. Jou[5]	55.26%	53.85%	52.50%	51.85%
F. Curticapean[6]	60.00%	57.42%	54.89%	53.64%
A. G. M. Strollo[7](typeI)	60.18%	57.51%	54.92%	53.66%
A. G. M. Strollo[7](typeII)	60.00%	57.42%	54.89%	53.64%
Proposed paper	62.46%	57.97%	54.51%	53.11%



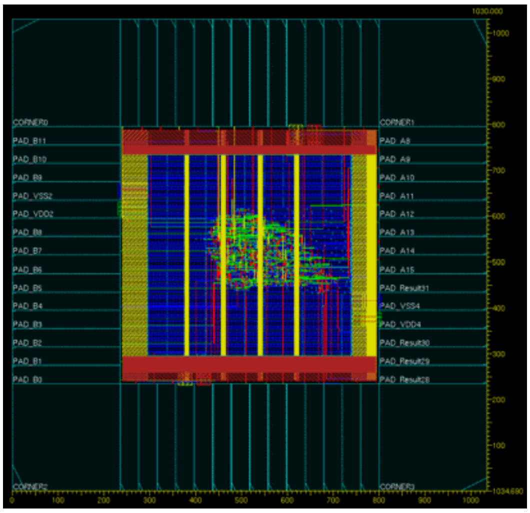
圖十二 八位元固定寬度乘法器絕對平均誤差對電路面積比例分佈圖



圖十三 十六位元固定寬度乘法器絕對平均誤差對電路面積比例分佈圖



圖十四　各乘法器誤差補償電路隨 6, 8, 12 和 16 位元輸入的增長趨勢



圖十五　本篇論文所設計的 16 位元固定寬度乘法器晶片佈局圖

V.　結論

本篇論文針對固定寬度乘法器的截斷誤誤差提出一個更為精確的誤差補方式，透過 MIC 的進位與 IC 補償之間的關係，隨著 β 值的改變，作彈性的誤差補償校正。經實驗結果，在 16 位元固定寬度乘法器的誤差補償上，相較於其他文獻能有更有效的利用硬體面積進行誤差補償。在絕對平均誤差，均方誤差和誤差變異數上均可達到比其他文獻更佳的誤差補償效果。因此，使用本篇論文所提出的固定寬度乘法器在 DSP 的應用上將更能顯示出其實用性。

REFERENCES

[1] Lim, Y. C., "Single-recision multiplier with reduced circuit complexity for signal processing applications," *IEEE Trans. On Computers*, Vol. 41, No. 10, pp. 1333-1336, 1992.

[2] Schulte, M. J. and Swartzlander, E. E., Jr., "Truncated multiplication with correction constant, " *Workshop on VLSI Signal Processing*, VI, pp. 388-396, 1993.

[3] Kidambi, S. S., El-Guibaly, F., and Antoniou, A., "Area-efficient multipliers for digital signal processing applications," *IEEE Trans. on Circuits & Systems II*, Vol. 43, No. 2, pp. 90-95, Feb. 1996.

[4] Jou, J. M., Kuang, S. R., and Chen, R. D., "Design of low-error fixed-width multipliers for DSP applications, " *IEEE Trans. on Circuits & Systems II*, Vol. 46, No. 6, pp. 836-842, June 1999.

[5] S. J. Jou, and H. H. Wang, "Fixed-Width Multiplier for DSP Application," IEEE International Symposium on Computer Design, Sept. 2000, pp318-322.

[6] F. Curticapean and J. Niittylahti, "A hardware efficient direct digital frequency synthesizer," in *Proc. IEEE Int. Conf. on Electronics, Circuits, and Systems (ICECS'01)*, vol. 1, St. Julians, Malta, Sep. 2–5, 2001, pp. 51–54.

[7] Antonio G. M. Strollo, Nicola Petra and Davide De Caro, "Dual-tree Error Compensation for High Performance Fixed-width Multipliers," *IEEE Transactions in Circuits and Systems II*, vol.52, no.8, pp.501-507 Aug. 2005.