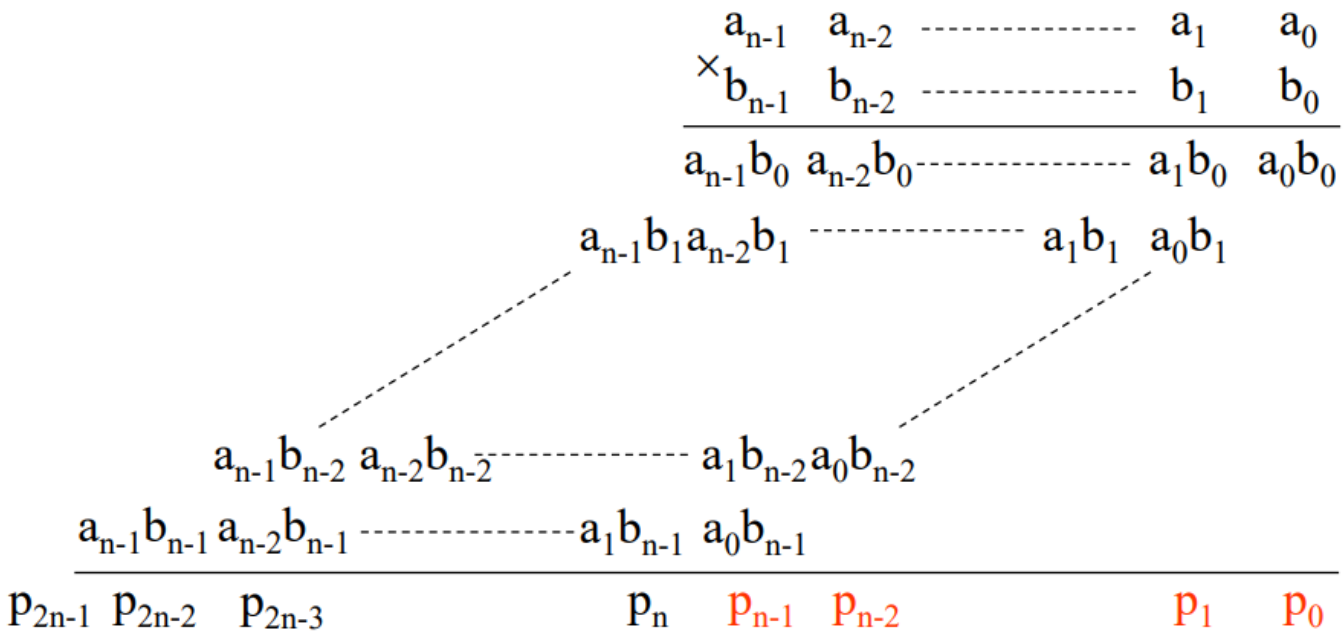
Traucated Multiplier

壹.設計原理與架構

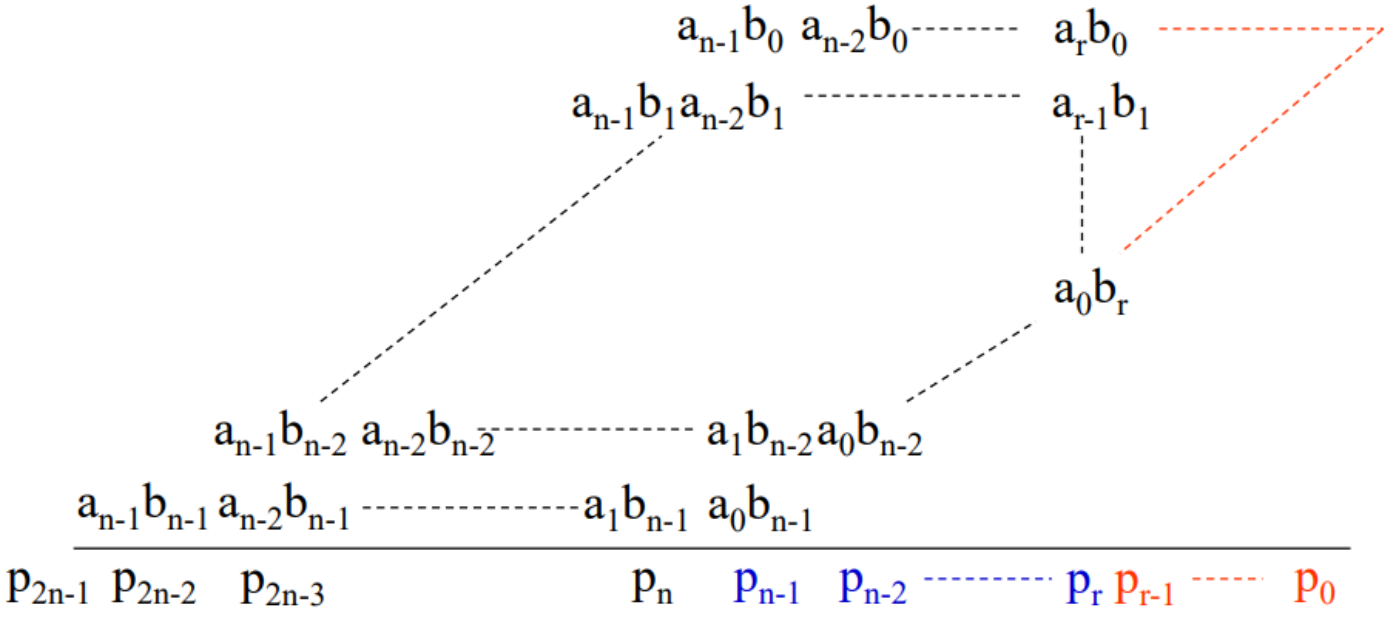
一、no_trunc

1.設計原理: unsigned a*b，用 bit by bit 相乘 row 相加，最後輸出 n bits。



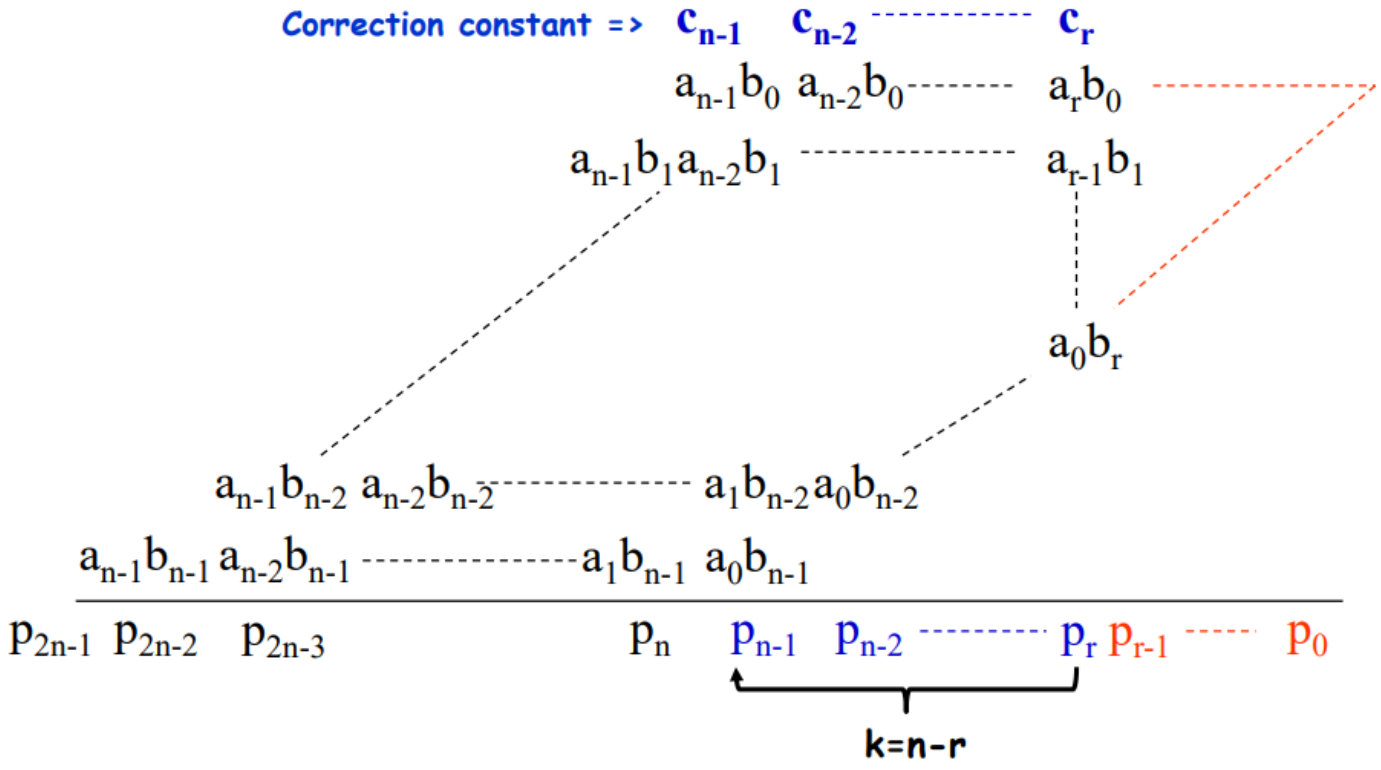
二、trunc_const_row (k=3)

1.設計原理: traucated column index 0~r-1 的項目，最後輸出 n bits。



三、trunc_no_corr (k=3)

1.設計原理: 在 trunc_const_row 的基礎上，加上 correction constant，取對應代號的 correction constant 位元數據。



▲ correction constant :

$$C = -\frac{E_{total}}{2} = (2^k + r - 2) \cdot 2^{r-1} + \frac{1}{2}$$

四、trunc_var_row(k=3)

1.設計原理: 將 traucated 的部分移到剩餘項目的首項，並且在首項到次尾項加 1，進行 row addition。

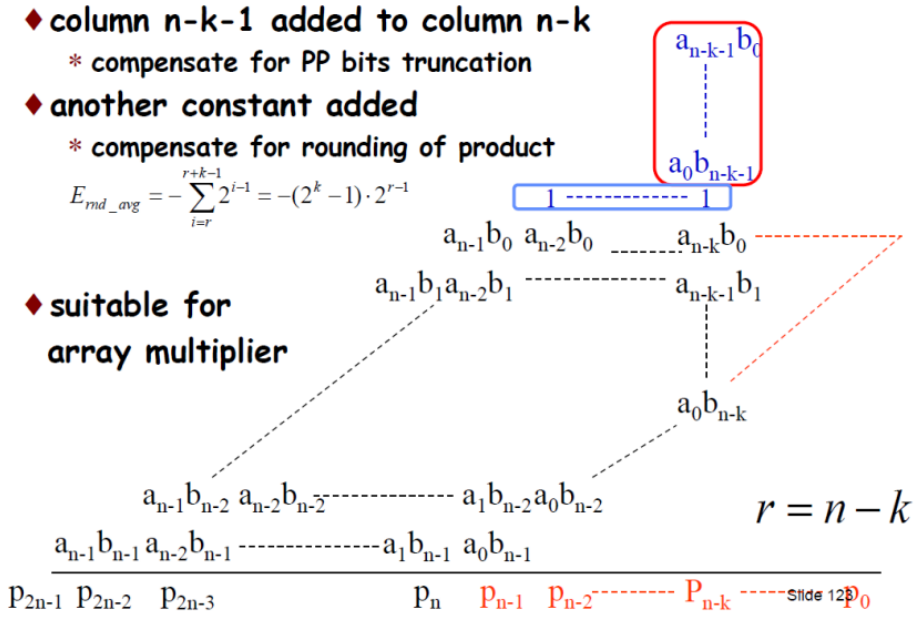
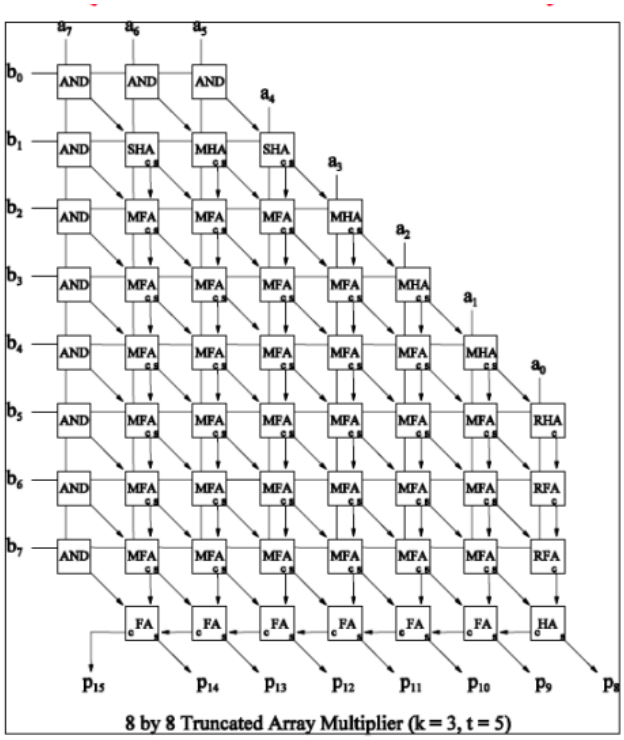


Fig. 2. Truncated multiplier with variable correction.

五、trunc_var_array(k=3)

1.設計原理: 將 variable correction 演算法轉換成 structure 架構。



MFA (modified FA): AND+FA RFA (reduced FA): FA producing carry out only

MHA (modified HA): AND+HA SHA (specialized HA): MFA with one input set to 1

Slide 126

貳.電路合成與計算軟硬誤差

一、 Comparison of Synthesis Results

truncate 8x8		area(um2)	time(ns)	power(w)		
		CL(total)	delay	dynamic	leakage	total
no_trunc	area	256.96	1.60	33.3811 u	2.9556 u	3.6335e-02 m
	mid	407.11	1.10	70.9366 u	4.9459 u	7.5881e-02 m
	delay	892.00	0.62	296.7810 u	17.6717 u	0.3145 m
trunc_const_row	area	670.42	2.00	73.9780 u	6.9171 u	8.0895e-02 m
	mid	858.44	1.50	128.3466 u	9.5692 u	0.1379 m
	delay	1570.36	1.00	387.2937 u	29.2377 u	0.4165 m
trunc_no_corr	area	215.01	1.50	30.0849 u	2.4790 u	3.2564e-02 m
	mid	382.16	1.10	69.2807 u	4.9980 u	7.4279e-02 m
	delay	785.18	0.73	235.1097 u	16.5202 u	0.2516 m
trunc_var_row	area	196.18	1.10	38.0522 u	2.2212 u	4.0273e-02 m
	mid	407.79	0.85	104.5805 u	6.3529 u	0.1109 m
	delay	560.88	0.65	171.6869 u	10.1777 u	0.1819 m
trunc_var_array	area	235.65	2.30	24.4740 u	2.4143 u	2.6888e-02 m
	mid	410.73	1.65	63.8365 u	7.0911 u	7.0928e-02 m
	delay	471.52	1.03	128.5801 u	9.6638 u	0.1382 m

truncate 16x8		area(um2)	time(ns)	power(w)		
		CL(total)	delay	dynamic	leakage	total
no_trunc	area	507.35	2.30	53.9442 u	6.3073 u	6.0250e-02 m
	mid	829.63	1.55	108.3714 u	9.4345 u	0.1178 m
	delay	2033.49	0.82	563.0808 u	40.1699 u	0.6032 m
trunc_const_row	area	641.62	2.00	68.9059 u	5.9383 u	7.4844e-02 m
	mid	803.10	1.50	117.3108 u	8.3803 u	0.1257 m
	delay	1590.09	1.00	388.1834 u	29.6525 u	0.4178 m
trunc_no_corr	area	467.43	2.10	55.0026 u	5.8102 u	6.0811e-02 m
	mid	904.71	1.50	131.3356 u	12.9441 u	0.1443 m
	delay	1685.80	0.86	457.0753 u	32.8562 u	0.4899 m
trunc_var_row	area	207.75	1.40	31.0121 u	2.3719 u	3.3383e-02 m
	mid	384.88	1.10	73.3827 u	5.2319 u	7.8613e-02 m
	delay	686.98	0.75	188.6193 u	13.8003 u	0.2024 m
trunc_var_array	area	x	x	x	x	x

	mid	x	x	x	x	x
	delay	x	x	x	x	x

truncate 16x16		area(um2)	time(ns)	power(w)		
		CL(total)	delay	dynamic	leakage	total
no_trunc	area	1096.58	3.20	91.8556 u	13.3892 u	0.1052 m
	mid	1904.44	2.20	196.9998 u	22.8253 u	0.2198 m
	delay	3987.14	1.20	817.3482 u	77.2156 u	0.8946 m
trunc_const_row	area	643.89	2.20	65.5245 u	6.2133 u	7.1738e-02 m
	mid	765.90	1.60	100.7993 u	7.6041 u	0.1084 m
	delay	1666.53	1.00	385.9305 u	30.1785 u	0.4161 m
trunc_no_corr	area	763.18	2.20	95.7562 u	9.1379 u	0.1049 m
	mid	1635.00	1.65	242.6292 u	24.2223 u	0.2669 m
	delay	2868.57	1.10	667.5333 u	56.8047 u	0.7243 m
trunc_var_row	area	712.15	2.10	90.9260 u	8.4688 u	9.9395e-02 m
	mid	1693.06	1.65	284.0857 u	27.5245 u	0.3116 m
	delay	2369.83	1.23	533.4427 u	48.0964 u	0.5815 m
trunc_var_array	area	x	x	x	x	x
	mid	x	x	x	x	x
	delay	x	x	x	x	x

二、如何驗證硬體算出的值是正確的，並計算軟硬誤差

(1) 使用 ans = a*b 直接產生答案

(2) 比對硬體輸出與 ans 的數值

(3) 計算誤差: 絕對誤差= (outcome – ans) ； 相對誤差= (outcome – ans)/ans

先將 ans 和 outcome 轉成 real type，接著套用誤差公式，同時輸出在 terminal 和寫出檔案。

```
 36 * 129 = sp: 00010010 result:00100100 ==> correct
absolute_error : 8.893182e-323  relative_error : 1.000000e+00
  9 *  99 = sp: 00000011 result:01111011 ==> correct
absolute_error : 5.928788e-322  relative_error : 4.000000e+01
 13 * 141 = sp: 00000111 result:00101001 ==> correct
absolute_error : 1.679823e-322  relative_error : 4.857143e+00
101 *  18 = sp: 00000111 result:00011010 ==> correct
absolute_error : 9.387247e-323  relative_error : 2.714286e+00
  1 *  13 = sp: 00000000 result:00001101 ==> correct
absolute_error : 6.422853e-323  relative_error : inf
118 *  61 = sp: 00011100 result:00011110 ==> correct
absolute_error : 9.881313e-324  relative_error : 7.142857e-02
237 * 140 = sp: 10000001 result:10011100 ==> correct
absolute_error : 1.333977e-322  relative_error : 2.093023e-01
249 * 198 = sp: 11000000 result:10010110 ==> correct
absolute_error : 2.075076e-322  relative_error : 2.187500e-01
197 * 170 = sp: 10000010 result:11010010 ==> correct
absolute_error : 3.952525e-322  relative_error : 6.153846e-01
229 * 119 = sp: 01101010 result:01110011 ==> correct
absolute_error : 4.446591e-323  relative_error : 8.490566e-02
```

(圖) 軟硬比較結果

```
------------------------------

 Allpass!!  #     0/  100

------------------------------
```

(圖) allpass 結果顯示

說明: 利用 verilog 直接產生答案，與電路輸出進行比對，並輸出比對成果。比對的 answer 與 hardware outcome 皆是取 MSB m bits (假設 a * b = m*n = 16*8 bits，則取 16 作為輸出答案 bit 數)。比對結果在+- 1 ulp 範圍內皆判斷正確，使用 real type 計算誤差，套用誤差公式可以得到絕對和相對誤差的數值，並將其輸出，誤差皆 < 0.1，部分會出現 ”inf” or ”nan” 的情況，由於只取 MSB m bits，所以可能出現分母為零的情況。

註: 因為 traucated 演算法只有考慮無號數的情況，所以輸入皆為無號數。

誤差分析比較:

目前誤差落在+- 1ulp，直觀上我認為是進位的問題，在 traucated 不同的演算法中，都是會有誤差。因此，進位誤差應該是 traucated 省略項的誤差，如 correction error and variable correction 雖然有彌補誤差，但仍然存在誤差值，約+- 1 ulp。

no traucated 沒有誤差，trunc_no_corr 沒有補誤差項，出現進位誤差的項目比較多。trunc_var_row 比 trunc_const_row 出現進位誤差的項目較少，相較之下減少了 average 、mean square and maximum error，但 hardware area 都需要比較多。