

Reduced Area and Low Power Square Root Carry Select Adder



Kandur Ashwini¹, Dr. G. Kanaka Durga², P. Koti Lakshmi³

¹M.E Student, India, kandurashwini@gmail.com

²Prof., India, mvsrcgd@yahoo.co.in

³Asst. Prof., India, lakshmi_ponnuri@yahoo.com

Abstract: Arithmetic operations are heart of computational units and data path logic systems. High performance power efficient adders occupying less chip area are necessary in battery powered portable devices. Square Root Carry Select Adder (SQRT CSLA) is one of the fastest adders as compared to all the existing adders. In the existing designs of SQRT CSLA there is possibility of reducing the power and area. In this work modification is carried out at the gate level to significantly reduce the area. The performance of the proposed modified design of SQRT CSLA in terms of delay, area, power and power delay product (PDP) is evaluated using Cadence tools using 0.18 μm CMOS technology. It is compared with the existing SQRT CSLAs for 8-bit, 16-bit, 32-bit and 64-bit on the same platform. Result analysis show that the proposed modified design for 64-bit is 34.28% more power efficient, occupies 19.32% less area than the regular SQRT CSLA with a slight increase in the delay.

Key words: low power, low chip area, SQRT CSLA

INTRODUCTION

Adders are important of DSP processor or data path applications. Design of area and power-efficient adders is one of the most substantial areas of research in VLSI system design. In digital adders, the speed of addition is limited by the time required to propagate the carry. The most basic adder used today is ripple carry adder (RCA). In a RCA the sum for each bit position in an elementary adder is generated sequentially only after the previous bit position has been summed and a carry propagated into the next position. To avoid this problem, a carry select adder (CSLA) [1] is used in many computational systems.

In CSLA the propagation delay is handled by independently generating multiple carries and then selects a carry to generate the sum [2]. However, the CSLA is not area efficient because it uses multiple pairs of RCA to generate partial sum and carry by considering carry input $C_{in} = 0$ and $C_{in} = 1$, then the final sum and carry are selected by the multiplexers (mux) [3]-[4]. In reference [5] SQRT CSLA [6] is used in place of regular CSLA. In SQRT CSLA a Binary to excess-1 converter (BEC) replaces RCA with $C_{in} = 1$ to generate sum and carry [5]. However by

looking at SQRT CSLA with BEC we can see that there is a scope for reducing area therefore in turn reducing the power.

In this paper, modified XOR gate and a modified half adder [7] are used to reduce area and power of the proposed adder. And a small modification is made in the structure of BEC to generate the carry to be propagated to the next stage without the use of mux [8] in the adder. The main advantage of the proposed adder is the reduction in power and area.

The paper is organized as follows: section I is the introduction to the paper. Section II describes the existing architectures of the existing designs which are used later for comparison with the proposed adder architecture. The proposed design is described in detail in Section III. The results are given in section IV and they are compared in section V. the work is concluded in section VI.

EXISTING DESIGNS OF SQRT CSLA

Carry select adder is a high speed adder. It is a compromise between RCA and carry look ahead adder. When compared to RCA, CSLA is faster and when compared to carry look ahead adder hardware complexity is less. The disadvantage of CSLA is its area. Among different types of CSLA, SQRT CSLA has balanced delay, less power required then linear CSLA [9].

First a regular (linear) SQRT CSLA is shown in Fig.1. It consists of two sets of RCA. The block size of RCA is designed to optimally match the signal arrival time at the final multiplexer input and the delay time of carry-in signal. This can be achieved by progressively adding more bits to the subsequent stages in the RCA, requiring more time for the generation of the carry signals. For example, the first stage of RCA adds 2 bits of input, the second stage adds 3bits, the third adds 4 bits, and so forth. For a 16-bit regular SQRT CSLA adder it is divided into 5 stages accordingly (Fig in ref [5]). Same inputs are given to both the sets of RCA with a difference of the input carry. The Carry input (C_{in}) to one set of RCA is '0' and for other set is '1' and accordingly the output is calculated. The selection of the correct output is done with the help of multiplexers. And the control signal for the multiplexer is C_{in} of the previous stage.

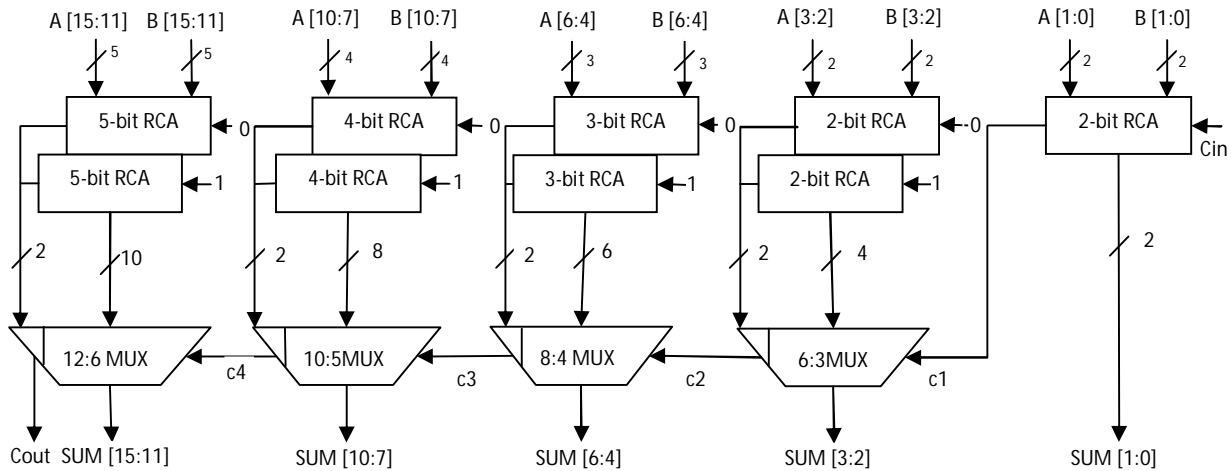


Fig.1 16-bit Sqrt CSLA

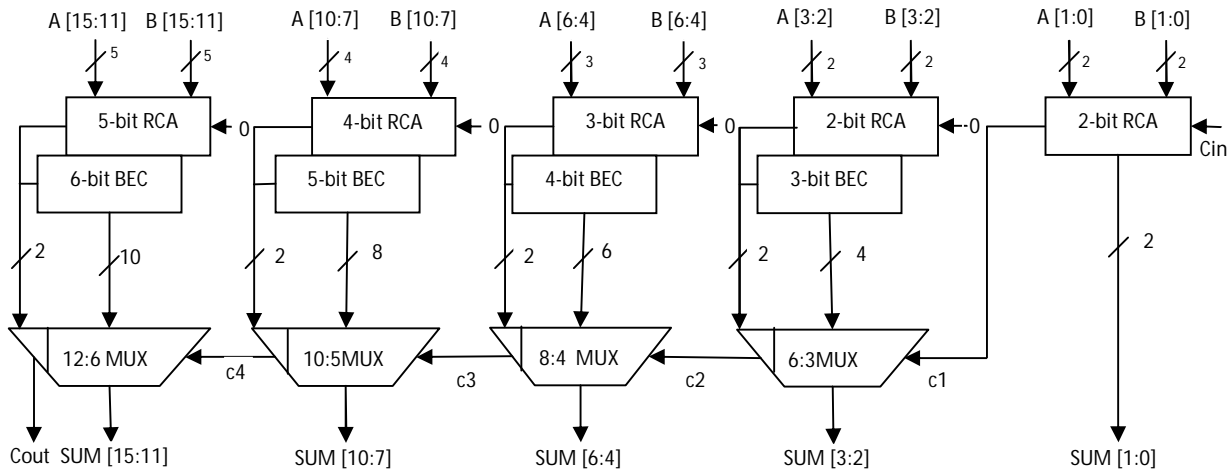


Fig.2 16-bit Sqrt CSLA with BEC

But it is clear from the Fig that there is a scope to reduce area. So instead of using 2 RCAs one was replaced by a Binary to Excess-1 Converter (BEC) [5].

Sqrt CSLA with BEC is designed with one set of RCA and BEC as shown in Fig.2. In this design n-bit RCA with $C_{in} = 0$ is used to generate sum and carry while n+1 bit BEC is used to generate sum and carry for $C_{in} = 1$. And the correct output is selected with the help of mux. It is clear that Sqrt CSLA with BEC is better than Sqrt CSLA, but the area can be further optimized thus reducing the power as described in the next section.

PROPOSED DESIGN OF Sqrt CSLA

The proposed Sqrt CSLA is developed with the help of modified half adder (HAM), modified full adder (FAM) and

modified XOR gate (XORM). In place of BEC, combinational logic block (CLB) is used. XORM has 1 gate less than the conventional XOR gate of 5 gates (AND-OR-NOT implementation) [8] as in Fig.3.

HAM has 2 gates less than the conventional half adder as shown in Fig.4. The full adder is constructed with two HAMs and an AND gate [8] shown in Fig.5 has only 9 gates, 4 gates less than conventional full adder. As the number of gates reduce in the basic building blocks of the proposed Sqrt CSLA area is also reduced.

A part from the above modifications, one more small change is made in the design which allows us to get the carryout of the group without using the mux [7].

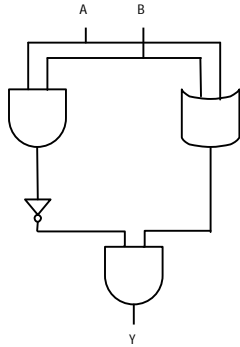


Fig.3. Modified XOR gate (XORM)

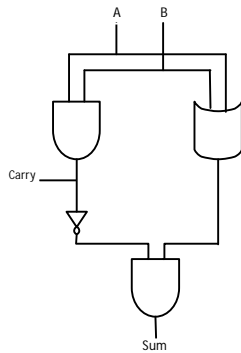


Fig.4. Modified half adder (HAM)

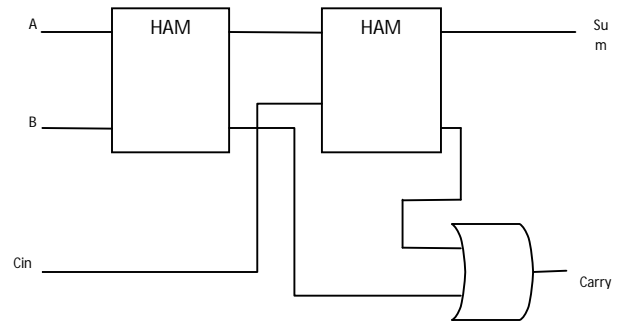


Fig.5 Modified full adder (FAM) using two HAMs

The proposed design Sqrt CSLA with CLB is as shown in Fig.6.

A 16-bit model is presented which is divided into 5 groups. Each group consists of different size of RCAs, CLBs and multiplexers. The CLB is used instead of RCA for $C_{in} = 1$. The structure of a 4-bit CLB is given in Fig.7. The sizes of RCA differ from 2-bit in first two groups and consequently increase to 5-bit in group 5. Similarly the size of CLB also increases from 3-bit in group 2 to 6-bit in group 5. The groups in the proposed adder are shown in Fig.8.

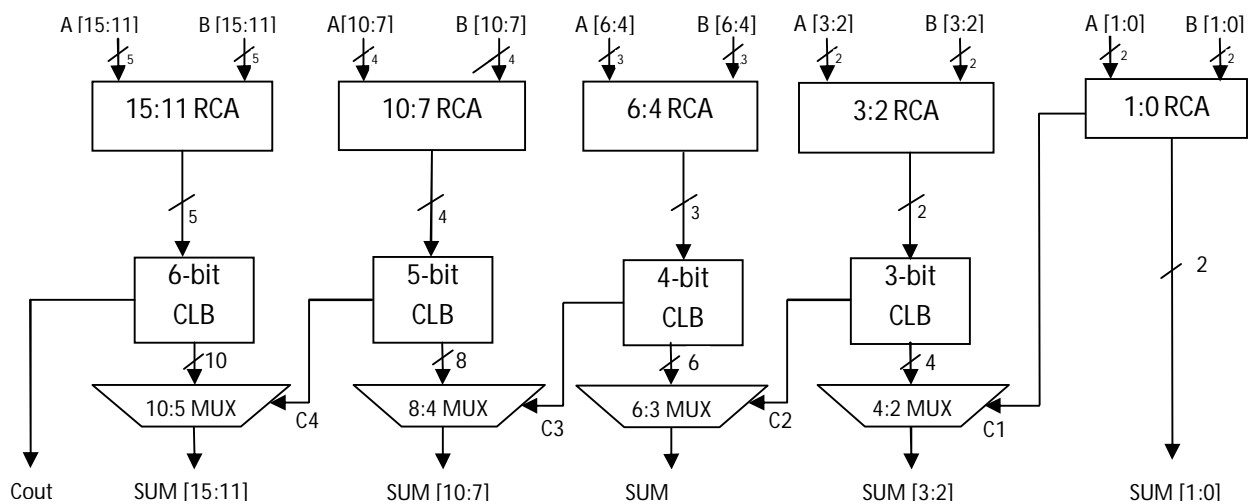


Fig.6. 16-bit Sqrt CSLA with CLB

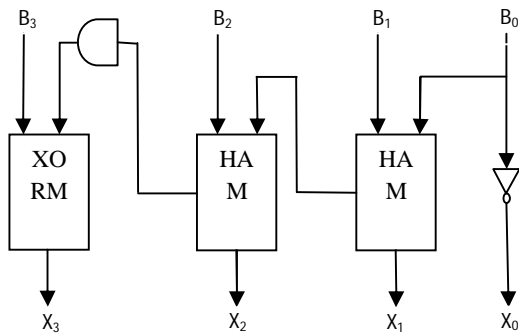
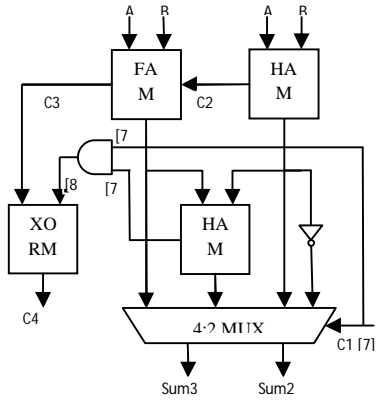
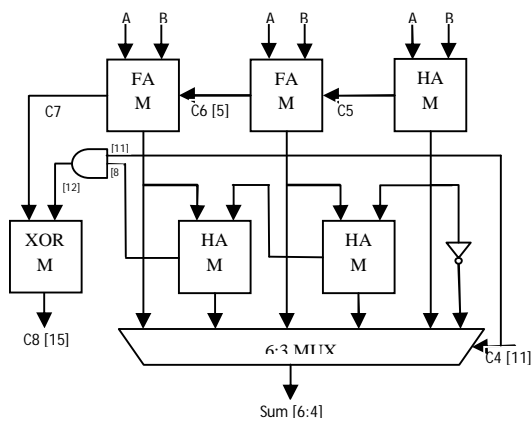


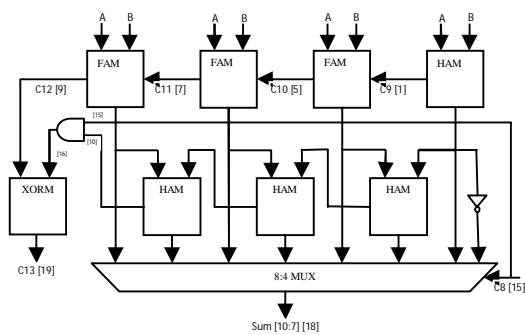
Fig7. 4-bit CLB



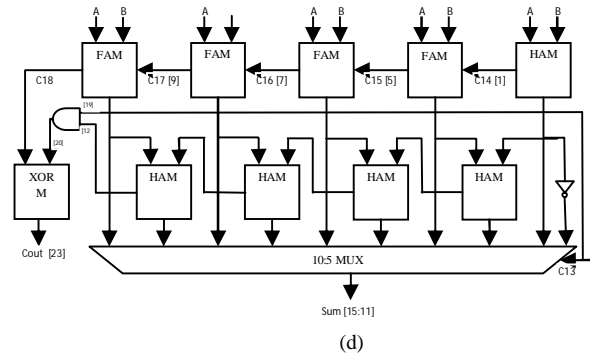
(a)



(b)



(c)



(d)

Fig.8. (a) group 2, (b) group3, (c) group4 and (d) group5 of Sqrt CSLA with CLB

Group 2 of the proposed adder consists of 2-bit RCA, 3-bit CLB and 4:2 mux. RCA is constructed using one FAM and one HAM. The CLB has one NOT gate, one HAM, one AND gate and a XORM. For the remaining groups the size of RCA and CLB is listed in Table I.

Table1. Gate Count of Sqrt CSLA with CLB

Groups	RCA		Combinational logic block (CLB)			Total no. of gates
	FAM	HAM	NOT gate +AND gate	HAM	XORM	
Group1	2(*9)	0	0	0	0	18
Group 2	1(*9)	1(*4)	1+1	1(*4)	1(*4)	31
Group 3	2(*9)	1(*4)	1+1	2(*4)	1(*4)	48
Group 4	3(*9)	1(*4)	1+1	3(*4)	1(*4)	59
Group 5	4(*9)	1(*4)	1+1	4(*4)	1(*4)	82

The numerical in the brackets () indicates the number of gates required to built the module.

IMPLEMENTATION AND RESULTS

The performance evaluation of existing adder designs (Sqrt CSLA and Sqrt CSLA with BEC) and the proposed adder (Sqrt CSLA with CLB) is carried out using Cadence Encounter RTL compiler tool [9] in 180 nm technology. The adders are compared for word size of 8, 16, 32 and 64 bit, the number of gates required by the adder designs for 16-bit is provided in Table II. In the proposed design the total number gates required for 16-bit architecture is 238 which is 198 gates less than Sqrt CSLA and reduction of 98 gates when compared to Sqrt CSLA with BEC.

Table2. Gate count of different Sqrt CSLA

Group	Number of gates		
	Sqrt CSLA	Sqrt CSLA (BEC)	Sqrt CSLA (CLB)
Group 1	26	26	18
Group 2	57	43	31
Group 3	87	66	48
Group 4	117	89	59
Group 5	147	112	82
Total	434	336	238

Power, delay and area reports were generated after synthesizing the architectures of the adders for 8, 16, 32 and 64 bit. The power in Table 3 indicates the total power which is the sum of dynamic and static power. The area indicates the total cell area of the design and the delay indicates the propagation delay.

The power-delay product and area-delay product for each design are calculated and provided in table III.

Table3. Comparison of Sqrt CSLA, Sqrt CSLA with BEC and Sqrt CSLA with CLB

Word size	Adders	Delay (ns)	Area (μm^2)	Total Power (μW)	Power-delay product (10^{-15}) (J)	Area-delay product (10^{-15})
8-bit	Sqrt CSLA	1.68	1035	76.687	129.2954	1745.01
	Sqrt CSLA with BEC	1.82	888	66.735	121.4579	1616.16
	Sqrt CSLA with CLB	1.85	845	53.186	98.394	1563.25
16-bit	Sqrt CSLA	2.903	2259	171.797	498.729	6557.87
	Sqrt CSLA with BEC	2.858	1873	139.209	397.859	5353.03
	Sqrt CSLA with CLB	3.524	1866	116.850	411.779	6609.37
32-bit	Sqrt CSLA	4.310	4787	390.199	1681.758	20631.97
	Sqrt CSLA with BEC	4.405	3922	307.931	1356.438	17276.41
	Sqrt CSLA with CLB	5.206	3895	253.098	1317.628	20277.37
64-bit	Sqrt CSLA	7.119	9883	829.125	5902.543	70357.08
	Sqrt CSLA with BEC	6.970	8007	628.192	4378.499	55808.79
	Sqrt CSLA with CLB	7.978	7973	545.083	4348.621	63608.59

From the table III it is clear that the total power consumed by Sqrt CSLA with CLB is reduced by 30.64%, 31.98%, 35.13% and 34.25% in 8-b, 16-b, 32-b and 64-bit respectively. Interestingly the area is also uniformly reduced by 18%, 17%, 18% and 19% in 8-b, 16-b, 32-b and 64-b respectively. But there is a slight increase in the delay. Interestingly the power delay product is also reduced by 19%, 27%, 31% and 35% for bits as mentioned above respectively. Also the area delay product is lowered by 23%, 17%, 21% and 26% for 8-b, 16-b, 32-b and 64-b respectively for the Sqrt CSLA with CLB. The percentage reduction in delay, area and power are plotted in Fig.11 (a) and the graph of reduction in Power-delay product and Area-delay product is shown in Fig.11 (b).

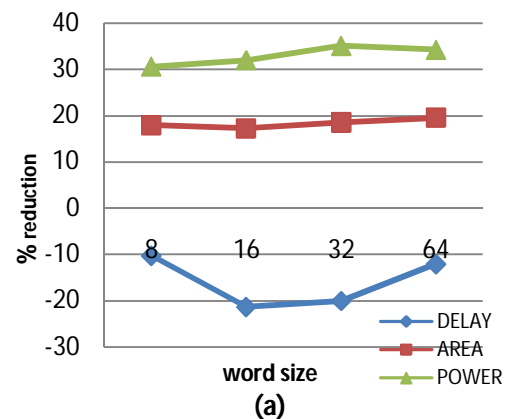


Fig.11 (a) Percentage reduction in delay, area and power of Sqrt CSLA with CLB

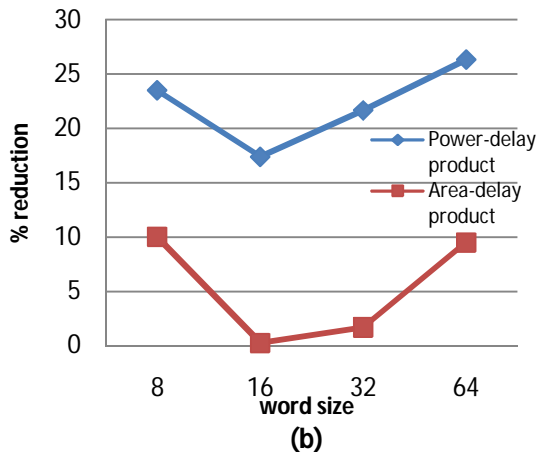


Fig.11 (b) Percentage reduction in PDP and ADP

CONCLUSION

In this paper a modified Sqrt CSLA with CLB is proposed. The proposed architecture requires less number of gates when compared to Sqrt CLSA and Sqrt CSLA with BEC. This reduction leads to the reduction in area, power with a slight increase in delay. The results show that for 64-bit Sqrt CSLA with CLB with a slight increase in delay of 10% there is a reduction in area by 19% and also the total power by 34%. Hence the values of all the parameters are significantly reduced. There is a significant reduction in power delay product and also area delay product for all the bits in the proposed design, which indicates the success of the method and not the mere tradeoff of delay, power and area. Hence the proposed adder is faster and, power and area efficient. The adder can be tested for higher word length.

REFERENCES

- [1] O. J. Bedrij, "Carry-select adder," *IRE Trans. Electron. Comput.*, pp.340–344, 1962.
- [2] Y. Kim and L.-S. Kim, "64-bit carry-select adder with reduced area," *Electron. Lett.*, vol. 37, no. 10, pp. 614–615, May 2001
- [3] B. Ramkumar, H.M. Kittur, and P. M. Kannan, "ASIC implementation of modified faster carry save adder," *Eur. J. Sci. Res.*, vol. 42, no. 1, pp. 53–58, 2010.
- [4] T. Y. Ceiang and M. J. Hsiao, "Carry-select adder using single ripple carry adder," *Electron Lett.*, vol. 34, no. 22, pp. 2101–2103, Oct. 1998.
- [5] B. Ramkumar and Harish M Kittur "Low-Power and Area-Efficient Carry Select Adder" *IEEE Trans. on very large scale integration systems* 2012.
- [6] Y. He, C. H. Chang, and J. Gu, "An area efficient 64-bit square root carry-select adder for lowpower applications," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2005, vol. 4, pp. 4082–4085.
- [7] Neelima.V, Dr. G. Kanaka Durga "Area Efficient Carry Select Adder With Modified Logic Blocks" National Conference on Emerging Trends in Electronics & Communication, Feb 2014.
- [8] Jucemar Monteiro, José Luís Güntzel, Luciano Agostini" A1CSA: An Energy-Efficient Fast Adder Architecture for Cell-Based VLSI Design" the Brazilian Council for Scientific and Technological Development (CNPq). ©2011
- [3] M. Basak, M. Sutradhar, B. Santra, M. Saha, D. Chowdhury, J. Samanta "STUDY THE PERFORMANCE ANALYSIS OF LOW POWER-HIGH SPEED CARRY SELECT ADDER " *International Journal of VLSI and Embedded Systems-IJVES*, Vol 04, Article 06102; June 2013
- [9] Cadence, "Encounter user guide," Version 6.2.4, March 2008