Power Efficient Weighted Modulo 2^n+1 Adder

C.Venkataiah[#1] C.Vijaya Bharathi[*2] M.Narasimhulu[#3]
[#]*Assistant Professor, Dept. of Electronics & Communication Eng, RGMCET, Nandyal, Kurnool (dist), Andhra Pradesh, India.*
[*]*M.Tech (ES) student, RGMCET, Nandyal, Kurnool (dist), Andhra Pradesh, India.*

Abstract- The comparison of three different architectures for modulo 2^n+1 adders are introduced in this paper. The first two architecture can be implemented different power consumptions, while maintain the same delay. The partitioned Sklansky structure compared to previous architecture can be implemented less power consumptions, while maintain the different delay and gate counts. The modulo adder 2^n+1 adders can be easily derived by adding extra logic of modulo 2^n-1 adder. Power efficient modulo 2^n+1 adders are appreciated in a variety of computer applications such as cryptography,RNS.The modulo 2^n+1 adder is synthesized using Xilinx 9.1i tool and implemented FPGA spartan2 kit.
Keywords- Sklansky-style parallel prefix adder, kogge-stone parallel prefix adder, FPGA Spartan 2 kit ,VLSI.

I. INTRODUCTION

The residue number system is a non-weighted number system which speeds up arithmetic operations by dividing them into smaller parallel operations. Since the arithmetic operations in each modulo are independent of each other, there is no carry propagation among them so residue number system is carry-free addition, multiplication and borrow-free subtraction. Residue number system is one of the most effective techniques for power dissipation reduction in VLSI system design .Some application of the residue number system are digital signal processing. Digital filters [6].

The modulo 2^n+1 arithmetic unit complexity is determined by chosen for the operands representation. Three representations are considered namely, the normal weighted-one, diminished-one and the redundant representation. In above only we consider the first two representations, Since the redundant representation of modulo 2^n+1 additions demand significantly more area than those of diminished-1 and weighted representations.

In the normal-weighted representation, each operand requires n+ 1 bit for its representation but only utilizes 2^n+1 representation out of the 2^n+1 that these can provide. A denser encoding of the input operands and simplified arithmetic operations modulo 2^n+1 are offered by the diminished-1 representation. The range of inputs for weighted representations is larger than that of the diminished-1 case (i.e., {0, 2^n} vs. {0, 2^n-1}). Besides, the zero-detection hardware is not required for the weighted representations, and unlike the other, it does not involve any area-overhead for the translation from/to the binary weighted system. In this paper, we therefore, focus on the design of an efficient weighted modulo 2^n+1 adder and to compare the different parallel prefix structures like sklansky –style,kogge-stone for n=8 and 2x4 partitioned parallel prefix control units.

II.PARALLEL PREFIX ADDITION BASICS

Generally parallel-prefix n-bit adder considered as a three stage circuit. They are pre-processing-stage, carry-computation-unit and post-processing-stage. Suppose that $A=A_{n-1}\,A_{n-2}\ldots\ldots A_0$ and $B = B_{n-1}\,B_{n-2}\ldots B_0$ represent the two numbers to be added and $S = S_{n-1}\,S_{n-2}\ldots S_0$ denotes their sum

.

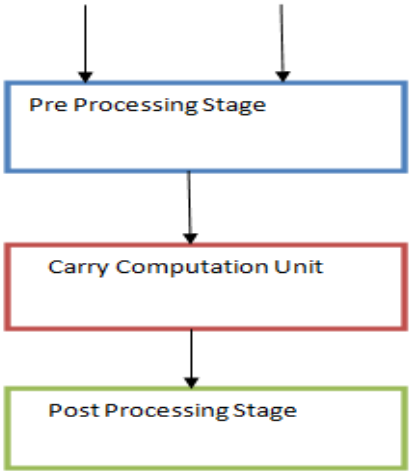


Fig .1: Parallel Prefix Addition Basics

A). Pre Processing Stage

The preprocessing stage computes three type of signal bits. They are carry-generate bits G_i, the carry-propagate bits P_i, and the half-sum bits H_i, for every I, $0 \le i \le n-1$, according to

$G_i = A_i \cdot B_i$ $P_i = A_i + B_i$ $H_i = A_i \oplus B_i$

Where ·, +, ⊕ denote the logical AND, OR, and EXCLUSIVE-OR, respectively. The pre-processing-stage is shown in the figure.2.

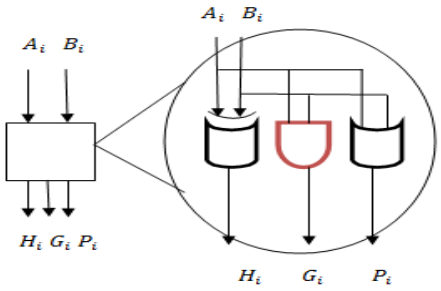


Fig.2: Pre-Processing-Stage

B). Carry Computation Unit

The second stage of the adder, here after called the carry computation unit, computes the carry signals C_i, for $0 \leq i \leq n-1$ using the carry generate and carry propagate bits Gi and Pi. Carry computation transformed into a parallel prefix problem using the ◦ operator, which associate pairs of generate and propagate signals and defined as

(G, P) ◦ (G', P') = (G + P . G', P·P')

In a serious of associations of consecutive generate/propagate pairs (G, P) , the notation ($G_{k:j}$, $P_{k:j}$) with k>j, used to denote the group generate/propagate term produced out of bits k, k-1, . . .j, that is,

($G_{k:j}$, $P_{k:j}$) = (G_k, P_k) ◦ (G_{k-1}, P_{k-1}) ◦ . . . ◦ (G_j, P_j)

Since every carry $C_i = G_{i:0}$, a number of algorithms have been introduced for computing all the carries using only ◦ operator. The prefix operator is shown in the fig.3.

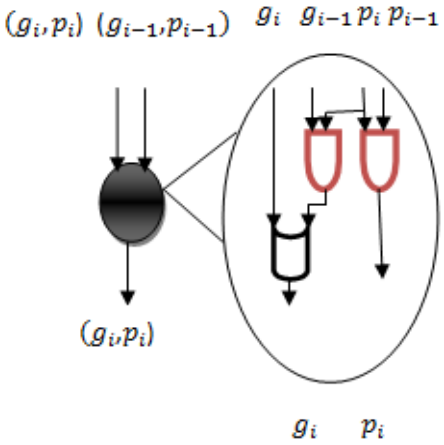


Fig.3: Carry Computation Unit

C). Post Processing Unit

The third computes the half sum bits according to Si = Hi ⊕Ci-1.The post processing stage is shown in the figure.4.

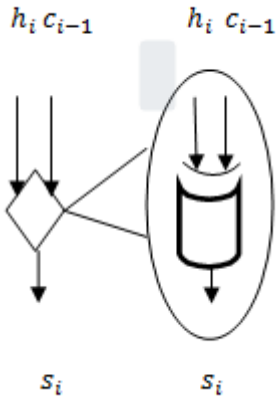


Fig.4:Post Processing Unit

III. REVIEW OF WEIGHTED MODULO 2^n+1 ADDER USING SKLANSKY-STYLE STRUCTURE

In this adder for (n+1) bit 2 inputs A and B where $0 \leq A, B \leq 2^n$ the weighted modulo sum 2^n+1 is represented as

$$\left||A+B|_{2^n+1}\right|_{2^n} = |A+B-(2^n+1)|_{2^n} = |Y' + U'|_{2^n} + \overline{C_{n-1}VFIX}$$

Here Y′ and U′ are the carry and sum vectors of the summation of A,B and-(2^n+1), where Y′=$y'_{n-2}y'_{n-3}$.....$y'_0\overline{y'n-1}$ and U′=$u'_{n-1}u'_{n-2}$…..u'_0 and $FIX = a_n b_n V b_n a_{n-1} V a_n b_{n-1}$ respectively, where *FIX* and c_{n-1} represent the correction and the end around-carry signals, respectively

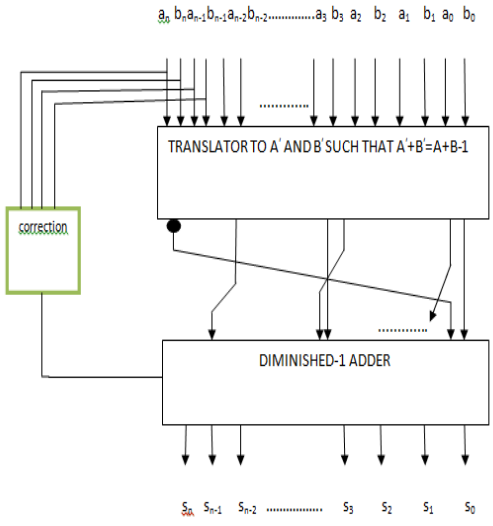


Fig.5:The Architecture of Weighted modulo2^n+1 adder using simple correction unit

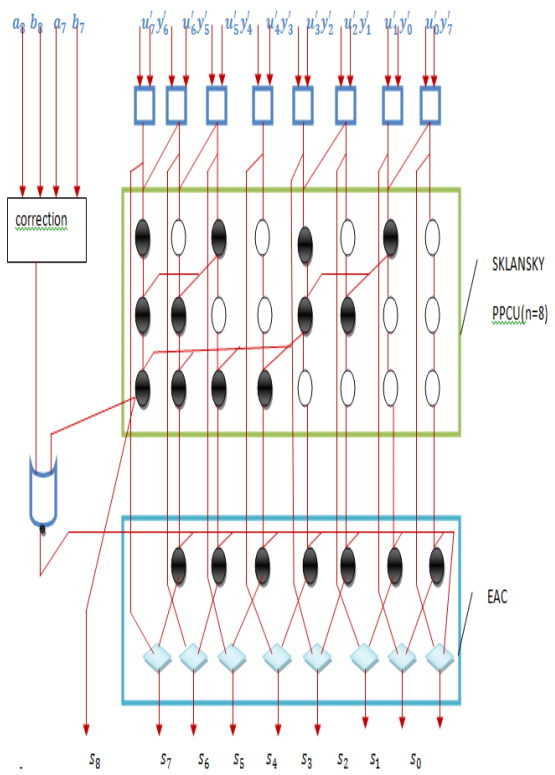


Fig. 6: The diminished-one adder based on Sklansky-style parallel-prefix structure with the correction circuits for weighted modulo 2^8+1 adder

The above fig.6 white represent prefix operator. A square represents the logic that produces bit-level carry propagate and generate signals. A black represent associative that produces both block carry-propagate and carry-generate signals and diamond represents sum- formation logic.

IV. REVIEW OF WEIGHTED MODULO 2^n+1 ADDERUSING KOGGE-STONE STRUCTURE

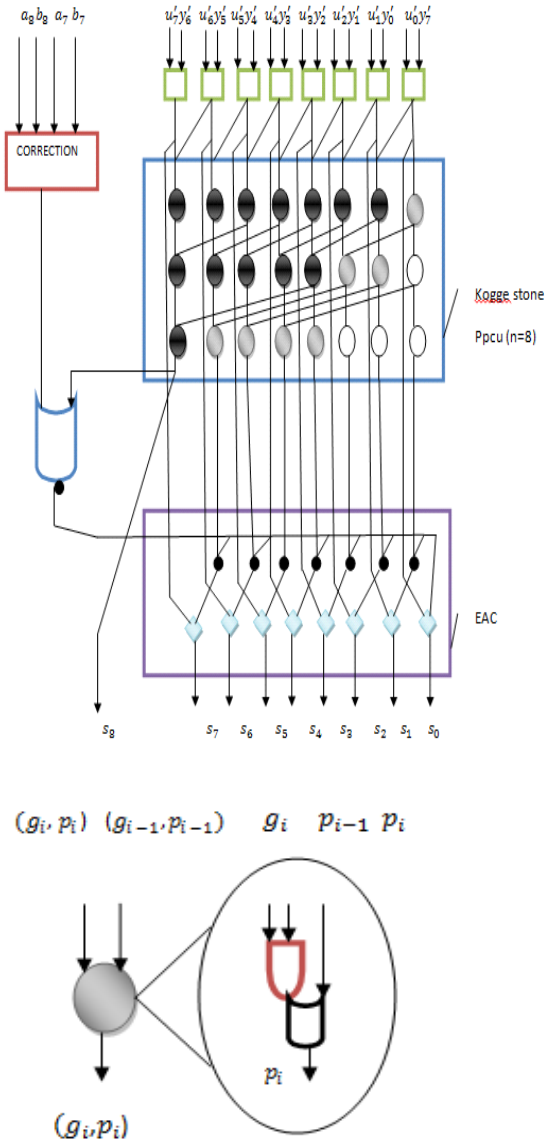


Fig 7(a): The diminished-one adder based on Kogge-stone parallel-prefix structure with the correction circuits for weighted modulo 2^8+1.(b) Gray cell

In the above fig.7(a).black represent associative that produces both block carry-propagate and carry-generate signals and A gray cell that produces the carry-generate signal without carry-propagate signal.

V.EFFICIENT MODULO ADDER USING SKLANSKY PARTITIONED PREFIX CONTROL UNIT WITH ENHANCED CIRCULAR CARRY GENERATION CIRCUIT

- Here n-bit Parallel Prefix Computational Units (PPCU) are replaced by m blocks of r-bit PPCU's.For efficient computation of the output carries Enhanced Circular Carry Generation scheme is used.

- Proposed method can reduce the area and time complexities since it requires less carry nodes compared to that of the existing single block parallel-prefix computation units.

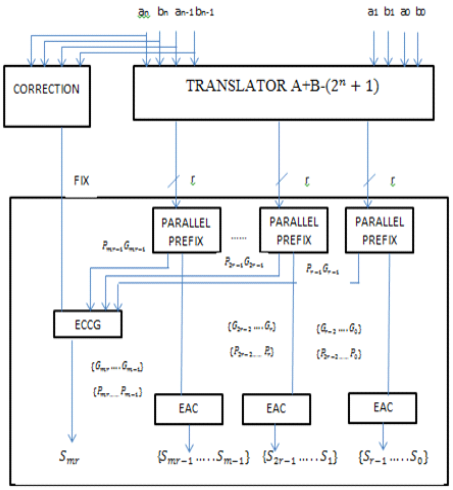


Fig.8: Architecture Of Modulo 2^n+1Adder Using Enhanced Circular Carry Generation Unit.

- In the above architecture a 'n' bit PPCU is partitioned in into *m* blocks of small PPCUs where each block is of r bits Where n = r × m bits.
- Here each block is used to produce an. i.e. propagation p_i and generate g_i signals.
- Here the square node computes the generate and propagate signals g_i and p_i, respectively, for Y' and U' as shown $g_i=y'_{i-1}u'_i$, $p_i=y'_{i-1}\oplus u'$ for i=1, 2, 3….n-1.
  For i=0 $g_0=\overline{y'n-1}\,u'_0$, $y_0=\overline{y'n-1}\oplus u'_0$.
- From the black circle the carry out signal c_{n-1} is obtained as

$$c_{n-1}=g_{n-1}+\sum_{j=0}^{n-2}(\prod_{k-j+1}^{n-1}p_k)g_j$$

- The final modulo sum s_i(i=0 to n-1)
  $s_i=(g_{i-1}\sum_{j=0}^{n-1}(p_k)g_j+\overline{c_{n-1}+FIX}\prod_{k=0}^{i-1}p_K)$
- For i=n, s_n is directly computed from the carry-out of the parallel-prefix Computation $s_n=\overline{FIX}p_{n-1}$.

A). End around carry unit

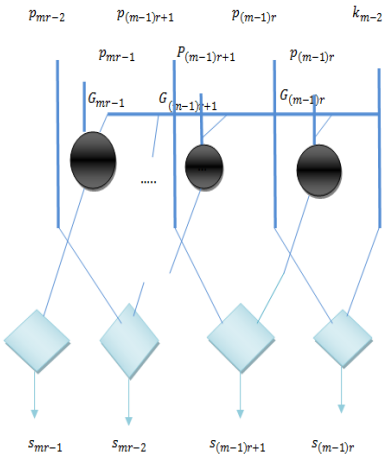


Fig.9: End Around Carry

In the end around carry group the generate and propagate signals can be obtained as follow. For t=0 to m-1

$$G_t^*=g_{tr+(r-1)}+\sum_{j=tr}^{tr+(r-2)}(\prod_{k=j+1}^{tr+(r-1)}p_k)g_j$$

$$P_t^*=\prod_{k=tr}^{tr+(r-1)}p_k$$

B).Enhanced circular carry generation unit

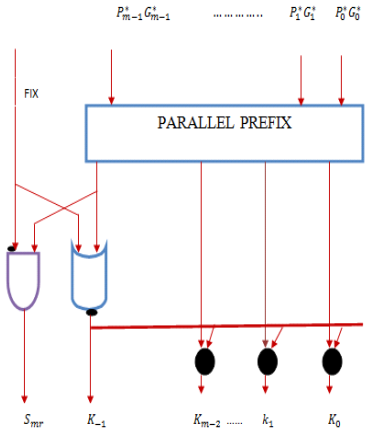


Fig.10: Enhanced Circular Carry Generation Unit

- Here internal carries are generated and also carry out to produce the final corrected modulo sum So the internal carry k_{t-1} and carry-out signal is given

$$k_{t-1}=G_{t-1}^*+\sum_{j=0}^{t-2}(\prod_{t=j+1}^{t-1}p_t^*)G_j^*+\overline{c_{n-1}FIX}\prod_{t=0}^{t-1}p_t^*$$

$$c_{n-1}=G_{m-1}^*+\sum_{j=0}^{m-2}(\prod_{l=j+1}^{m-1}p_t^*)\,G_j^*$$

- The final modulo sum is

$$s_i = \begin{cases} k_{[i/r]-1} \oplus p_i & if\ (i) mod(r) = 0 \\ (G_i + P_i \bullet k_{[i/r]-1}) \oplus p_i & otherwise \end{cases}$$

From the above figure.10 S_{mr} is directly computed.

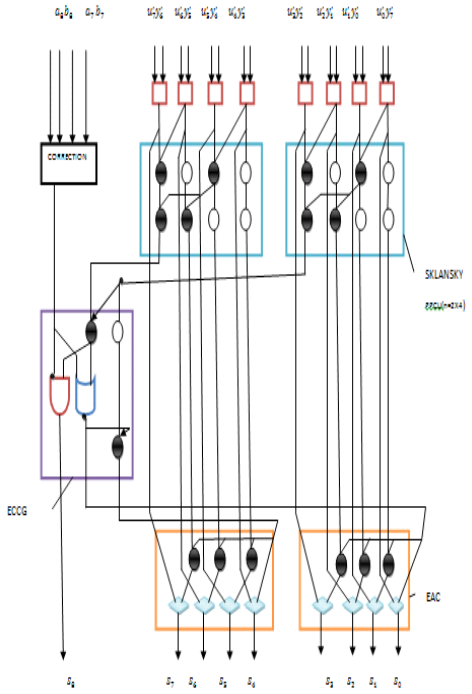


Fig.11: Modulo adder with Sklansky-style 2x4 partitioned blocks of Parallel Prefix Control Unit

Two examples for our proposed addition methods are given as follows.

Example 1: Suppose n=4, $A=16_{10}=10000_2$, and $B=15_{10}=01111_2$,respectively.

Step 1) $(A+B)-(2^n+1) \Rightarrow Y'=1100_2$, $U'=0000_2$, FIX=1.

Step 2) $Y'+U'=1110_2$,$C_{Out}=0$, $\Rightarrow Y'+U'+\overline{C_{out}} \vee FIX = 1110_2=|16+15|_{17}=14_{10}$.

Example 2: Suppose n=4, $A=11_{10}=01011_2$, and $B=5_{10}=00101_2$,respectively.

Step 1) $(A+B)-(2^n+1) \Rightarrow Y'=1110_2$, $U'=0001_2$, FIX=0.

Step 2) $Y'+U'=1111_2$,$C_{Out}=0$, $\Rightarrow Y'+U'+\overline{C_{out}} \vee FIX = 10000_2=|11+5|_{17}=16_{10}$.

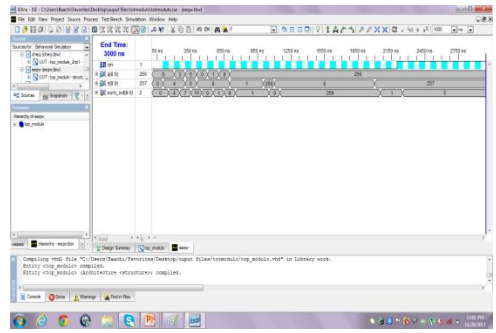
VI. SYNTHESIS RESULT

we have VHDL coded and Synthesized the Modulo 2^n+1 Sklansky style and Kogge-stone as well as Sklansky partitioned prefix control unit using Xilinx tool to reduced the power consumption.

Table.1: Output Results Using Xilinx tool

PARAMETERS	Modulo adder using Sklansky-style(n=8)	Modulo adder using Kogge-stone(n=8)	Modulo adder using Sklansky partitioned prefix(n=8)
Delay(ns)	13.160	13.513	20.234
Power(mW)	24	18	7
Gate count	129	207	381

Table.2:Comparision of Delay,Gate count and Power Kogge-stone and different Sklansky structures



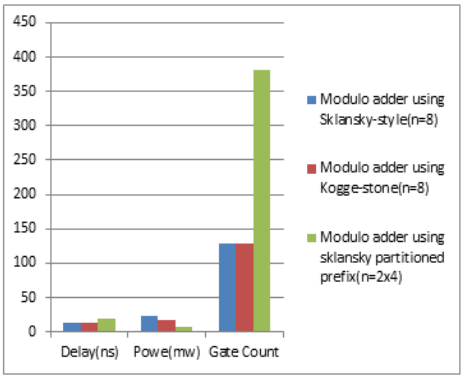


Fig.12:Design of Modulo 2^8+1 Using Different structures

VII. CONCLUSION

The comparison of three different architectures for modulo 2^n+1 adders are introduced in this paper. The first two architecture can be implemented different power consumptions, while maintain the same delay. The partitioned Sklansky structure compared to previous architecture can be implemented less power consumptions, while maintain the different delay and gate counts. The

modulo adder 2^n+1 adders can be easily derived by adding extra logic of modulo 2^n-1 adder. Power efficient modulo 2^n+1 adders are appreciated in a variety of computer applications such as cryptography, RNS.The modulo 2^n+1 adder is synthesized using Xilinx 9.1i tool and implemented in FPGA Spartan 2 kit .

REFERENCES

[1]. Efficient Weighted Modulo +1 Adders by Partitioned Parallel-Prefix Computation and Enhanced Circular Carry Generation Tso-Bing Juang*, Member, IEEE, Pramod Kumar Meher**, Senior Member, IEEE, and Chin-Chieh Chiu*, 2011.

[2]. T. –B. Juang, C. –C. Chiu and M. –Y. Tsai, "Improved area-efficient weighted modulo +1 adders design with simple correction schemes,"*IEEE Transactions on Circuits and Systems II, Exp. Briefs* Vol. 57, No.3, pp. 198-202, March 2010.

[3]. H. T. Vergos and C. Efstathiou, "A unifying approach for weighted anddiminished-1 modulo +1 addition," *IEEE Transactions on Circuits and Systems II, Exp. Briefs*, Vol. 55, No. 10 , pp. 1041-1045, Oct. 2008.

[4]. H. T. Vergos and D. Bakalis, ''on the use of diminished-1 adders for weighted modulo +1 arithmetic components,'' Proc. 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, pp. 752-759, Sept. 2008.

[5]. Design and Characterization of Parallel Prefix Adders using FPGAs David H. K. Hoe, Chris Martinez and Sri Jyothsna Vundavalli Department of Electrical Engineering the University of Texas, Tyler @2011 IEEE.

[6]. Nannarell, M Re, and G. C. Cardarilli, "Tradeoffs between residue number system and traditional FIR filters," Proc. of the IEEE International Symposium on Circuits and Systems (ISCAS), pp. 305-308, May 2001.

[7]. K. Kaluri, W. F. Leong, K. –H. Tan, L. Johnson, and M. Soderstrand,"FPGA hardware implementation of an RNS FIR digital filter, "Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers, pp. 1340-1344, Nov. 2001.

[8]. M.Parimaladevi R.Karthi "Analysis of Power Efficient Modulo 2^n+1 Adder Architectures" *International Journal of Computer Applications (0975 – 8887) Volume 70– No.4, May 2013.*