**Sklansky and Kogge Stone Adder Digital Filter Design and Implementation**

**Kalikivayi Ajay Kumar1,Mr.Ch. Arun Prakash2, Doopam Mahesh Babu3, Kota Thirumala Eswar kumar4, Nulaka Sudheer Kumar Reddy5**

**2Assistant Professor and 1,3,4&5IV-B.Tech(ECE)**

**Department of Electronics and Communication Engineering**

**RISE Krishna Sai Prakasam Group of Institutions,Ongole,AP,India**

 **(****1ajaykumarkalikivai@gmail.com;2 arungvpe@gmail.com**

 **;** **3maheshdoopam21@gmail.com****;** **4eswarkuma308@gmail.com****;** **5nulakasudheerkumarreddy@gmail.com** **)**

**Abstract:**Arithmetic and logic unit has been the most significant unit in any electronic devices. In the recent advancement, for an arithmetic and logic unit to be significant it needs to have an efficient algorithmic operation such as Multiplications and addition. Multiplication is the most essential operation in digital signal processing, artificial intelligence, neural networks, and machine learning. In Digital era, filters can be used as memory elements to store data. The most commonly used adders to implement digital filters, are the full adders. Here in this project the implementation of digital filter is done with PPA. Among various PPA, the sklansky adder is more efficient than others. When implementing kogge stone adder, it experiences a larger hardware complexity. So that it is necessary to reduce the hardware complexity the architecture of sklansky adder. The comparative analysis made among various parameters such as numbers of logic gates and delay. The above proposed architecture tends to reduce the hardware complexity using various parameter.

**KEyWORDS-**Sklansky,Kogge Stone, Fir Filter.

**1 INTRODUCTION**

Generally there are some basic processes to design a digitalfilter. The design filter needs to be redesign the frequencies,calculate the parameters and have to adjust the filter at eachtime. In some rare cases, the redesign requires filters to beexchanged with their filter types or with its length to fitaccording to their requirements.

A complicated DSP system consists of multiple adders andmultipliers[2]. The effective design of DSP machine enchancethe performance of the system. An adder, which is afundamental component is frequently employed in manynetworks that are used in system like controllers andprocessing chips[3]. In this system a performance is improvedby the running ability of adder and multiplier.

In Digital filters can use FIR or IIR filter systems forimplement and there adaptive algorithm to update theirparameters but in practice, FIR structured digital filters can beeffortlessly done with high performance so that it has beingmore widely used. We also use FIR digital filter for ourstudies[5]. Normally FIR filter will compute the sum ofinput signal by analyzing the different types of multipleand accumulate performance.

It is common in DSP to say that a filter input and output signals are in time domain. This is because signals are usually created by sampling at regular intervals of time. But this is not the only way sampling can take place. The second most common way of sampling is at equal intervals in space. For example imagine taking simultaneous readings from an array of strain sensors mounted at one centimeter increments along the length of an aircraft wing. Many other domains are possible; however, time and space are by far the most common. When you see the term time domain in DSP, remember that it may actually refer to samples taken over time, or it may be a general reference to any domain that the samples are taken in. Every linear filter has an impulse response, a step response and a frequency response.

**2 LITERATURE SURVEY**

**Design of Fast FIR Filter Using Compressor and Carry Select Adder**. Subject 1:Speed and area are now a day’s one of the fundamental design issues in digital era. To increase speed, while doing the multiplication or addition operations, has always been a basic requirement of designing of advanced system and application. Carry Select Adder (CSA) is a fastest adder used in many processors to accomplish fast arithmetic function. Many different adder architecture designs have been developed to increase the efficiency of the adder. It is very commonly known that per second any processors performed millions of work functions in semiconductor industry. So when we do designing of multipliers, one of the main standards is performing speed that should be taken in the mind. In this paper, we propose a technique for designing of FIR filter using multiplier based on compressor and carry select adder.

**Implementation of Kogge Stone Adderfor Signal Processing Application.**

Low power system design has turn out to be a significant performance goal. The Finite Impulse Response Filter is an efficient component for digital signal processing applications. Adders and multipliers plays an essential role in implementation [R1] of FIR filter. The proposed FIR Filter is designed by using Kogge stone adder and booth multiplier. Kogge stone adder is a high speed adder which is used for designing high performance circuits and Booth multiplier is a multiplication algorithm which is used to perform multiplication by using 2‟s complement notation.

**Implementation of programmable fir filter using dadda multiplier and parallel prefix adder.**

Digital filters have a magnificent role in various applications related to signal processing. It is the performance of the filters that made DSP popular. Filtering is usually done to obtain a desired output by manipulating the input data. Various types of filters are used to manipulate the data that helped in creating different applications to benefit the world. The major part in any filter design is the multiplication block as the performance of any filter depends on how the multiplication is performed. There are various multipliers of which Dadda multiplier is one, the significance of this is that it makes use of very few gates to perform multiplication. Keeping this in view a programmable FIR filter design has been carried out and implemented using the MAC (Multiply and Accumulate) unit.

**Design of Fast FIR Filter Using Compressor and Carry Select Adder.Authors: .**Deepak Kumar Patel, Raksha Chouksey, Dr. Minal Saxena.

Speed and area are now a day's one of the fundamental design issues in digital era. To increase speed, while doing the multiplication or addition operations, has always been a basic requirement of designing of advanced system and application. Carry Select Adder (CSA) is a fastest adder used in many processors to accomplish fast arithmetic function. Many different adder architecture designs have been developed to increase the efficiency of the adder. It is very commonly known that per second any processors performed millions of work functions in semiconductor industry

**3 EXISTING SYSTEM**

**Sklansky Adder:** The Sklansky Adder Is A Type Of Parallel Prefix Adder Used In Digital Circuits For High-Speed Arithmetic Operations, Particularly For Addition. It Is Named After J. Sklansky, Who Introduced It In His Work On Carry Propagation Techniques In Binary Addition. Features

Structure: It Is A Parallel Prefix Adder, Meaning It Computes The Carry Signals In Parallel Using A Tree Structure.The Sklansky Adder Generates Carry Signals Efficiently By Grouping Bits And Calculating Carry Signals In A Hierarchical Manner.

Carry Computation: The Adder Computes Carry Signals Using A Generate-Propagate (G-P) Logic: Generate (Gig\_I): A Carry Is Generated By Bit Ii.Propagate (Pip\_I): A Carry Is Propagated Through Bit Ii.

Advantages: Low Logic Depth: The Tree Structure Minimizes The Propagation Delay, As It Reduces The Number Of Sequential Logic Gates On The Critical Path.High-Speed Performance: The Logarithmic Time Complexity (Log⁡2n\Log\_2 N, Where Nn Is The Number Of Bits) Makes It Suitable For High-Speed Arithmetic In Processors.



Disadvantages:High Fan-Out: The Structure Requires Multiple Signals to Branch Out and Connect to Multiple Gates, Leading to Higher Fan-Out At Some Nodes.Area Consumption: The Complexity of Interconnects Can Result in a Larger Area for Implementation Compared to Other Adders like the Brent-Kung Adder.

**4 PROPOSED SYSTEM**

**Kogge Stone Adder:** The Kogge-Stone Adder Is A High-Performance Parallel Prefix Adder Widely Used In Digital Circuits For Fast Arithmetic Operations, Particularly In Processors And High-Speed Computing Systems. It Was Introduced By Peter M. Kogge And Harold S. Stone In Their 1973 Paper On Carry Propagation. Features

**Structure:**It Is A Parallel Prefix Adder That Computes Carry Signals In Parallel Using A Prefix Tree Structure.The Kogge-Stone Adder Employs A Balanced Binary Tree For Carry Generation And Propagation, Ensuring Minimal Delay.

**Carry Computation**:The Adder Relies On The Generate-Propagate (G-P) Logic:

Generate (Gig\_I): A Carry Is Generated By Bit Ii.Propagate (Pip\_I): A Carry Is Propagated Through Bit Ii.

**Stages:**The Adder Consists Of Multiple Levels Of Computation Where Carry Signals Are Progressively Combined.Each Stage Computes Intermediate Results For Smaller Groups Of Bits, Ultimately Producing Carry Signals For All Bits.

**Advantages:**Low Logic Depth: The Carry Computation Time Is Proportional To Log⁡2n\Log\_2 N, Where Nn Is The Number Of Bits. This Ensures High-Speed Performance.

Regular Structure: The Regularity Simplifies Hardware Implementation, Making It Easier To Design And Optimize.

**Disadvantages:**High Interconnect Complexity: The Large Number Of Wires And Interconnections Increases Area And Power Consumption.Area Cost: Requires More Hardware Resources Compared To Simpler Adders Like The Ripple Carry Adder Or The Brent-Kung Adder.

**Comparison With Other Adders**

****

**Table: Comparison of Adders**

**Preprocessing Stage**:Compute The Generate (Gig\_I) And Propagate (Pip\_I) Signals For Each Bit: Gi=Ai⋅Big\_I = A\_I \Cdot B\_I Pi=Ai⊕Bip\_I = A\_I \Oplus B\_I

**Carry Propagation Stage:**Use The Prefix Tree To Combine The Gg And Pp Signals Across The Bits To Compute All Carry Signals.

**Summation Stage:**Compute The Sum Bits Using The Propagate Signals And Carry-Out Values: Si=Pi⊕Ci−1s\_I = P\_I \Oplus C\_{I-1}

The Kogge-Stone Adder Is Often Favoured For Its Speed And Regular Structure, Making It Ideal For Systems Where Performance Is A Critical Factor. However, Its Interconnect And Area Costs Can Be Limiting Factors For Resource-Constrained Designs.

**Fir Filter Using Parallel Prefix Adder**

FIR filter is design using shift and add multiplier and parallel prefix adder..

The operation required for designing anFIR filter in direct implementation is called the Multiply and Accumulate operation (MAC).

 (3)

In case of 1 tap filter, the filter co-efficient h0is directly multiplied with variable x0and the result is assigned to the output. In a 4-tap filter, the filter co-efficient are multiplied with corresponding variables, the result of 4 multipliers are added and assigned to the result.



**Fig. 4-tap FIR Filter (Direct form)**

Fig. 1 shows the data flow graph diagram for a 4-tapFIR filter. When the number of taps increases the longest path delay also increases. Also, number of multipliers and adders required in the architecture also increases. The data flow graph shown in Fig. 1 is in direct form

The design of the Multiplier/Result module was performed at the RTL level. A block diagram ofthis module is shown in Figure 3-7 and the register assignment details.The input to the module is the multiplier, B\_in, which is loaded into bits 0 to 7 of the registerwhen the LOAD\_cmd is asserted. The adder block outputs (Cout and adder\_out) are also inputsto the module. The outputs of the module are, LSB, bit 0 of the register, RB, bits 8 to 15 of theregister, and RC, bits 0 to 15 of the register. LSB is fed back to the controller to determine thenext state, while RB is fed into the adder in order to be summed with the multiplicand. RC is thefinal multiplication result and is considered valid only when the controller asserts the STOPsignal.

**Parallel Prefix Adders**

Parallel prefix adder (PPA) is a multi-bit carry-propagateadder which is used for parallel addition of two multi-bitnumbers. PPA extend the generated and propagated logic of thecarry look-ahead adder to perform addition even faster . Asthe basic schematic structure of the various PPA, perspectivearchitecture is analyzed, it consists of three stages : pre-processing stage, prefix computation stage and finalprocessing stage. Let consider each stage in more detail.



**Fig.Addition procedure using Parallel Prefix tree structures**

 In every bit (i) of the two operand block, the two input signals (ai and bi) are added to the corresponding carry-in signal (carryi) to produce sum output (sumi) The equation to produce the sum output is:

Sumi = ai ^ bi ^ carryi (1)

 Computation of the carry-in signals at every bit is the most critical and time – consuming operation. In the carry- look ahead scheme of adders (CLA), the focus is to design the carry-in signals for an individual bit additions. This is achieved by generating two signals, the generate (gi) and propagate (pi) using the equations:

 Gi = ai ^ bi (2)

 Pi = ai ^ bi (3)

The carry in signal for any adder block is calculated by using the formula

 Ci+1 = gi V (pi) (4)

Where ci must be expanded to calculate ci+1 at any level of addition

Parallel Prefix adders compute carry-in at each level of addition by combining generate and propagate signals in a different manner. Two operators namely *black* and *gray* are used in parallel prefix trees are shown in fig 2(a), fig 2(b) respectively.

(a) black operator (b) gray operator

**Fig. Operators used in Parallel Prefix trees**

The black operator receives two sets of generate and propagate signals (gi , pi),(gi-1 ,pi-1), computes one set of generate and propagate signals (go , po) by the following equations:

 Go = gi V (pi ^ gi-1) (5)

 Po = pi ^ pi-1 (6)

The gray operator receives two sets of generate and propagate signals (gi, pi),(gi-1 ,pi-1), computes only one generate signal with the same equation as in equation



**Fig. 16 bit Kogge-Stone adder**

The construction of the first level of the prefix tree of thisadder is similar to the construction of Kogge-Stone adder. Themain structural difference begins from the second level of theprefix tree. At the second level of the prefix tree, the groups oftwo schematic nodes are formed, at the 3rd level – groupscompose four schematic nodes and at the 4th level – groupsincluding 8 schematic nodes, etc.

**5. RESULTS:**

**Simulationresult of fir filter:**

****

**fig. Simulation Result For fir Filter**

Here we can give the inputs as x1 = 7,x2 = 3 x3 = 3 x4 = 3,h1 = 3 h2 = 3 h3 = 3 h4 = 3 and result as y = 48.

**RTL SCHEMATIC PF MULTIPLIER**:

****

**Fig RTL Schematic of Multiplier**

**Internal Diagram Of RTL Schematic:**



**Fig Internal diagram of RTL schematic**

 **Estimation of power:**

****

**Fig Estimation of power**

**Estimation of Delay:**



**Fig Estimation of Delay**

**Estimation of Area:**



**Fig Estimation of Area**

 **Synthesis Report:**

|  |  |  |
| --- | --- | --- |
|  | **Fir filter using sklansky** | **Fir filter using KSA** |
| **Area**  | **7355** | **7184** |
| **Delay**  | **111.454ns** | **108.642ns** |
| **Speed**  | **89.72Mhz** | **92.04Mhz** |

**Table . Comparison between power and delay**

**CONCLUSION:**

The FIR filter was designed and result shown in Xilinx ISE 14.7. The result of the FIR filter using KSAis compared with the FIR filter using sklansky adder.After comparison we came to know that FIR filter using kogge stone adder architecture is faster compared to FIR filterusing sklansky adder.

**REFERENCES:**

[1] Haichen Zhao, Shaolu Hu, Linhua Li, Xiaobo Wan. “NLMS AdaptiveFIR Filter Design Method”.

[2] B. Ramkumar and Harish M Kittur, “Low-Power and AreaEfficient Carry Select Adder”, IEEE Transsactions on Very Large ScaleIntegration (VLSI) Systems, VOL. 20, No. 2 Feb 2012.

[3] Deepak Kumar Patel, Raksha Chouksey, Dr. Minal Saxena “Design ofFast FIR Filter Using Compressor and Carry Select Adder”, 2016 3rdInternational Conference on Signal Processing and Integrated Networks(SPIN).

[4] A.Abinaya , M.Maheswari. “Implementation of Kogge Stone Adder forSignal Processing Applications”, International Journal of AdvancedResearch in Computer and Communication Engineering Vol. 8, Issue 5,May 2019.

[5] S. Madhavi, K. Rasagna, N. Kavya, M. Sindhu. “Implementation ofprogrammable fir filter using dadda multiplier and parallel prefixadder”,IEEE Xplore Compliant Part Number:CFP18N67-ART;ISBN:978-1-5386-2456-2.

[6] V. Jamuna, P. Gomathi and A. Arun, “Design and Implementation ofFIR Filter Architecture using High Level

Transformation Techniques”Indian Journal of Science and Technology.

[7] M.Moghaddam, M. B. Ghaznavi-Ghoushchi. “A New Low-Power, Lowarea,Parallel Prefix Sklansky Adder with Reduced Inter-StageConnections Complexity”.

[8] Aung Myo San, Alexey N. Yakunin “Reducing the HardwareComplexity of a parallel prefix adder”.