International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

COPY RIGHT

IJIEMR Transactions, online available on 25th Sept 2018. Link
:http://www.ijiemr.org/downloads.php?vol=Volume-7&issue=ISSUE-10

Title: **DESIGN AND ANALYSIS OF FAULT TOLERANT PARALLEL FFTS BASED ON ERROR CORRECTION CODES AND PARSEVAL CHECKS**

Volume 07, Issue 10, Pages: 105–113.
Paper Authors

NALABOTHU TEJASWI, J.RAVI SANKAR
Krishnaveni Engineering College for Women , Kesanupalli.

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

DESIGN AND ANALYSIS OF FAULT TOLERANT PARALLEL FFTS BASED ON ERROR CORRECTION CODES AND PARSEVAL CHECKS

NALABOTHU TEJASWI[1], J.RAVI SANKAR [2]

[1]PG Scholar, ECE Department, Krishnaveni Engineering College for Women , Kesanupalli.
[2] Professor, ECE Department, Krishnaveni Engineering College for Women, Kesanupalli.

Abstract:

The intricacy of interchanges and flag processsing circuits builds each year. This is made conceivable by the CMOS innovation scaling that empowers the mix of more transistors on a solitary gadget. This expanded many-sided quality makes the circuits more defenseless against mistakes. In the meantime, the scaling implies that transistors work with bring down voltages and are more susceptible to blunders caused by commotion and manufacturing varieties. Delicate blunders represent a dependability risk to present day electronic circuits. This makes assurance against delicate mistakes a necessity for some applications. Correspondences and flag preparing frameworks are no exemptions to this pattern. For a few applications, an intriguing choice is to utilize algorithmic-based adaptation to non-critical failure (ABFT) techniques that attempt to misuse the algorithmic properties to identify and rectify blunders. Flag preparing and correspondence applications are appropriate for ABFT. One precedent is quick Fourier changes (FFTs) that are a key building hinder in numerous frameworks. A few insurance plans have been proposed to identify and amend mistakes in FFTs. Among those, likely the utilization of the Perseval or whole of squares check is the most generally known. In present day correspondence frameworks, it is progressively normal to discover a few squares working in parallel. As of late, a procedure that endeavors this reality to execute adaptation to non-critical failure on parallel channels has been proposed. In this concise, this system is first connected to secure FFTs. At that point, two enhanced insurance plots that consolidate the utilization of mistake remedy codes and Perceval checks are proposed and assessed.

1 Introduction:

Mistake rectification code (ECC) methods have been broadly used to rectify transient blunders and enhance the unwavering quality of recollections. ECC words in recollections comprise of information bits and extra check bits on the grounds that the ECCs utilized in recollections are commonly from a class of straight square codes. Amid the compose activities of recollections, information bits are composed in information bit clusters, and check bits are simultaneously created utilizing the information bits and put away in line bit exhibits. The check bit exhibits, much the same as the information bit clusters, ought to be tried wisely for a similar blame models if solid mistake revision is to be insured[1]. Quick Fourier change is utilized to change over a flag from time space to recurrence and this is required with the goal that you

can see the recurrence parts present in a signs. In the event that you know the recurrence segments present in a signs you can play with the signs :) Let's say, you need to plan a low pass channel and need to settle on the cut off recurrence of the channel. In the event that you have the recurrence area subtle elements for a signs yooooou can unmistakably recognize the recurrence parts which you need to hold and the ones which u need to take out[2]. Natural obstruction and physical deformities in the correspondence medium can cause arbitrary piece mistakes amid information transmission. Mistake coding is a strategy for identifying and remedying these blunders to guarantee data is exchanged unblemished from its source to its goal. Blunder coding is utilized for blame tolerant registering in PC memory, attractive and optical information stockpiling media, satellite and profound space communications, arrange interchanges, cell phone systems, and some other type of computerized information correspondence. Blunder coding utilizes numerical recipes to encode information bits at the source into longer piece words for transmission. The "code word" would then be able to be decoded at the goal to recover the information. The additional bits in the code word give excess that, as per the coding plan utilized, will enable the goal to utilize the deciphering procedure to decide whether the correspondence medium presented mistakes and now and again rectify them with the goal that the information require not be retransmitted. Distinctive blunder coding plans are picked relying upon the sorts of mistakes expected, the correspondence medium's normal mistake rate, and regardless of whether information retransmission is conceivable. Quicker processors and better interchanges innovation make more perplexing coding plans, with better blunder distinguishing and amending capacities, workable for littler implanted frameworks, taking into consideration more vigorous correspondences. Be that as it may, tradeoffs among data transfer capacity and coding overhead, coding many-sided quality and suitable coding delay between transmissions, must be considered for every application. Transient mistakes can frequently resentful in excess of one piece creating multi-bit blunders with a high likelihood of mistake event in neighboring memory cells . Bit interleaving is one procedure to cure multi-bit blunders in neighboring memory cells as physically contiguous bits in memory cluster are appointed to various legitimate words [5],[6]. The single-mistake adjustment, twofold blunder location, and twofold adjoining mistake revision (SEC-DED-DAEC) codes have already been introduced to rectify nearby twofold piece blunders [4]-[7]. The required number of check bits for the SEC-DED-DAEC codes is the same as that for the SEC-DED codes.

What's more, the region and timing overheads for encoder and decoder of the SEC-DED-DAEC codes are like those of the SEC-DED codes. Thus, nearby twofold piece mistakes can be cured with next to no

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

extra cost utilizing the SECDED-DAEC codes. The SEC-DED-DAEC codes might be an alluring option in contrast to bit interleaving in giving more prominent adaptability to streamlining the memory format. Besides, the SEC-DED-DAEC code can be utilized related to bit interleaving and this strategy can productively manage neighboring multi-bit blunders [1]

The FFTs in parallel expands the extent of applying mistake rectification codes together. Creating equality together for parallel FFTs likewise helps in limiting the unpredictability in some ECC [15]. By expecting that there must be a solitary blunder on the framework on account of radiation-initiated delicate mistakes and might be two in most pessimistic scenario. The proposed new method depends on the mix of Partial Summation joined with equality FFT for numerous mistake revision.

The FFTs in parallel expands the extent of applying blunder amendment codes together. Creating equality together for parallel FFTs likewise helps in limiting the unpredictability in some ECC [15]. By expecting that there must be a solitary mistake on the framework on account of radiation-incited delicate blunders and might be two in most pessimistic scenario. The proposed new procedure depends on the mix of Partial Summation joined with equality FFT for various blunder amendment.

Fast Fourier Transform

Fast Fourier Transform (FFT) algorithm converts a signal from time domain into a sequence in the frequency domain [13]. Fast Fourier transforms are widely used for many applications which include engineering, science, and mathematics. It computes transformations through DFT matrix. The FFT operation starts with decomposing N-point time domain signal and calculating N frequency spectra and finally forming a single spectrum.

The Discrete Fourier Transform (DFT)

Discrete Fourier Transform (DFT) is an important unit in many communication applications like OFDM, etc. DFT is also measured as one of the tools to act upon frequency analysis of discrete time signals. The Discrete Fourier Transform is a continuous Fourier transform for the use of discrete functions. Given a real sequence as the input, the DFT outputs them as a sequence of complex numbers. The mathematical representation of the transform is

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-\frac{j2\pi kn}{N}}, n = 0,1, \ldots \ldots N-1 \qquad (1)$$

If an N – point DFT is implemented directly, the necessity of arithmetic units is of the order of O(N2) that is N2 multiplications and N (N-1) additions. Thus FFT is used for designing the DFT. Depending on inputs being real or complex, the design of adders and multipliers are formed.

Divide and Conquer Approach to Computation of the DFT

The reduction of computational complexity algorithm for DFT is made possible by using a divide and conquers approach. This approach decomposes a larger DFT into smaller one forming a collective FFT

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

algorithm. Let us consider N-point DFT. It is one of the well-organized ways to implement Discrete Fourier Transform (DFT) due to its compact use of arithmetic blocks. The FFT and inverse FFT of an N point signals are given below.

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad (2)$$

$$x(n) = \frac{1}{N} \sum_{n=0}^{N-1} X(k) W_N^{-kn} \quad (3)$$

Where

$$W_N^{kn} = e^{-j\frac{2\pi}{N}kn}$$

From the above condition it is obvious that both FFT and its opposite are relatively same with little variety. Reverse FFT configuration is shaped by utilizing separate by N-point and taking the conjugate of the twiddle factors. Because of the continuous utilization of FFT in present day remote plans, higher radix FFTs, for example, radix-4, radix-8, radix-2k, split radix, and so forth are intended for enhancing the execution timing and diminishing the multifaceted nature. The contrast between these outlines depends on their butterfly units.

Fundamental idea of 4-focuses DIF FFT circuit which applies Radix-2 design is in Figure 1 indicates estimation flag

stream diagram about discrete Fourier coefficient of N=4. Here W40, W41 are the twiddle variables of the four point Fast Fourier Transform. Note, under the bolt of is subtraction; twiddle factor at the highest point of line is increase. In butterfly preparing component that is indicated red line make be comparing to next square in the slide outline. Another butterfly preparing component hues are same. This is the 4 focuses FFT circuit for parallel info. By and large, FFT dissects an info flag succession by utilizing destruction in-recurrence (DIF) or devastation in-time (DIT) deterioration to outline an effective flag stream chart (SFG). Here, the printed material centers DIF deterioration since it matches with different pipelined plans. x(0), x(1), x(2) and x(3) are the info time area signals with 1, - 1 and – j as the twiddle factors delivering X(0), X(1), X(2) and X(3) as the recurrence space yields.

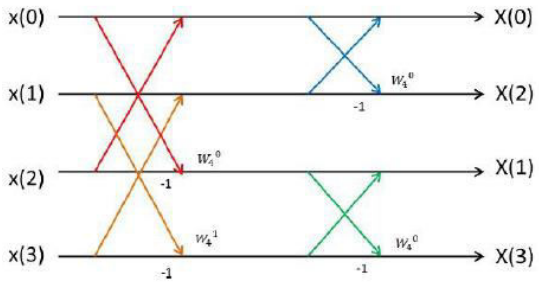


Figure 1. Signal Flow Graph of 4-point FFT

2. Error Tolerant Techniques for Parallel FFTs

Error Correction based on Hamming Codes

The point of mistake tolerant plan is to shield parallel FFTs from blunders. Different plans have been proposed for blunder discovery and revision in FFTs. One of the essential and straightforward techniques is mistake redress utilizing hamming codes. Not at all like equality code which can distinguish just odd piece blunder, the hamming code can identify no good mistakes and right one blunder. Like other blunder redress codes, hamming codes additionally uses the equality bit which is created for the comparing input succession for recognizing mistakes [14]. It

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

accomplishes higher code rate with least separation of three. The quantity of equality bits relies upon the aggregate number of information bits. For instance, hamming code with 4 data bits produces 7 encoded information bits with its distinction being the equality. For this situation, the three equality bits h1, h2, h3 are processed as an information bits c1, c2, c3, c4 as portrayed underneath:

$$h1 = c1 \oplus c2 \oplus c3$$
$$h2 = c1 \oplus c2 \oplus c4$$
$$h3 = c1 \oplus c3 \oplus c4$$

The limitations of is that, during the multiple error scenario hamming code will not be able to exactly identify the individual FFTs with error.

Fault tolerant FFT based on Parseval's check

Parseval's strategy is one of the systems to distinguish blunders parallel in different FFT. This is accomplished with Sum of Squares (SOSs) check [5] in light of Parseval's hypothesis. The mistake free FFT ought to have its Sum of Squares of the information paralleling the Sum of Squares of its recurrence area yield. This relationship can be utilized to recognize mistakes with least overhead. For parallel FFTs, the Parseval's check can be joined with the mistake amendment codes to limit the zone overhead. Various blunder recognition and remedy is accomplished through this blend. One of the simple courses is to create the excess contribution for single FFT with all the four FFT inputs. To rectify blunder the equality FFT yield is XORed with blame free yields of the FFTs. Contrasted with the past plans exhibited in the Fault Tolerant Parallel FFTs Using Error Correction Codes and Parseval Checks [1], this method decreased the aggregate number of Sum of Squares utilized. Another current work done is by joining SOS checks with hamming codes as opposed to utilizing Parseval's check separately as appeared in Figure 2.

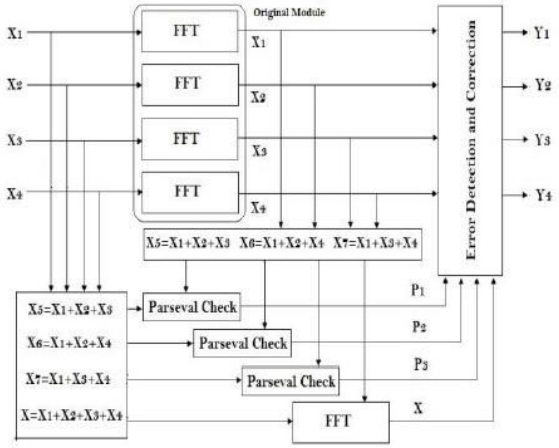


Figure 2. Parity-SOS-ECC fault-tolerant parallel FFTs

This strategy consolidates the element of equality figuring of hamming codes and blunder recognition procedure of Sum of Squares. Simultaneous Error Detection (CED) plans for the FFT are the Sum of Squares (SOS) check in view of Pa hypothesis. The utilization of parseval check is exponentially diminished to the immediate correlations of FFTs information sources and yields used to ensure parallel FFTs

3. PROPOSED PROTECTION SCHEMES FOR PARALLEL FFTS

The beginning stage for our work is the security plot in light of the utilization of ECCs that was exhibited in [17] for

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

computerized channels. This plan is appeared in Fig. 1. In this model, a basic single mistake remedy Hamming code [18] is utilized. The first framework comprises of four FFT modules and three repetitive modules is added to distinguish and adjust mistakes. The contributions to the three repetitive modules are direct blends of the information sources and they are utilized to check straight mixes of the yields. For instance, the contribution to the main excess module is

$$x_5 = x_1 + x_2 + x_3 \quad (1)$$

and since the DFT is a linear operation, its output $z5$ can be used to check that

$$z_5 = z_1 + z_2 + z_3. \quad (2)$$

This will be meant as c1 check. A similar thinking applies to the next two excess modules that will give checks c2 and c3. In view of the distinctions saw on every one of the checks, the module on which the blunder has happened can be resolved. The distinctive examples and the comparing blunders are abridged in Table I.

Once the module in blunder is known, the mistake can be rectified by recreating its yield utilizing the rest of the modules. For instance, for a blunder influencing z1, this should be possible as takes after:

$$z_{1c}[n] = z_5[n] - z_2[n] - z_3[n]. \quad (3)$$

Comparable revision conditions can be utilized to amend blunders on alternate modules. Further developed ECCs can be utilized to amend mistakes on numerous modules if that is required in a given application. The overhead of this method, as talked about in [17], is lower than TMR as the quantity of repetitive FFTs is identified with the logarithm of the quantity of unique FFTs. For instance, to secure four FFTs, three redudant FFTs are required, yet to ensure eleven, the quantity of repetitive FFTs in just four. This shows how the overhead declines with the quantity of FFTs.

$c_1\, c_2\, c_3$	Error Bit Position
0 0 0	No error
1 1 1	z_1
1 1 0	z_2
1 0 1	z_3
0 1 1	z_4
1 0 0	z_5
0 1 0	z_6
0 0 1	z_7

TABLE I ERROR LOCATION IN THE HAMMING CODE

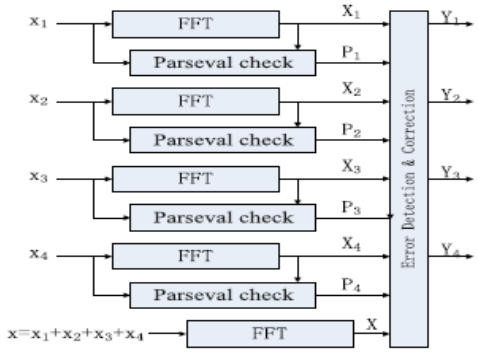


Fig. 3. Parity-SOS (first technique) fault-tolerant parallel FFTs.

In Section I, it has been specified that throughout the years, numerous systems have been proposed to ensure the FFT. One of them is the Sum of Squares (SOSs) check [4] that can be utilized to recognize blunders. The SOS check depends on the Parseval hypothesis that expresses that the SOSs of the contributions to the FFT are equivalent to the SOSs of the yields of the

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

FFT with the exception of a scaling factor. This relationship can be utilized to identify mistakes with low overhead as one augmentation is required for each info or yield test (two increases and adders for SOS per test). For parallel FFTs, the SOS check can be joined with the ECC way to deal with lessen the security overhead. Since the SOS check can just identify blunders, the ECC part ought to have the capacity to execute the adjustment. This should be possible utilizing what might as well be called a straightforward equality bit for all the FFTs. Furthermore, the SOS check is utilized on each FFT to identify mistakes. At the point when a blunder is identified, the yield of the equality FFT can be utilized to redress the mistake. This is better clarified with a precedent. In Fig. 2, the primary proposed conspire is shown for the instance of four parallel FFTs. An excess (the equality) FFT is included that has the total of the contributions to the first FFTs as information. A SOS check is likewise added to every unique FFT. On the off chance that a mistake is identified (utilizing P1, P2, P3, P4), the rectification should be possible by recomputing the FFT in blunder utilizing the yield of the equality FFT (X) and whatever remains of the FFT yields. For instance, if a mistake happens in the main FFT, P1 will be set and the blunder can be rectified by doing

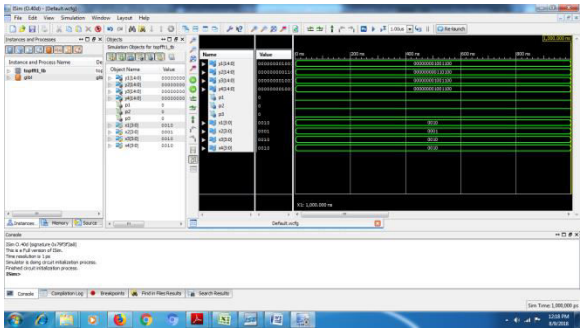
$$X_{1c} = X - X_2 - X_3 - X_4. \quad (4)$$

This blend of an equality FFT and the SOS check decreases the quantity of extra FFTs to only one and may, along these lines, lessen the insurance overhead. In the accompanying, this plan will be alluded to as equality SOS (or first proposed system). Another plausibility to consolidate the SOS check and the ECC approach is as opposed to utilizing a SOS check for each FFT, utilize an ECC for the SOS checks. At that point as in the equality SOS conspire, an extra equality FFT is utilized to adjust the mistakes. This second method is appeared in Fig. 3. The primary advantage over the principal equality SOS plot is to decrease the quantity of SOS checks required. The blunder area process is the same concerning the ECC conspire in Fig. 1 and redress is as in the equality SOS conspire. In the accompanying, this plan will be alluded to as equality SOS-ECC (or second proposed procedure). The overheads of the two proposed plans can be at first evaluated utilizing the quantity of extra FFTs and SOS check squares required. In every one of the methods talked about, delicate blunders can likewise influence the components included for insurance. For the ECC strategy, the insurance of these components was talked about in [17]. On account of the excess or equality FFTs, a mistake will have no impact as it won't spread to the information yields and won't trigger a revision. On account of SOS checks, a blunder will trigger an amendment when quite is no mistake on the FFT. This will cause a pointless adjustment however will likewise deliver the right outcome. At long last, mistakes on the identification and remedy obstructs in Figs. 2 and 3 can spread blunders to the yields. In our usage, those squares are ensured with TMR. The same applies for the adders used to process the contributions to the excess FFTs in Fig. 1 or

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

to the SOS checks in Fig. 3. The triplication of these squares smallly affects circuit many-sided quality as they are considerably more straightforward than the FFT calculations. A last perception is that the ECC plan can distinguish all mistakes that surpass a given edge (given by the quantization used to execute the FFTs) [17]. Then again, the SOS check identifies most blunders however does not ensure the discovery of all mistakes [4]. Hence, to look at the three procedures for a given usage, blame infusion tests ought to be done to decide the level of blunders that are really rectified. This implies an assessment must be done both as far as overhead and blunder inclusion.

5.RESULTS



6. CONCLUSIONS

Identifying and revising blunders, for example, basic dependability are troublesome in flag preparing which builds the utilization of blame tolerant usage. In current flag preparing circuits, usually to discover a few channels working in parallel. Proposed is a territory productive procedure to identify and revise single mistakes. This brief has exhibited another plan to ensure parallel FFT utilizing cordic that is regularly found in present day flag preparing circuits. The methodology depends on applying SOS-ECC check to the parallel FFT yields to identify and adjust blunders. The SOS checks are utilized to recognize and find the mistakes and a straightforward equality FFT is utilized for adjustment. The 8 point FFT with the info bit length 32 is ensured utilizing the proposed procedure. . The discovery and area of the blunders should be possible utilizing a SOS check for each FFT or then again utilizing an arrangement of SOS watches that shape an ECC. This strategy can distinguish and revise just single piece mistake and it lessens region results in fast contrasted with existing methods.

7. FUTURE WORK

In Future, utilization of DCT instead of FFT will be carried out. Since SOS-ECC technique can detect and correct only single bit fault, this will be extended to multi bit faults by using the trellis code and hence area will be further reduced.

REFERENCES

[1] R. Baumann. 2005. Soft errors in advanced computer systems. IEEE Des. Test Comput. 22(3): 258-266.

[2] M. Ergen. 2009. Mobile Broadband-Including WiMAX and LTE. New York, NY, USA: Springer- Verlag.

[3] Z. Gao *et al*. 2015. Fault tolerant parallel filters based on error correction codes. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 23(2): 384-387.

[4] R. W. Hamming. 1950. Error detecting and error correcting codes. Bell Syst. Tech. J. 29(2): 147-160.

[5] T. Hitana and A. K. Deb. 2004. Bridging concurrent and non-concurrent error

detection in FIR filters. in Proc. Norchip Conf. pp. 75-78.

[6] J. Y. Jou and J. A. Abraham. 1988. Fault-tolerant FFT networks. IEEE Trans. Comput. 37(5): 548-561.

[7] N. Kanekawa, E. H. Ibe, T. Suga and Y. Uematsu. 2010. Dependability in Electronic Systems: Mitigation of Hardware Failures, Soft Errors, and Electro-Magnetic Disturbances. New York, NY, USA: Springer-Verlag.

[8] E. P. Kim and N. R. Shanbhag. 2012. Soft N-modular redundancy. IEEE Trans. Comput. 61(3): 323-336.

[9] M. Nicolaidis. 2005. Design for soft error mitigation. IEEE Trans. Device Mater. Rel. 5(3): 405-418.

[10] S. Pontarelli, G. C. Cardarilli, M. Re and A. Salsano. 2008. Totally fault tolerant RNS based FIR filters. In Proc. 14th IEEE Int. On-Line TestSymp. (IOLTS). pp. 192-194.

[11] A. L. N. Reddy and P. Banerjee. 1990. Algorithmbased fault detection for signal processing
applications. IEEE Trans. Comput. 39(10): 1304- 1308.

[12] P. Reviriego, C. J. Bleakley and J. A. Maestro. 2012. A novel concurrent error detection technique for the fast Fourier transform. In Proc. ISSC, Maynooth, Ireland. pp. 1-5.

[13] P. Reviriego, S. Pontarelli, C. J. Bleakley and J. A. Maestro. 2012. Area efficient concurrent error detection and correction for parallel filters. IET Electron. Lett. 48(20): 1258-1260