International Journal of Scientific Engineering and Technology
Volume No.5 Issue No.3, pp: 134-138

ISSN:2277-1581
01 March.2016

Design and Analysis of Algorithm Based Error Detect-Corrections of Parallel FFT

Jessy.L[1], Nithya.K[2]
ECE Department, (VLSI Design), JCT College of Engineering and Technolgy
yssejjessy@gmail.com, nithi.vasantha@gmail.com

Abstract: The overture of the work is principally focused on Error Corrections Code (ECC) for the Parallel Transforms. And this error corrections code doesn't handle the more designing area of the actual design. To detect and correct errors with the help of algorithmic properties such as Algorithm Based Fault Tolerance (ABFT) techniques makes the avoidance of soft errors. By using algorithm based fault tolerance technique, the communication systems and signal processing systems, can be implemented with fast Fourier transforms (FFT) which are considered to be the basic building blocks of all signal processing systems. The proposed method can able detect multiple errors and multiple error corrections also possible upon best-case analysis.

Keywords- ECC, ABFT, FFT, SPS.

I INTRODUCTIONS

Error correction code (ECC) techniques have been widely used to correct transient errors and improve the reliability of memories. ECC words in memories consist of data bits and additional check bits because the ECCs used in memories are typically from a class of linear block codes. During the write operations of memories, data bits are written in data bit arrays, and check bits are concurrently produced using the data bits and stored in check bit arrays. The check bit arrays, just like the data bit arrays, should be tested prudently for the same fault models if reliable error correction is to be insured[1].

Fast Fourier transform is used to convert a signal from time domain to frequency & this is needed so that you can view the frequency components present in a signals. If you know the frequency components present in a signals you can play with the signals :) Let's say, u want to design a low pass filter and want to decide on the cut off frequency of the filter. If you have the frequency domain details for a signals u can clearly identify the frequency components which u want to retain & the ones which u want to take out[2].

Simultaneous testing of data bit and check bit arrays has been proposed in order to reduce the test time and hardware overheads required for separate check bit array tests. Simultaneous testing of data bit and check bit arrays using the conventional SEC code brings decreases of about 23.8%, 15.8%, and 9.9% in the time required for memory array tests for 16, 32, and 64 data bits per word[2],[3]. Although SEC codes and SEC-DED codes are capable of correcting single bit errors and are widely used in memories, they cannot correct a double or more bit error in an ECC word.

Environmental interference and physical defects in the communication medium can cause random bit errors during data transmission. Error coding is a method of detecting and correcting these errors to ensure information is transferred intact from its source to its destination. Error coding is used for fault tolerant computing in computer memory, magnetic and optical data storage media, satellite and deep space communications, network communications, cellular telephone networks, and almost any other form of digital data communication.

Error coding uses mathematical formulas to encode data bits at the source into longer bit words for transmission. The "code word" can then be decoded at the destination to retrieve the information. The extra bits in the code word provide redundancy that, according to the coding scheme used, will allow the destination to use the decoding process to determine if the communication medium introduced errors and in some cases correct them so that the data need not be retransmitted.

Different error coding schemes are chosen depending on the types of errors expected, the communication medium's expected error rate, and whether or not data retransmission is possible. Faster processors and better communications technology make more complex coding schemes, with better error detecting and correcting capabilities, possible for smaller embedded systems, allowing for more robust communications. However, tradeoffs between bandwidth and coding overhead, coding complexity and allowable coding delay between transmissions, must be considered for each application.

Transient errors can often upset more than one bit producing multi-bit errors with a very high probability of error occurrence in neighboring memory cells . Bit interleaving is one technique to remedy multi-bit errors in neighboring memory cells as physically adjacent bits in memory array are assigned to different logical words [5],[6]. The single-error-correction, double-error-detection, and double-adjacent-error-correction (SEC-DED-DAEC) codes have previously been presented to correct adjacent double bit errors [4]-[7]. The required number of check bits for the SEC-DED-DAEC codes is the same as that for the SEC-DED codes.

In addition, the area and timing overheads for encoder and decoder of the SEC-DED-DAEC codes are similar to those of the SEC-DED codes. Consequently, adjacent double bit errors can be remedied with very little additional cost using the SEC-DED-DAEC codes. The SEC-DED-DAEC codes may be an attractive alternative to bit interleaving in providing greater flexibility for optimizing the memory layout. Furthermore, the SEC-DED-DAEC code can be used in conjunction with bit interleaving and this method can efficiently deal with adjacent multi-bit errors [1]

doi : 10.17950/ijset/v5s3/302

The concept of multiple bit error detections and soft error Corrections for Parallel Transforms are new for the Error detections and corrections for DSP applications which is explained by the below sections.

II FUNCTIONAL FORMULATION

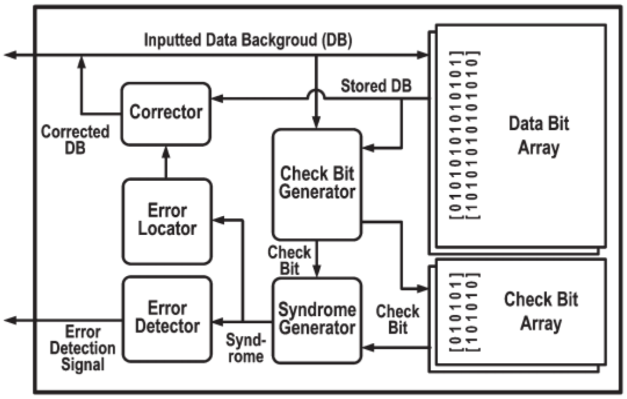


Figure 1: Basic structure of error handling mechanism

The figure 1 shows Error handling procedure of the entire mechanism. Starting form error detections and checking sections and corrections are detail described in the diagram. The first and second constraints are related with the single error-correction (SEC) function. The first constraint guarantees the syndrome to be a non-zero vector when a single bit error occurs. The second constraint ensures that all single bit errors are correctable. The third constraint is related with the DED function. In the odd-weight-column code which is the most widely used SEC-DED code in memories, the XOR-sum of any two odd-weight columns always becomes an even-weight vector and then the third constraint is satisfied. When the odd weight-column code is used, if the weight of the syndrome is an even number, a double bit error is deemed to be detected [3].

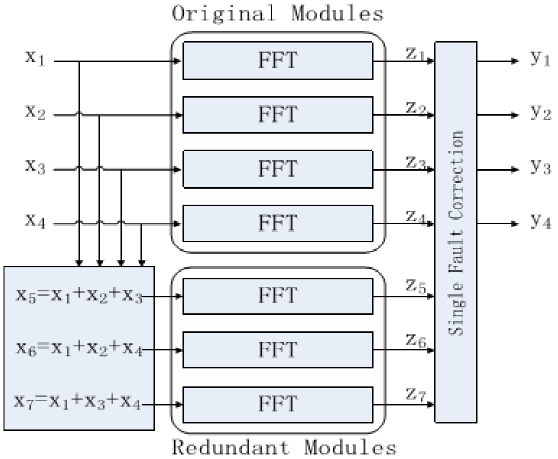


Figure 2: Parallel FFT protection using ECCs.

The figure 2 indimate the Error Correction Code of the Parallel FFT[2].For this purpose the ECC need Additionally 3FFT causing more area.the detailed functions of this method is

x5 <= x1 xor x2 xor x3 (1)

x6 <= x1 xor x2 xor x4 (2)

x7<= x1 xor x3 xor x4 (3)

which is continue vice versa for parallel FFT respectively.

The starting point for our work is the protection scheme based on the use of ECCs that was presented in for digital filters. This scheme is shown in Fig.2. In this example, a simple single error correction Hamming code is used.The original system consists of four FFT modules and three redundant modules is added to detect and correct errors. The inputs to the three redundant modules are linear combinations of the inputs and they are used to check linear combinations of the outputs.for this pupose we have to chosen the 4point fft as well as 8point FFT.the designing of the fft is also checked with fault injected fft. The overhead of this technique, as discussed in is lower than TMR as the number of redundant FFTs is related to the logarithm of the number of original FFTs. For example, to protect four FFTs, three redudant FFTs are needed, but to protect eleven, the number of redundant FFTs in only four. This shows how the overhead decreases with the number of FFTs. In Section I, it has been mentioned that over the years, many techniques have been proposed to protect the FFT. One of them is the Sum of Squares (SOSs) check that can be used to detect errors. .

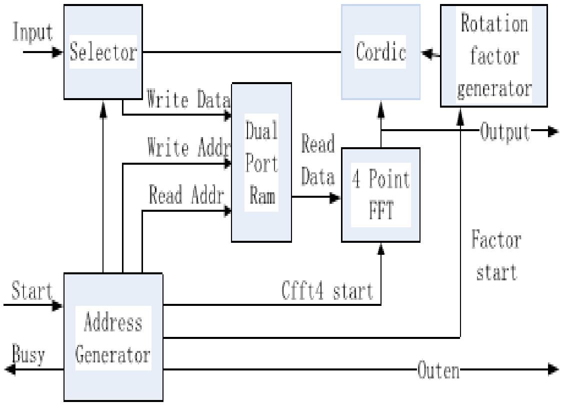


figure 3:Architecuter of the FFT implementations.

The figure 3 shows the architecure of the fft implementations which is taken from the Fault Tolerant Parallel FFTs Using Error Correction Codes.The data and parity check bits are stored and can be recovered later even if there is an error in one of the bits. This is done by recomputing the parity check bits and comparing the results with the values stored. In the example considered, an error on *d*1 will cause errors on the three parity checks; an error on *d*2 only in *p*1 and *p*2; an error on *d*3 in *p*1 and *p*3; and finally an error on *d*4 in *p*2 and *p*3. Therefore, the data bit in error can be located and the error can be corrected. This is commonly formulated in terms of the generating *G* and parity check *H* matrixes.

The overhead of this technique, as discussed in is lower than TMR as the number of redundant FFTs is related to

ISSN:2277-1581
01 March.2016

the logarithm of the number of original FFTs. For example, to protect four FFTs, three redudant FFTs are needed, but to protect eleven, the number of redundant FFTs in only four. This shows how the overhead decreases with the number of FFTs. In Section I, it has been mentioned that over the years, many techniques have been proposed to protect the FFT. One of them is the Sum of Squares (SOSs) check that can be used to detect errors. The SOS check is based on the Parseval theorem that states that the SOSs of the inputs to the FFT are equal to the SOSs of the outputs of the FFT except for a scaling factor. This relationship can be used to detect errors with low overhead as one multiplication is needed for each input or output sample (two multiplications and adders for SOS per sample).

The part of 4point FFT algoithm is described belo by using VHDL.

```
T0      <= conv_integer(x0) + conv_integer(x2) ;
T1      <= conv_integer(x1) + conv_integer(x3) ;
 T2      <= conv_integer(x0) - conv_integer(x2) ;
  Temp_T3  <= conv_integer(x1) - conv_integer(x3) ;
T3      <= Temp_T3 * (-1);
Temp_op0    <= T0 + T1    ;
Temp_op2    <= T0 - T1    ;
Temp_op1_re <= T2         ;
Temp_op1_im <= T3         ;
Temp_op3_re <= T2         ;
Temp_op3_im <= (-1) * (T3);
```

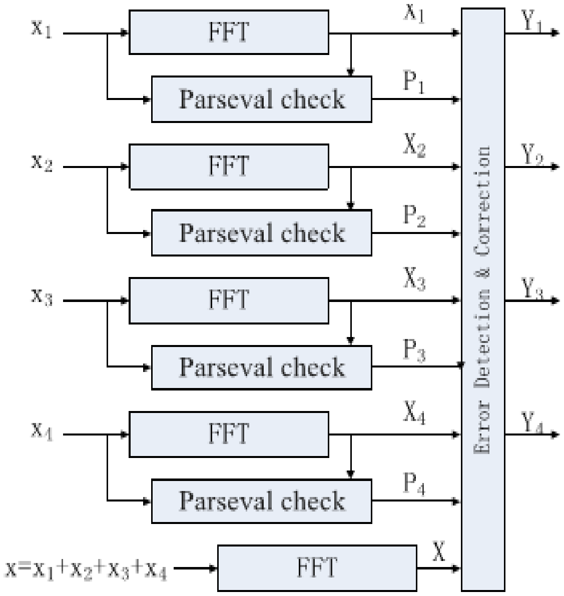


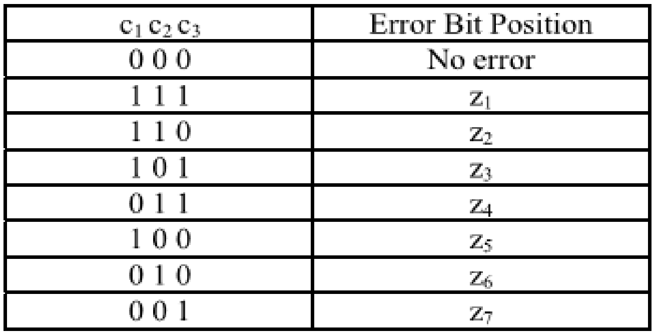
Figure 4: ECC with Pareval check

The figure 4 clearly explain the less area utilizations of the original module i.e FFT.which is done by the Parseval check. For parallel FFTs, the SOS check can be combined with the ECC approach to reduce the protection overhead. Since the SOS check can only detect errors, the ECC part should be able to implement the correction. This can be done using the equivalent of a simple parity bit for all the FFTs. In addition, the SOS check is used on each FFT to detect errors. When an error is detected, the output of the parity FFT can be used to correct the error. This is better explained with an example. In Fig. 2, the first proposed scheme is illustrated for the case of four parallel FFTs. A redundant (the parity) FFT is added that has the sum of the inputs to the original FFTs as input. An SOS check is also added to each original FFT. In case an error is detected (using P_1, P_2, P_3, P_4), the correction can be done by recomputing the FFT in error using the output of the parity FFT (X) and the rest of the FFT outputs. For example, if an error occurs in the first FFT, P_1 will be set and the error can be corrected by doing.

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

The **Fault injected FFT** Environmental interference and physical defects in the communication medium can cause random bit errors during data transmission. Error coding is a method of detecting and correcting these errors to ensure information is transferred intact from its source to its destination. Error coding is used for fault tolerant computing in computer memory, magnetic and optical data storage media, satellite and deep space communications, network communications, cellular telephone networks, and almost any other form of digital data communication.

Error coding uses mathematical formulas to encode data bits at the source into longer bit words for transmission. The "code word" can then be decoded at the destination to retrieve the information. The extra bits in the code word provide redundancy that, according to the coding scheme used, will allow the destination to use the decoding process to determine if the communication medium introduced errors and in some cases correct them so that the data need not be retransmitted. Different error coding schemes are chosen depending on the types of errors expected, the communication medium's expected error rate, and whether or not data retransmission is possible. Faster processors and better communications technology make more complex coding schemes, with better error detecting and correcting capabilities, possible for smaller embedded systems, allowing for more robust communications. However, tradeoffs between bandwidth and coding overhead, coding complexity and allowable coding delay between transmissions, must be considered for each application.



$c_1 c_2 c_3$	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 1: Error Bit Positions calculations from fault injected FFT

The above table is taken from the FFT fault corrections code[2],single bit error calculations value of the soft error were corrected by using this table.definitly it is much good to calculate the single error but if we consider multi bit detections and corrections the Decimal Matrix Codes is suitable.

III SYNTHESIS AND SIMULATIONS RESULTS

The simulation and synthesis work is finally done by the xilinix and modelsim respectively.

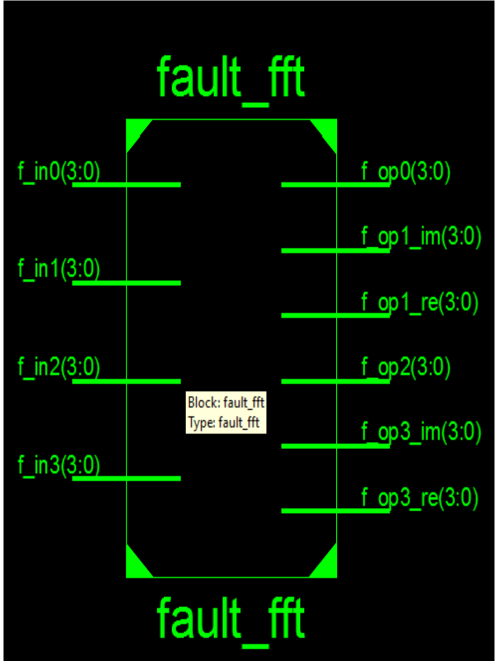


Figure 5:synthesis results of Fault FFT.

The figures intimate the fault injected FFT,which is checked by the manual error injected via all diferent possibilities by using RTL scripting. Eventhough the soft error is added in the FFT the error detector code 100% detect the errors and corrector correct the errors.

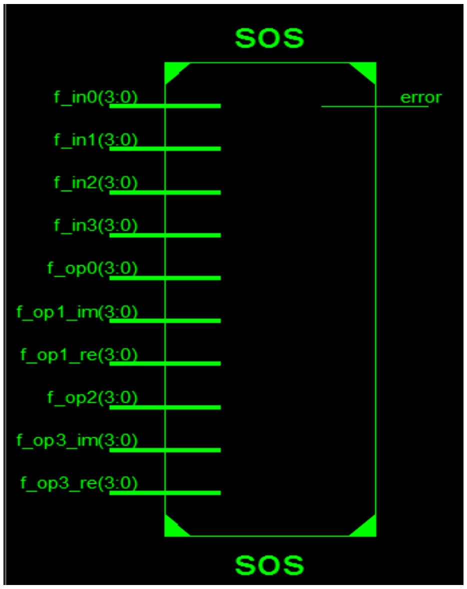
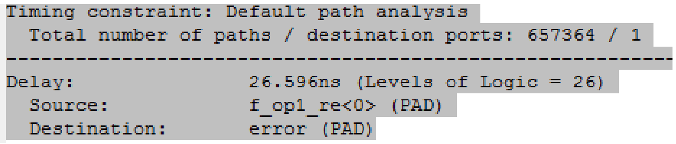


Figure 6:synthezised diagram of DMC with Sum of square algm

The above synthezised diagram is sucessfully completed by using Xilinix Syntheziser. Sum of Squares (SOSs) check that can be used to detect errors,SOS check is based on the Parseval theorem that states that the SOSs of the inputs to the FFT are equal to the SOSs of the outputs of the FFT except for a scaling factor.DMC is used for multiple bit error Detection and corrections but number of redundancy bit high in DMC based error corrections codes.

The proposed ECC codes utilize less area than previous ECC module.



```
Timing constraint: Default path analysis
  Total number of paths / destination ports: 657364 / 1
-------------------------------------------------------------
Delay:                    26.596ns (Levels of Logic = 26)
  Source:                 f_op1_re<0> (PAD)
  Destination:            error (PAD)
```

In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended to the end of data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number. Error correction code (ECC) techniques have been widely used to correct transient errors and improve the reliability of memories.here we were tried for FFT.

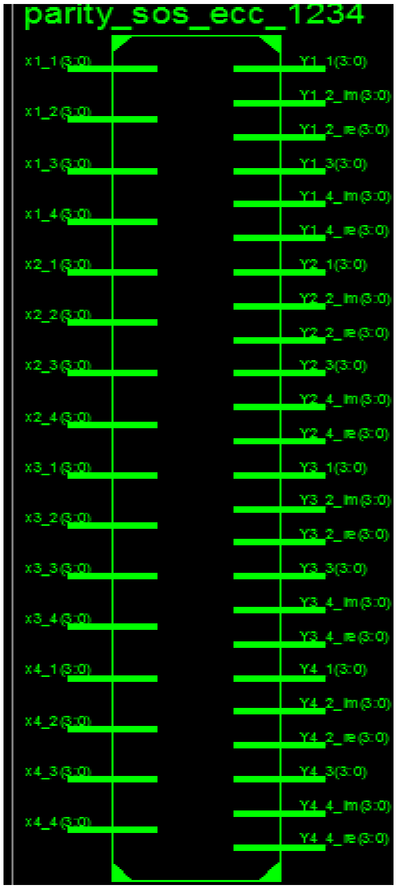


Figure 7:synthesis diagram of SOS based ECC for FFT.

The figure 7 is desinged by using verilog language with xilinix synthesis tool.for this design we had to use 4 to 8 bit Fault FFT with ECC Concept.The ECC codes utilize the less area than previous module.

doi : 10.17950/ijset/v5s3/302

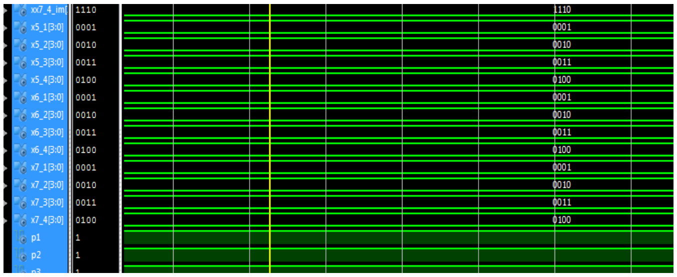


Figure 8: simulations result of Effieint ECC for parallel FFT

The figure 8 shows the simulations result of the SOS based ECC for parallel FFT,which is checked by the random test bench code in xilinix tool.here we have to reduce almost best case redundancy minimizations.the wave form is indicate the flag register for intimate the Error if happened.in the soft error testing.

IV.CONCLUSION

From the analysis of various Error Detections and Correction codes we had chosen SOS for Parallel FFT.becuase of most of the error corections codes utlize more area from the original module.which is cause the memory overhead.but SOS utilize less area than any other module. In terms of error protection, fault injection experiments show that the ECC scheme can recover all the errors that are out of the tolerance range. The fault coverage for the parity-SOS scheme and the parity-SOS-ECC scheme is ~99.9% when the tolerance level for SOS check is 1.For multiple error detections the DMC based codes is cheked,but in the parrallel FFT design we were considered single bit error.Here we concentrate the area overhead problem.The experimental results also checked and verified by using Xilinx14.7.

V.REFERENCE

i. *“Single-Error-Correction and Double-Adjacent-Error-Correction Code for Simultaneous Testing of Data Bit and Check Bit Arrays in Memories”by Sanguhn Cha (2014 march)*

ii. *“Fault Tolerant Parallel FFTs Using Error Correction Codes and Parseval Checks.”by Zhen Gao (aug 2015)*

iii. *“Mitigation of Hardware Failures, Soft Errors, and Electro-Magnetic Disturbances N. Kanekawa, et all Uematsu,, USA: Springer-Verlag, 2010.*

iv. *“Soft N-modular redundancy,” IEEE Trans. Comput., E. P. Kim and N. R. Shanbhag, Mar. 2012.*

v. *“Current and future challenges in radiation effects on CMOS electronics,P. E. Dodd, IEEE Trans.Aug. 2010.*

vi. *, “Bridging concurrent and non-concurrent error detection in FIR filters,” T. Hitana Nov. 2004,*

vii. *“Energy-efficient soft error-tolerant digital signal processing,”B. Shim and N IEEE Trans. Very Large Scale Integer. (VLSI) Syst, Apr. 2006.*

viii. *“Enhanced Memory Reliability Against Multiple Cell Upsets Using Decimal Matrix Code” by Jing Guo,jan 2014.*

ix. *“Impact of scaling on neutron induced soft error in SRAMs from an 250 nm to a 22 nm design rule E. Ibe,” IEEE Trans. Electron Jul. 2010.*

x. *“Improved decoding algorithm for high reliable reed muller coding,”C. Argyrides in Proc. IEEE Int. Syst. On Chip Conf., Sep. 2007,*