ATLANTIS PRESS

Advances in Computer Science Research, volume 92

International Conference on Mathematics, Big Data Analysis and Simulation and Modeling (MBDASM 2019)

Design of External Memory Error Detection and Correction and Automatic Write-back

Jianlei Wei

Research and Development Dept .ORANGE Technology (Tianjin) CO.LTD, Tianjin, China

***Abstract*—For space application, in order to improve the reliability of memory operation, the SEC-DED (40, 32) Hamming code is used for single event flip (SEU) fault protection of external memory for 8-bit width parity memory. Based on the (39,32) Hisao code, a parity bit is added to minimize the error correction probability of 3-bit error. By reusing the encoder and decoder, EDAC circuit area is reduced by 16.5%. By modifying Finite State Machine (FSM) of the memory controller, the faults detection and correction, and automatic write-back function of the corrected data are all complemented. Experimental results indicate that all 1 bit faults can be auto corrected and don't affect the program results; all 2 bits faults can be detected, and the access memory operation can be terminated in time and take error detection trap. By adding fault tolerant design, read operation would have performance loss, but the SEU failure rate is lower by 6 orders of magnitude.**

Keywords—Single Event Upset (SEU); external memory; memory controlle; Error Detection and Correction (EDAC) code; radiation hazards

I. INTRODUCTION

Reliability is the most important problem in space application. In high-scale technology, the probability of proton induced multi bit flip (MBU) increases, because the device size is very small relative to the particle event track size. In order to reduce MBU, cell interleaving technology [1,2] is adopted. All fault bits of MBU come from different words, so sec ded Hamming code [3] can be used. Since the parity memory has 8-bit width, we use (40, 32) Hamming code to prevent SEU failure. Based on the (39,32) Hisao code, a parity bit is added to minimize the error correction probability of 3-bit error. The encoder and decoder are reused because accessing memory operation is serial; EDAC circuit area is reduced by 16.5%. By modifying FSM of memory controller, the faults detection and correction, and automatic write-back function of the corrected data are all complemented. Experimental results are provided showing that all 1 bit faults can be auto corrected and don't affect the program results; all 2 bits faults can be detected, and the access memory operation can be terminated in time and take error detection trap. By adding fault tolerant design, read operation would have performance loss, However, the failure rate of SEU is 6 orders of magnitude lower, and higher reliability is of great value to high safety system.

II. FAULT TOLERANCE SCHEME

Due to external memory having a great size and memory content can't being recurrent, the adopted scheme should ensure that adding area overhead for check-bits isn't too large, and should have error correction ability. Therefore, a SEC-DED (40, 32) Hamming code is adopted; the adding parity memory area is only 25%. To implement the SEC-DED ability for 32-bit data, at least 7 bits checksum are needed. To minimizing the hardware area and speed, Hisao calculated some best matrixes [4]. We choose such a matrix G7*32:

$$\begin{pmatrix} 1&0&0&0&1&0&1&0&1&0&0&0&0&0&1&0&0&0&0&0&1&1&1&1&0&0&0&1&1&0&1&1 \\ 0&0&0&1&0&0&0&0&0&0&0&1&1&1&1&1&0&1&1&1&0&0&0&1&0&1&1&0&0&0&0&1 \\ 0&0&0&1&0&1&1&0&1&1&1&1&0&0&0&0&1&0&0&1&0&0&1&0&1&0&1&0&0&1&1&0 \\ 1&1&1&1&1&1&1&1&0&0&0&0&0&0&0&0&1&1&0&1&0&0&1&0&0&0&1&0&0&0&1&0&0 \\ 0&1&1&0&1&1&0&0&1&1&1&1&1&1&1&1&0&0&0&0&1&0&0&0&0&0&0&0&1&0&0&0 \\ 0&0&1&0&0&0&0&1&0&0&1&0&0&1&0&0&1&1&1&1&1&1&1&1&1&1&0&0&1&0&0&0 \\ 1&1&0&0&0&0&0&1&0&1&0&0&1&0&0&0&0&1&0&0&0&0&0&0&1&1&1&1&1&1&1&1 \end{pmatrix}$$

The parity-check H=[G7*32, I7*7], I7*7 is the 7*7 unit matrix. Since the parity memory has an 8-bit width, we can add parity bits. In addition to 1-bit and 2-bit faults, the probability of 3-bit faults should be the highest, and the 8th parity bit hopes that the number of corrections of 3-bit faults will be the least. For all 3-bit faults of (39, 32) small code, there are 5452 bit errors (3-bit faults are regarded as 1-bit faults). In order to reduce the error rate, 5452 equations are formed.

$$x_i^m \oplus x_j^m \oplus x_k^m \oplus x_l^m = 1 \text{ ; } 0\delta\iota, \varphi, \kappa, \lambda\delta 38, \mu=1,2,..,5452$$

x_i^m, x_j^m, x_k^m and x_l^m indicate whether the 8th parity bit verify the corresponding component of 32-bit data, due to the check-bit not verifying other check-bit, when $32 \le p \le 38$, $x_p^m = 0$. For example, if the 0th, The first and fourth components of 32-bit data are scrambled once, and the (39, 32) Xiao code scheme thinks that the third component of the check bit is wrong, and the equation is $x0 \oplus x1 \oplus x4 \oplus x35=1$ is formed, and x35=0. Therefore, the 8th check-bit should verify (one of) the former 3 components, the aim is, for the 0th ,1st , 4th and 35th columns of H matrix, any 3 columns' xor different from the spare column, namely miscoding doesn't occur. Any 3 bits are upset synchronously, the (39,32) Hsiao code can take it for the spare bit is erroneous, the 3/4 of the 5452 equations are reduplicate,

so we can search a vector x=(x0, x1,…, x31) from the following equation group:

$$x_i^m \oplus x_j^m \oplus x_k^m \oplus x_l^m = 1;\ 0 \le i < j < k < l \le 38,\ m=1,2,\ldots,1363 \quad (1.1)$$

$$x_p^m = 0;\ 32 \le p \le 38 \quad (1.2)$$

$$\sum_{q=0}^{31} x_q \le 14 \quad (1.3)$$

$$x_t = 0/1;\ t=0,1,\ldots,38 \quad (1.4)$$

This vector should meet the 1363 equations to the best of its abilities; the inequation (1.3) shows that the number of 1 of the increased 8th row is not greater than that of other rows of the H matrix; the aim is the increased 8th checksum doesn't add the EDAC circuit's critical path delay. We obtain a optimal solution x=(0,0,0,0,0,0,0,1,0,1,1,0,1,1,1,1,1,1,0,0,0,0,1,0,1,1,0,1,0,1,0,1,0,1,0), it can satisfy 726 equations, so the generating matrix of (40, 32) Hamming code is [G7*32T,xT]T.

For the (39,32) Hsiao code, in the all 3-bit faults (C_{39}^3=9139), there are 5452 faults regarded as 1-bit faults, 3687 faults can be detected, if we also consider the 3-bits faults, the SEU failure rate is

$$P_{39} = 1-[(1-p)^{39} + C_{39}^1 p(1-p)^{38} + C_{39}^2 p^2(1-p)^{37} + 3687 p^3(1-p)^{36}]$$

The p is the SEU failure rate of the one bit data. In Leo, the magnitude of SEU failure efficiency is 10-7 errors / (bit · day), which can be increased in some abnormal areas, here p=10-6error/(bit·day) is selected, so P39=6.55e-15 error/(bit·day).

TABLE I. EDAC CAPABILITY ON MEMORIES

Memory regions	size	Bit width	EDAC protected
PROM	512MB	8bit	yes
		16bit	yes
		32bit	yes
IO	512MB	8bit	No
		16bit	No
		32bit	No
SRAM	1GB	32bit	yes

For the (40, 32) Hamming code, in the all 3-bit faults (C_{40}^3=9880), 7332 can be detected, so the SEU failure rate is:

$$P_{40} = 1-[(1-p)^{40} + C_{40}^1 p(1-p)^{39} + C_{40}^2 p^2(1-p)^{38} + 7332 p^3(1-p)^{37}]$$

P40=3.77e-15 error/(bit·day). P39 is greater than P40, therefore, in this paper, the SEC-DED (40, 32) Hamming code is adopted.

Table 1 shows the EDAC protection regions. Due to instructions being 32 bits, whether the memory width of PROM is 8/16/32 bits, the accessing type is only 32 bit word, so every word uses 8-bit checksum in this design. If fault tolerant design is not added PPROM=1-(1-p) 134217728*32=1.00 error/(bit·day), PSRAM=1-(1-p)268435456*32=1.00 error/(bit·day). By adopting the EDAC scheme, PPROM_EDAC=1-(1-P40)134217728=1.49e-06 error/(bit·day) and PSRAM_EDAC=1-(1-P40)268435456=2.98-06error/(bit·day), the SEU failure rate is lower by 6 orders of magnitude.

III. EDAC CIRCUIT REUSING STRUCTURE

Figure 1 shows the EDAC circuit integrated structure. The memory controller is integrated in the AHB (Advanced High-performance Bus), when there is a writing operation from the AHB bus, the written data Hwdata is encoded and the checksum is obtained, then they are written synchronously to the data memory and parity memory respectively. When a byte or half word is written to SRAM, the RMW (read-modify-write) technique needs to be used, because SRAM only support 32-bit width memories. When there is a read operation, the data and the check-bits are read synchronously. For the 8/16-bit width PROM, the data memory need to be read times without number and a whole 32-bit instruction is made up. The read date and check-bits are all sent to the decoding logic, the corrected data is returned to the AHB bus. The encoder and decoder can be reused since accessing memory operation is serial, only one of the encoder and decoder is used in any time, and data need to be recoded in the decoding process, by comparing new checksum with that from parity memory, error detection and correction can be implemented. The encoding logic and detecting and correcting error circuit have respective enable signals. If there is a writing operation, only encoding logic is enabled. If there is a read operation, the two enable signals are all valid, for one thing, data from data memory is sent to the encoding logic, the new checksum is attained, and then they are sent to detecting and correcting error circuit, finally the corrected data is returned. By reusing encoder and decoder, EDAC circuit area is reduced 1242.496774um2 (using SIMC 130nm standard CMOS process to synthesize), it is about 16.5% of the total EDAC circuit area.

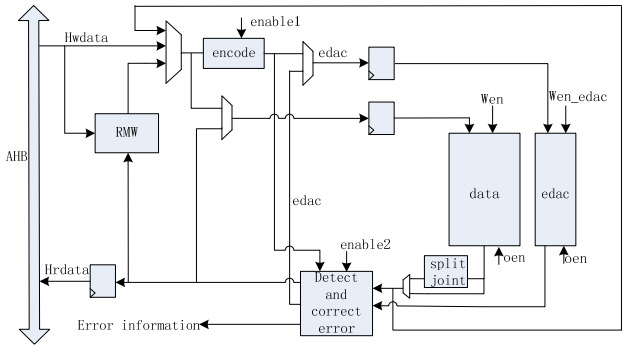

FIGURE I. EDAC CIRCUIT INTEGRATED STRUCTURE

IV. MEMORY CONTROLLER FSM DESIGN

When data has a correctable fault, this fault can be corrected directly, the data returning to AHB bus is accurate, but the data in the memory is still erroneous. To avoid faults accumulation, the erroneous data in the memory should be corrected in time. To ensure the minimum performance loss, the hardware automatic write-back mechanism is adopted; processor doesn't participate in the error correction process. If data has a correctable fault, after the corrected data and checksum are written-back to memory, this operation is just completed.

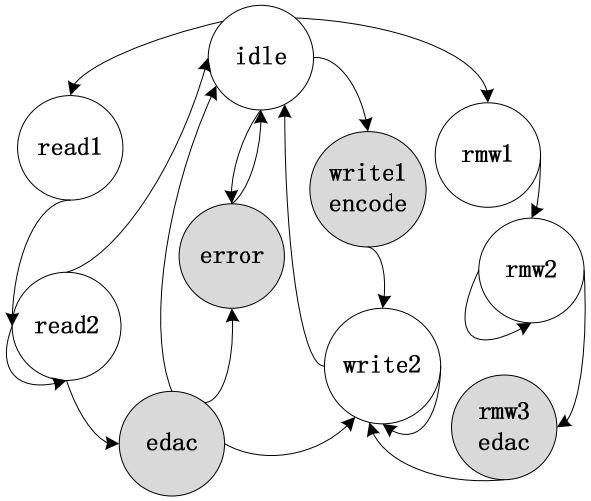

FIGURE II. 32-BIT DATA WIDTH TIMING FSM

A. EDAC Control for 32-bit Width Memory

Figure 2 shows timing control of accessing 32-bit width memory. In the idle state, whether processor has a request is still detected. When there is a writing operation, switch to write1, the address and data are given, chip selection signals valid. The next cycle, switch to write2. In write2 state, write enable signal is valid until this operation completed. So encoding need to be implemented in the write1 cycle, the data and checksum are prepared in this cycle, there is not performance loss. When there is a read operation, switch to read1, the address is given, chip selection signal and read enable one are valid. After one cycle, switching to read2, the data of the external memory is locked. To protect external storage, an EDAC state is added. In EDAC state, if the decoder results in no error, the data will be propagated to the processor. If a correctable error occurs, switch to write2, the corrected data will be written back to memory, and the data sent to the processor is correct. If there are detectable errors, switch to the error state and the memory error information will be propagated to the processor.

When there is a writing byte/half word operation for SRAM, switch to rmw1, the written data is locked, chip selection signal and read enable signal are valid, switch to rmw2, the data from SRAM is read, read enable signal is closed, switch to rmw3, the 32-bit data is pieced together, then reusing write2 state, data is written to SRAM. To guarantee the correctness of the pieced together data, firstly the data from SRAM must be decoded, the decoding logic is executed in rmw3 state, secondly the corrected data is pieced together, finally the pieced data is encoded, and namely the encoding logic is executed in rmw3 state too. By reusing write2 state, the final data is written back.

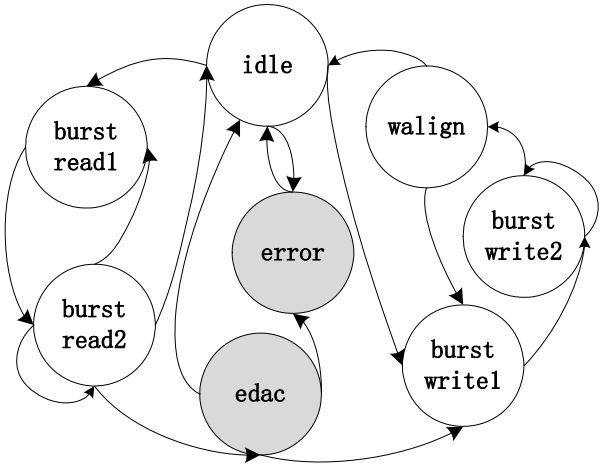

FIGURE III. 8/16-BIT DATA WIDTH TIMING FSM

B. EDAC Control for 8/16-bit Width Memory

Figure 3 shows timing control of accessing 8/16-bit width memory. PROM use to save instructions, so writing PROM is forbidden. When there is a writing operation to IO memory, switch to burstwrite1, the address and data are given, chip selection signal is valid. In burstwrite2 state, the corresponding byte/half word is written to IO memory, then switch to walign, if the written data width is greater than memory width, the new written data and address will be generated, switch to burstwrite1 again until written operation is completed. When there is a read operation, switch to burstread1, the address is given, chip selection signal and read enable one are valid. After one cycle, switch to burstread2, the data from external memory will be locked. If the read data width is greater than memory width, switch to burstread1 again until read operation is completed. After the whole instruction word and the checksum

are read, by reusing eadc state, protecting PROM against SEU error is complemented.

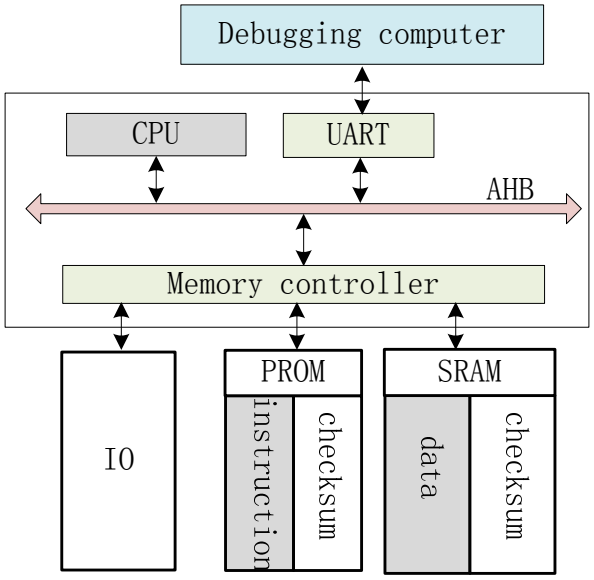

FIGURE IV. THE PROGRAMS EXECUTION TIME COMPARISON

V. EXPERIMENTAL RESULTS ANALYSES

To verify design validity, we must be able to inject any fault into data and/or parity memory, figure 4 shows the fault injection environment architecture. Debugging host computer accesses external memory by UART. Due to the AHB bus data width being 32 bits, data and checksum can't be accessed at one time, the configurable EDAC test register is added, and it has 3 bits: 'debug', 'data_test' and 'edac_test'. If faults are injected into memory, the 'debug' bit is valid. If data memory is injected faults, the 'data_test' bit is reset, if parity memory is injected faults, the 'edac_test' bit is reset at one time. So data memory and parity one can use the same physical address, the address assignment is solved.

During our experiments, we use the program which adds two 3x3 matrices. Table 2 shows the experimental results. For case 1 and 2, no faults injected into PROM and SRAM, if EDAC function is enabled, every read access will take more one cycle to decode, so the execution cycles of the case 2 are more than that of the case 1. For case 3 and 4, all addition instructions in PROM and part data in SRAM are injected 1 bit-flip faults, if EDAC function is disabled, the result will be erroneous, otherwise, the execution cycles will be increased, but the result will be correct, and by reading the fault data/instructions in the debug mode, we can find these faults have been resumed. For case 5 and 6, all addition instructions in PROM and part data in SRAM are injected 2 bit-flip faults, the instructions become multiplication instructions. If EDAC function is disabled, the result will be erroneous, otherwise, the error detection trap will be take, and some software techniques [5] will be used to resume the faults. Therefore, by adopting EDAC design and automatic error correction mechanism, the system dependability is improved.

TABLE II. EXPERIMENTAL RESULTS

case	Fault injection	EDAC function	cycle	results
1	No	disable	2993	correct
2	No	enable	3197	correct
3	1 bit flip	disable	2993	error
4	1 bit-flip	enable	3485	correct
5	2 bit-flip	disable	3038	error
6	2 bit-flip	enable	322	trap

VI. CONCLUSIONS

In order to improve the reliability of memory operation, this paper adopts a sec ded (40, 32) Hamming code scheme to protect the external memory from SEU errors. Due to the 8-bit width of the parity memory, this scheme is based on the (39,32) Hisao code, adding a parity bit to minimize the probability of 3-bit error being corrected. By reusing encoder and decoder, the EDAC circuit area is reduced by 16.5%. By modifying FSM of the memory controller, the faults detection and correction, and automatic write-back function of the corrected data are all complemented. By adding fault tolerant design, read operation would have performance loss, but the failure rate of SEU is reduced by six orders of magnitude, the higher reliability is valuable for high-safety system.

ACKNOWLEDGMENT

We would like to thank XXX for support and encouragement during this work.

REFERENCES

[1] A. D. Tipton, J.A. Pellish, R.A. Reed, R.D. Schrimpf, R.A. Weller, M.H. Mendenhall et al (2006) Multiple-bit upset in 130nm CMOS technology. IEEE Trans. On Nucl. Sci., 53:3259-3264.

[2] H. J. Tausch (2009) Simplified birthday statistics and hamming EDAC. IEEE Trans. On Nucl. Sci., 56:474-478.

[3] J.Gaisler (2002) A portable and fault-tolerant microprocessor based on the SPARC v8 architecture. In Dependable Systems and Networks. DSN 2002. Proceedings. International Conference on, 409-415.

[4] M.Y. Hsiao. (1970) A class of optimal minimum odd-weight-column SEC-DED codes. IBM J. Res. Develop., 14:395-401.

[5] X.Li, Y.Hu, L. Zhang (2011) Digital integrated circuit fault tolerant design. Beijing: Science Press.