

Performance Comparison of Real Time EDAC Systems for Applications On-Board Small Satellites

Y. Bentoutou

Abstract—On-board Error Detection and Correction (EDAC) devices aim to secure data transmitted between the central processing unit (CPU) of a satellite onboard computer and its local memory. This paper presents a comparison of the performance of four low complexity EDAC techniques for application in Random Access Memories (RAMs) on-board small satellites. The performance of a newly proposed EDAC architecture is measured and compared with three different EDAC strategies, using the same FPGA technology. A statistical analysis of single-event upset (SEU) and multiple-bit upset (MBU) activity in commercial memories onboard Alsat-1 is given for a period of 8 years

Keywords—Error Detection and Correction; On-board computer; small satellite missions

I. INTRODUCTION

ERROR Detection and Correction (EDAC) devices on-board satellites aim to secure data transmitted between the central processing unit (CPU) of a satellite onboard computer and its local memory [1]-[5]. In space applications it is well known that in Low Earth Orbits (LEO), stored digital data suffers from Single Event Upsets (SEU). These upsets are induced naturally by radiation. Bit-flips caused by SEU are a well-known problem in memory chips and error detection and correction techniques have been an effective solution to this problem [1]-[5]. For the transmission of secure data between the CPU of the on board computer and its local RAM, the program memory on small satellites have generally made use of Hamming (12,8) code EDAC, or have used TMR, which is a hardware implementation that includes replicated memory circuits and voting logic to detect and correct a faulty value [1]-[3]. These error correction techniques are well-known and widely used. The Hamming (12,8) code allows the correction of a single bit per word. For computers on board a satellite, there is however a definite risk of two or more bit errors occurring within one byte of stored data; either from the impact of a particularly energetic event, or from a second SEU creating a second error [1]-[5]. As a result, later SSTL satellites made use of the TMR voting EDAC technique, which can correct any number of bit-flips in a single byte. This scheme offers excellent protection against the effects of SEUs and single-byte multiple-bit upsets (MBUs) but it cannot correct severe errors, i.e., bit-errors affecting the same logical bit position in more than one of the three physical memory locations. Another disadvantage of TMR is the large memory overhead of 200% in terms of storage [3]. In recent work, a new real-time, low-complexity codec has been described and implemented in FPGA technology for application to small satellite solid-state data recorders, such as the one operating on-board Alsat-1[1].

Another powerful EDAC architecture has been developed recently for implementation in future Algerian Earth Observation satellites, such as the Alsat-1 in order to reduce the total number of SEU in LEO [3]. This paper aims to present a comparison of the newly developed architecture of an on-board EDAC device for future Earth observation small satellite missions with three different EDAC methods, and their implementation in FPGA technology. This paper presents also a statistical analysis of SEU and MBU activity in commercial memories onboard Alsat-1 for a period of 8 years. The in-orbit observations show that the typical SEU rate at Alsat-1's orbit is 4.04×10^{-7} SEU/bit/day, where 98.55% of these SEUs cause single-bit errors, 1.27% cause double-byte errors, and the remaining SEUs result in multiple-bit and severe errors.

II. SMALL SATELLITE MISSIONS

Earth observation missions using small satellites in LEO provide fast and cheap access to space [6]. To make a start-up in space technologies in Algeria, the National Center for Space Technology (CNTS) has initiated a small satellite project named Alsat-1. Alsat-1 is the first step in CNTS's plan to develop Algeria's national space infrastructure [7]. It is part of a wider international collaboration to launch the first disaster monitoring constellation of Earth observation satellites. The primary goal of the mission is to provide daily imaging worldwide for the monitoring and mitigation of natural and man-made disasters as well as dynamic Earth observation. Alsat-1 was launched into a 686 km sun-synchronous orbit in November 2002. This microsatellite carries specially-designed Earth imaging cameras which provide 32-meter resolution imaging in 3 spectral bands (green, red, and near infrared) with an extremely wide imaging swath of 600 km on the ground that enables a revisit of the same area anywhere in the world at least every 4 days with just a single satellite. Images are stored in an 8-Gbit Solid State Data Recorder (SSDR) for high capacity onboard storage of image data, and transmission to the ground is via an 8-Mbps S-band downlink [3]. The Alsat-1 main On-Board Computer (OBC) is an Intel 80C386EX based system that was designed, built and tested at Surrey Satellite Technology Limited (SSTL), a company owned by EADS-Astrium at the University of Surrey in Guildford, UK. The OBC plays a dual role for Alsat-1, acting as the key component of the payload computer as well as the command and control computer for the microsatellite. It has also been adopted by several other satellite projects [3]. Fig. 1 shows the system block-diagram of the OBC. It is a general purpose computer for space applications. It features 4 Mbytes of program memory which is protected by error detection and correction (EDAC) and 32 Kbytes of firmware storage EPROM. The EDAC is

Y. Bentoutou is with Centre for Space Technology, CTS Arzew, Algeria (e-mail: bentyou@yahoo.com)

implemented using Triple Modular Redundancy (TMR) with voting logic, requiring 12 Mbytes in total. The board also supports multiple data inputs and can store a maximum of 128 Mbytes of data. This memory is normally referred to as the Ramdisk. Compared to the program memory there is no hardware protection on the Ramdisk, instead a double-error correcting modified Reed-Solomon RS (256,252) code is used [4]. The encoding and decoding process is executed entirely in software. Data are transferred in 256-byte blocks, which comprise 252 bytes of information, three RS code bytes and one byte is used to give the capability of double byte correction. The RS (256,252) code can correct any number of bit-errors affecting no more than two bytes per block. Thus, three or more bytes would have to be affected to cause a "severe" error under this scheme [4].

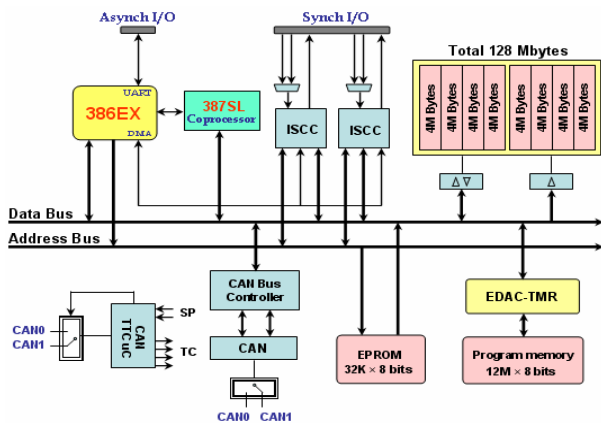


Fig. 1 Block diagram of the Alsat-1 primary OBC

III. UPSET DETECTION AND CORRECTION

Since Alsat-1 is placed in a LEO of 686 km, elevated levels of radiation caused by the lack of atmosphere increase the probability of SEUs. A SEU is a non-destructive error which usually affects logic cells in such a way that it can cause a bit in a memory device to change logic states. This phenomenon is caused by a false charge created by the transit of a single ionizing particle through a memory chip. There are various methods of error detection and correction, but the most commonly used solutions are based on Hamming and Reed-Solomon codes, and TMR.

A. TMR Based EDAC

A well-known technique for providing tolerance against single hardware component failures is triplication of the component, called triple modular redundancy. It is a method of protecting a program memory from both hard and soft failures. The implementation involves using three identical program memories containing the same data. The output from each memory is then processed by a voter circuit, which outputs the majority of the three inputs. As long as two of the memories are functioning properly, the output will be valid [1]-[3]. The integrity of the program memory is important for proper operation of the OBC. A single-bit error in one memory location can cause the 80C386 to enter an endless

loop or corrupt a command to an actuator. For the transmission of secure data between the CPU of the OBC and its local RAM, the memory is protected by TMR. The EDAC is implemented in two Actel A1020B 2000 gate FPGA as shown in Fig. 2. The disadvantages of the TMR-based EDAC are the large memory overhead of three times the required memory and the large number of I/O pins required on the EDAC logic. The 4 Mbytes of memory requires an additional 8 Mbytes to implement the TMR. Each individual memory bank of the TMR can be accessed by the CPU. This allows the TMR to be tested and exercised while in orbit. The individual banks can be accessed by changing two bits of the 386EX parallel port.

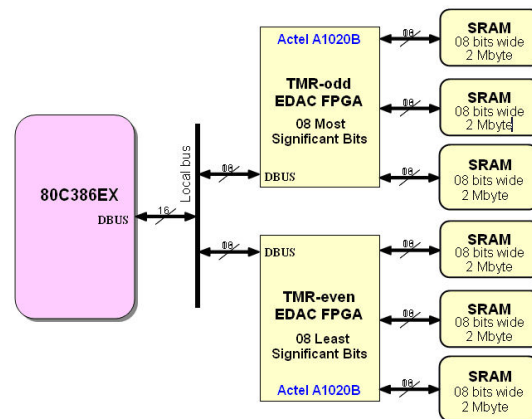


Fig. 2 Block diagram of the TMR based EDAC Device and its memory

B. New proposed EDAC Systems for applications on-board small satellites

In recent work, new real-time, low-complexity EDAC systems have been described and implemented in FPGA technology for application to small satellite computers, such as the one operating on-board Alsat-1 [1]-[3]. These EDAC devices are based on the quasi-cyclic codes, which are a generalization of cyclic codes [8]. The reason for choosing the quasi-cyclic EDAC scheme was for the error detecting and correcting capabilities. The proposed EDAC circuits based on the quasi-cyclic code is needed for computers on board a satellite when there is a definite risk of severe errors, i.e., bit-errors affecting the same logical bit position in more than one of the physical memory locations and which cannot be corrected by TMR. The overall system cost is a 100% increase in stored data, which is significantly smaller than when using TMR in a comparable context (the overall system cost is a 200% increase in stored data when using TMR). The increase in delay time, from implementing the quasi-cyclic code in a typical application, is small. Fig. 3 shows the block diagram of the quasi-cyclic EDAC. The implementation of the quasi-cyclic EDAC method is described in detail in [2]. The EDAC device, when encoding, uses the quasi-cyclic codec to read an 8-bit data vector m from the 386EX microcontroller to generate a parallel 8-bit parity vector P . The data vector is stored unaltered in RAM (referred to as the

data memory). Simultaneously the encoder stores the parity vector into parallel RAM (referred to as the parity memory). The initial 4 Mbytes of data memory requires an additional 4 Mbytes of parity memory to implement the quasi-cyclic EDAC [2]. The EDAC device, when decoding, uses the quasi-cyclic codec to read an 8-bit data vector m and an 8-bit parity vector from the memory to generate the corrected 8-bit data vector and the corresponding error vector.

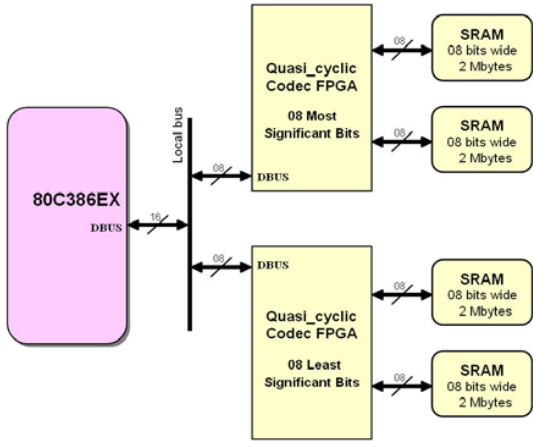


Fig. 3 Block diagram of the Quasi-cyclic based EDAC Device and its memory

A very robust hybrid scheme of TMR coupled with quasi-cyclic checking has been recently proposed in [3]. The main contribution of the presented Hybrid EDAC architecture is firstly to prevent SEU in the FPGA implementing a yet known EDAC architecture by triplication of this function into one FPGA, and secondly to prevent FPGA failure by triplication of the FPGA itself. The memory array is duplicated to prevent SRAM failure. Fig. 4 shows the block diagram of the hybrid EDAC scheme. The implementation of the method is described in detail in [3]. Implementation of these EDAC architectures is transparent to the computer: there are no interrupts and no additional computation [1]-[3].

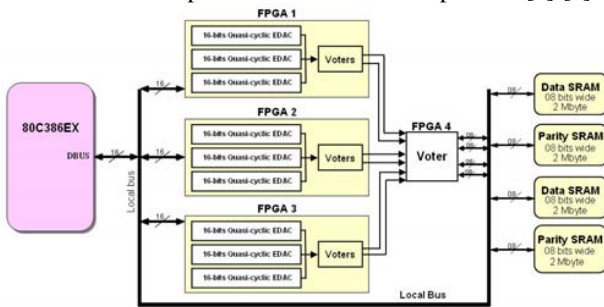


Fig. 4 Block diagram of the Hybrid EDAC device with its memory

C. Single event upsets on-board Alsat-1 OBC

Recently, in-orbit observations of SEU and MBU in the SRAM memories of the OBC for the Alsat-1 in a LEO of 686 km have been presented in [9] from November 2002 to June

2007. In this paper, the period of the in-orbit observations has been extended to include the Alsat-1 life-time (i.e. from November 29 2002 to August 14 2010). The Ramdisk memories are based on eight Samsung SYS 84000 parts of 4 Mbytes each in size. In Fig. 5 the yearly averages of the number of SEU/day and single-bit errors/day are plotted for the period from 29/11/2002 to 14/08/2010. This figure shows the increase of the upset rates with the increasing time. During this period, in 2622 days of observation, 265649 errors were logged in the memory devices, giving a mean error rate of 4.04×10^{-7} error/bit/day. In 2002 and 2009, the average SEU rate is 41 SEU/day and 119 SEU/day, respectively.

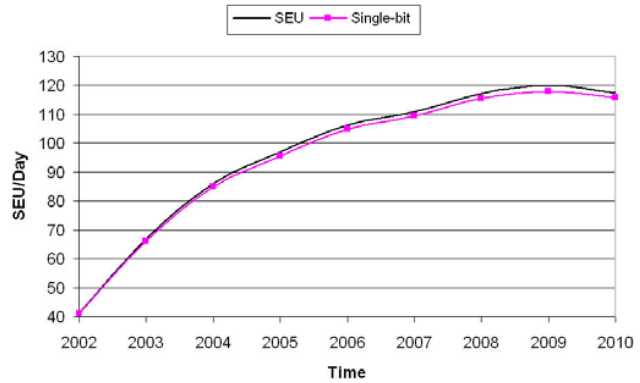


Fig. 5 Plot of the yearly averages of the number of SEU/day and single-bit errors/day versus time from November 2002 to August 2010

IV. FPGA IMPLEMENTATION OF THE EDAC ARCHITECTURES

The design of the two newly proposed EDAC systems with the classical ones that are based on TMR and the Hamming code, was implemented in VHDL. A powerful FPGA design and development software “Actel Libero” was used for compilation, fitting, and simulation of the design in an Actel FPGA. The design has been verified both functionally and with the timing model generated when fitting in a FPGA. A fault injection system described in VHDL was specifically developed to test and validate the proposed techniques. The fault injection system is able to randomly choose the instant of insertion of the fault, the faulty bit position, and the faulty memory chip. It is also able to inject the exact Alsat-1 memory faults in-orbit. The verification of the design was achieved by individual testing of smaller logical blocks of the design. Functional test benches were developed for each encoder/decoder block, the error detection block, and the chip select control block. The testability of the devices was achieved by instantiating the EDAC designs with an SRAM model into the top test bench, and by generating random test data (8 bits for each memory chip) with known error location and magnitude. A software version of the EDAC codes was written in Matlab to generate and write test data and error information in two separate files. The test bench reads the test data file and asserts the values from this file at appropriate times onto the EDAC data bus. The error file is used to

compare the location and magnitude of errors generated by the EDAC design with those generated by using Matlab.

V. COMPARISON OF THE PERFORMANCE OF EDAC SYSTEMS FOR FUTURE EARTH OBSERVATION SMALL SATELLITE MISSIONS IN LEO

In this section, the performance of the proposed EDAC circuits is measured by comparing between the delay times in the four different EDAC devices using the same FPGA technology. The first EDAC is based on the Hamming code, the second one is based on TMR technique, the third one is based on the quasi-cyclic code, and the fourth one is the proposed one (hybrid architecture). On this basis for the Actel A54SX08, a typical delay in encoding is significantly greater (2 ns for the TMR FPGA to 10 ns for the Hamming EDAC to 12 ns for the quasi-cyclic EDAC and 15 ns for the proposed EDAC); while the delay in decoding increases from 10 ns to 36 ns. For the OBC operation, this is not a significant increase in overhead, compared to a relatively long access time (typically 100 ns) of the usual low-power memory employed. Table I shows synthesis results (module count usage) after running the design through the place and root software.

TABLE I
DESIGN SYNTHESIS RESULTS (SED: SINGLE ERROR DETECTION, SEC: SINGLE ERROR CORRECTION, DED: DOUBLE ERROR DETECTION, DEC: DOUBLE ERROR CORRECTION, MED: MULTIPLE ERROR DETECTION, MEC: MULTIPLE ERROR CORRECTION)

EDAC Type	Hamming	TMR	Quasi-cyclic	The proposed Hybrid EDAC
Number of FPGAs	2	2	2	4
Total cells in each FPGA	25%	12%	27%	90%
Encoding delay	10 ns	02 ns	12 ns	15 ns
Decoding delay	16 ns	10 ns	26 ns	36 ns
Memory overhead	50%	200%	100%	100%
EDAC Capability	SEC-SED in a 12-bit word	MEC-MED	DEC-DED in a 16-bit word	DEC-DED in a 16-bit word -Module and FPGA failure detection and correction.
Severe errors correction	No	No	Yes	Yes
Corrected in-orbit errors	98.6%	98.7%	100%	100%

Experiments in the proposed EDAC have shown that 100% of the injected in-orbit faults can be detected and corrected. The proposed Hybrid EDAC based on TMR and the quasi-cyclic EDAC is needed for computers on board a satellite when there is a definite risk of two error bits occurring at the same logical bit position in two physical memory locations (containing data and parity codes), and which cannot be corrected by TMR and the Hamming (12,8) code. Implementation of this EDAC is transparent to the computer: there are no interrupts and no additional

computation. The overall system cost is a 100% increase in stored data, which is significantly smaller than when using TMR in a comparable context (the overall system cost is a 200% increase in stored data when using TMR). The increase in delay time, from implementing the proposed Hybrid EDAC code in a typical application, is small.

VI. CONCLUSION

This paper has presented a performance comparison of four EDAC systems for use on-board Earth observation small satellites. In-orbit observations of single and multiple event upsets are presented for the period between November 2002 and August 2010. In this period of time the mean SEU rate at Alsat-1's orbit is 4.04×10^{-7} SEU/bit/day, where 98.6% of these SEU cause single-bit errors, 1.21% cause double-byte errors, and the remaining SEU result in multiple-bit and severe errors. Experimental results show that the proposed powerful and efficient Quasi-cyclic and hybrid error correcting techniques are sufficient to handle the typical SEU rate at this LEO environment. The EDAC devices are implemented in FPGA technology. The hybrid EDAC is based on the combination of TMR and Quasi-cyclic EDAC techniques for the routine error protection of SRAM program memory for satellites in LEO. The proposed EDAC approaches were validated by using a fault injection procedure developed in VHDL. Experiments in the proposed EDAC devices have shown that 100% of the faults can be detected and corrected.

REFERENCES

- [1] Y. Bentoutou, "A Real Time Low Complexity Codec for use in Low Earth Orbit Small Satellite Missions", *IEEE Trans. Nucl. Sci.*, vol. 53, no. 3, part 1, pp. 1022-1027, 2006.
- [2] Y. Bentoutou, M. Djaifri, A.M. Si Mohammed, "Design and Implementation of a quasi-cyclic Codec for Random Access Memories on board Alsat-1", *Acta Astronautica*, Elsevier Science, vol. 66, no. 5-6, pages: 954-961, March/April 2010.
- [3] Y. Bentoutou, "A Real Time EDAC System for Applications On-Board Earth Observation Small Satellites", accepted for publication in *IEEE Transactions on Aerospace and Electronic Systems*, In press 2011.
- [4] M.S. Hodgart, "Efficient Coding and Error Monitoring for Spacecraft Digital Memory", *Int. J. Electronics*, vol. 73, no. 1, pp. 1-36, 1992.
- [5] M.S. Hodgart and H. Tiggeler, "A (16,8) Error Correcting Code (t=2) for Critical Memory Applications", in *proc. Data Systems In Aerospace*, Montreal, Canada, May 22-26, 2000.
- [6] A. Wicks, A. da Silva-Curiel, J. Ward, and M. Fouquet, "Advancing Small Satellite Earth Observation: Operational Spacecraft, Planned Missions and Future Concepts", in *Proc. 14th Annual AIAA/USU Conference on Small Satellites*, Logan, UT, Aug. 21-24, 2000.
- [7] M. Bekhti, M. Sweeting, and W. Sun, "Alsat-1: The first step into space for Algeria", in *Proc. 53rd IAC and World Space Congress*, Houston, TX, 2002, pp. 10-19.
- [8] R.L. Townsend and E.J. Weldon, "Self-orthogonal quasicyclic codes", *IEEE Trans. Inform. Theory*, vol. IT-13, no. 2, pp. 183-195, Apr. 1967.
- [9] Y. Bentoutou and M. Djaifri, "Observations of Single-Event Upsets and Multiple-Bit Upsets in Random Access Memories On-Board the Algerian Satellite", in *2008 IEEE Nuclear Science Symposium Conference Record*, Edited by Paul Sellin, Oct. 2008, pp. 2568-2570.