A Fault-tolerant 176 Gbit Solid State Mass Memory Architecture

G.C. Cardarilli, P. Marinucci[*], M. Ottavi[*], A. Salsano
Department of Electronic Engineering University of Rome "Tor Vergata", Italy
[*]*Consortium ULISSE*
g.cardarilli@ieee.org
panfilo@ing.univaq.it
marco.ottavi@libero.it
salsano@ing.uniroma2.it

Abstract

This paper presents a new Solid State Mass Memory (SSMM) suitable for space applications. The memory reliability is increased by using two different approaches. Firstly, memory mass fault-tolerance, with respect to hard failures, is obtained by using a fine-granularity hierarchical structure with a certain level of redundancy. A second strategy used for facing soft errors is based on Error Correction Codes (ECC) and periodic memory washing. A performance index has been developed for evaluating the main parameters of the SSMM architecture. This index takes into account the ECC capability, the memory weight and reliability, allowing to relate them to the required overhead.

1. Introduction

The design of a SSMM for space applications depends on the level of reliability and security required. The reliability is mainly related to the capability of memory of storing a minimum data quantity after a certain working time, while data security is related to assuring data integrity after hard- or soft-errors. Typically, design of SSMM is performed treating separately the reliability and data integrity requirements. Reliability increasing is faced introducing some degree of redundancy in the architecture, data integrity is obtained introducing suitable algorithms for error detection and correction - normally based on ECC.

Unfortunately, the definition of the architectural parameters of a fault-tolerant SSMM with a given level of reliability and security is very difficult. This difficulty is related to interdependency of the different design parameters and the required performance. For this reason the authors developed a specific design methodology [1,2,3]. In particular, paper [3] shows a decision model for designing memory modules considering the design constraints and the effects of the hard errors (as chip failures) and the soft ones (as Single Event Upsets).

In this paper we show the SSMM resulting from the above analysis. With respect to the previous work, in this paper we specifically study the effects of using a high reliability communication system -used for the connection of the SSMM to the various subsystems present aboard- on the SSMM performance. In fact, our architecture is based on a switching system that connects the instruments and the central processing unit to the memory modules. On this architecture has been performed a reliability analysis that takes into account the overall characteristics of the SSMM structure. In particular, section 2 shows the general structure of the SSMM. Section 3 defines the structure of the reliability tree of the proposed architecture and the effects this structure on the SSMM performance. Section 4 describes the system that controls

the I/O interfaces and switching matrix. Finally, in section 5 we shall discuss the structure of the memory module deriving from the methodology shown in [2,3].

2. Memory Architecture

In [4] the authors define a new class of SSMM architectures based on a communication system making use of a switching matrix. Starting from this proposal we defined the structure shown in fig 1. Two main blocks realize the SSMM.

- System Control Unit (SCU),
- Memory Kernel Unit (MKU).

The SCU performs the supervision of the memory and controls the connection with the ground stations, interfacing the SSMM with the board computer. The MKU is realized by a set of Independent Memory Array Modules (IMAM) and by a connection system based on a Switching Matrix (SM). Moreover, the SM is interfaced to the board instrumentation by a set of I/O interfaces. The memory modules are independent, in fact, each module contains all the functions for the detection and the correction of the errors as well as the functions required for the data transfer toward the I/O interfaces.

Through the SM, data are transferred from the board instruments to the memory modules and from the memory modules to the output interfaces, which are connected to transmission system.

3. Reliability structure

Space application requires systems that must satisfy two basic requirements
1. Small probability of failure propagation,
2. Working probability at the mission end of life.

The second requirement is frequently satisfied introducing a suitable level of system redundancy. Instead, the first requirement is faced dividing the system in a number of redundancy levels in order to insulate any failure and to decrease the overall required redundancy, for a given reliability value (Figure 2).

The definition of the reliability levels (or the failure tree) has been done considering the structure shown in fig. 1, see paper [5]. We can define the following levels.

Level 1 – This level considers the detection and the correction of the errors for the *word*.

Level 2 – This level considers the reconfiguration of the *word groups*. A *word group* is a subset of memory devices that store a whole word.

Level 3 – This level considers the reconfiguration of the *partitions*. A *partition* is constituted by a set of word groups, by the drivers and the protection system introduced for limiting the short circuit propagation – as those generate by latch-up or other failures.

Level 4 – This level concerns the reconfiguration of the *modules*. A *module* corresponds to a set of partitions and the interfacing circuits. It corresponds to a single board with the interfaces and the control system.

Level 5 – This last level is related to the reconfiguration of the *control system*. The *control system* includes commutation (switch matrix) and supervision (microcontroller) subsystems. Reconfiguration operations allow limiting the failure propagation of single interfaces to whole system.

Levels from 1 to 4 form the SSMM kernel. Figure 2 considers that each partition is electrically independent from each other. This allows restricting the impact of faults inside a single partition. In fact, the presence of additional drivers as well as the introduction of a control system for the power supply allows limiting of the failure propagation. The partition reconfiguration procedure, foreseen in Level 3, turns off the faulty partition, removes the

corresponding word group addresses and turns on the spare partition. Similar reconfiguration procedures are present in the Levels 4 and 5.

4. I/O Interfaces and Switching Matrix

Main elements of the SSMM are the I/O interfaces and the switching matrix. They are responsible for the connection between the serial links -coming from the measurement instrumentation- and the memories. They must have the following characteristics

*   Fault-tolerant with respect to a given fault set.
*   High transfers speed and low latency.
*   Flexible with respect to requirements of system reconfiguration.
    The main tasks that must be performed are
*   Flow control for a connection between input and output interfaces. The flow control mainly corresponds to the generation and control of the handshaking signals.
*   Access arbiter for an output resource shared by several input links. In this case the system must control the actual configuration of switches present in switching matrix. This is a static configuration.
*   Management of the file system. A microcontroller dynamically define the route on the switching matrix to connect the input interface to the output one for a given file number.

4.1. Crossbar Switch Matrix

Figure 4 shows the block scheme of the switching matrix. It allows the interconnection among M input and N output interfaces. Since for the specific application interconnections between two input interfaces -as well as between two output interfaces- are not required, the switch matrix shall implement only $M*N$ connections.

The fundamental characteristics of the switching matrix are the following.

*   The connections with the memories are half-duplex. Indeed, for any working time the memory can be alternatively in read or in write phase.
*   In a scientific satellite, the most of the input links access the memory in write phase.
*   Inside the matrix, data transfer is controlled by the flow control procedures, and eventually is arbitrated by the arbiter unit.

The use of half-duplex links allows the introduction of a single physical link. Each physical link is realized with a channel for data and another channel for the flow control.

With this structure, the failure of a connection does not cause the failure of the whole memory system, only the specific node cannot access to the matrix. Moreover, the memory partitioning allows the concurrent access of several users, maintaining high access and transfer speeds. A connection system implemented using a switch matrix allows multiple parallel links between users and resources. This architectural redundancy increases the reliability of the system, in fact the failure of a link implies the loss of a part of the whole functionality of the system.

Moreover, the use of a microcontroller (SCU) can reduce the impact of a failure operating two subsequent steps:

*   Change of traffic routing within the matrix to recover part of system functionality
*   Start of procedures of test and recovery

In a link we can distinguish a chain of three functional blocks (Figure 5): *MEM I/F, Switch, LINK I/F,* we will now describe the behavior of the system in case of failures occurring on one of these three devices:

1. *Failure of a MEM I/F:* microcontroller reconfigures routing toward other memory modules preserving data memorization coming from connected link. Data memorized in the memory module are not available. Microcontroller starts the operations of diagnostic and recovery.

2. *Failure of a LINK I/F*: loss of data coming from the connected apparatus. The microcontroller routes towards an other escape the operations of reading in memory and starts the operations of diagnostic and recovery.

3. *Failure of a switch (S.M)*: system has minor flexibility not being able to connect the link i-th with the memory block j-th. The microcontroller modifies the routing of the traffic inside of the switching matrix.

The introduction of cool redundancy (1:1) based on the duplication of the modules seen above would increase the overall system reliability. The switch between the work module and the spare module is operated by SCU (Figure 6).

4.2. Serial Link Interfaces

The serial links are based on space version -named *Spacewire*- of the protocol IEEE 1355. This protocol introduces some levels of fault detection on the serial links using parity encoding for the data.

There are two different types of interfaces toward the serial links.

- Unidirectional interfaces. These interfaces are normally used for writing in the memories.
- Bi-directional interfaces. They read and write the memories.

Most of the interfaces will be used as unidirectional ones. They correspond to the links leading the measurement information. A small number of interfaces require reading and writing the memories. The most important interface connects the memories to the telemetry circuitry. This circuit transmits collected information from the satellite to the earth station.

4.2.1. Unidirectional Interfaces

The basic element of the interfacing devices is the unidirectional interface. The structure of the unidirectional interface is shown in Fig. 7. The main blocks are the following.

1. **LVDS I/F**. This block realizes the electric interfacing between the differential signals LVDS (Low Voltage Differential Signaling) and Data and Strobe single ended signals.
2. **DS I/F**. This interface interprets the serial signal DS extracts the clock signal and translates it in a parallel word. It implements the flow and the parity flow following the procedures of protocol IEEE 1355.
3. **IN FIFO**. The input FIFO is written with the parallelized tokens coming from the serial IEEE 1355 links. The FIFO depth is chosen in order to avoid data loss for the latency of the successive elements. Since serial link can reach 200Mbps, the FIFO speed is up to 20 Mtps (Mega tokens per second, with average 10 bits per token).
4. **IN LINK I/F**. The input packets are read from the input FIFO. Packets have 1 data byte and 1 flag bit -that indicates if the packet is a header or a payload. The interface reads the packet header -corresponding to the file number- and the flag bit – that indicates read or write request.

4.2.2. Bi-directional Interfaces

Using the half-duplex transmission mode, bi-directional interface is equivalent to a unidirectional interface that can work, alternatively, in read or write mode. Figure 8 shows the two working modes. For each serial interface we can also work in full duplex mode. In this case, we use two matrix inputs for a single full duplex transmission. This mode of operation introduces a fault-tolerance capability in the system. In fact, if one of the two links fails, the other link can be used in half duplex mode.

The differences between unidirectional and bi-directional interfaces are in the interfacing with the switching matrix. In the latter case, the interfaces must allow the reading from the memories. In the following we shortly describe the blocks needed for this interface.

1. **IN LINK I/F**. Is similar to the IN I/F used in the unidirectional interface. It is associated to a single user channel.

2. **OUT LINK I/F**. It generates the handshaking signals for controlling the data flow from the switching matrix to the memories.

3. **I/O LINK I/F**. This block merges the functionality of the previous blocks. It is used for the half duplex communication. It also performs some arbiter functions for the management of the shared resources.

Similar interfaces are present for interfacing the switching matrix to the memory modules (**I/O MEM I/F**).

5. Structure of the memory modules.

Memory modules are the core of SSMM. Their structure is shown in Figure 7. The main characteristics required to such modules concern the capacity, the organization and the memory package. The capacity of each module should be sufficiently large but the physical volume required by the memory chips limits it –a memory module corresponds to a single board. In the memory, a very high level of complexity characterizes module. In fact, the required memory capacity implies the use of a large number of memory chips. For this application the use of space qualified chips is unsuitable for two reasons. The first reason is related to the cost of space qualified chips that, due to the large number, would make the cost of the SSMM prohibitively high. The second reason is the unavailability of space qualified memory chips with a sufficiently level of integration. All the above reasons push toward the use of Commercial off-the-shelf (COTS) chips. Unfortunately, these chips are not protected against the effects induced by spatial environment and the requirements for certain level of fault-tolerance imply the introduction of suitable design strategies.

The characteristics required for the SSMM are the following

Net capacity 16 Gbit/module (BOL).

Memory organization: array of r rows and c columns. The actual values of these two parameters must be chosen for obtaining a square board -in order to simplify the mechanical organization and the shielding of the SSMM.

Suitable packaging of the memory chips. The chosen package must allow the control of the power supply for the single chip (we preferred a package containing 4 chips of 64 MBIT SDRAM).

The memory architecture uses, for the word group, k chips for data storing, $n-k$ chips for storing the check symbols, and s chips for the cold spare. Each of the above chips belongs to a different package -this choice avoids that a package fault could produce multiple symbol error.

Each word group is arranged on 4 rows and each row is composed by $(n+s)/4$ packages (corresponding to $(n+s)/4$ columns). Figure 10 shows the organization of the memory board. This board organization allows reconfiguring the memory module with different ECC structures, depending on the application considered. In fact, it is possible to reduce the codeword down to $1/4\ n$. Increasing the memory washing frequency, it is also possible to reduce the code length maintaining the overall BER. Of course, this solution reduces the memory availability.

Using the optimization tools presented in [2,3], we have defined the code parameters. Figures 11 and 12 show the input parameters and the applied constraints, and the optimization results, respectively. The obtained code has the structure *(55, 45, 8)*. For practical reasons, we have chosen the code parameters *(56, 44, 8)* that results in a memory size (BOL) of *22 Gbytes*. An interesting result of the optimization phase is the absence of any cold spare. In fact, the

optimization tool increased the code redundancy, using a part of the redundancy for the counterbalancing of the permanent faults of the memory chips.

7. Conclusions.

In this paper we show a new architecture of solid state mass memory for satellite applications. It is based on a switching block and a set of memory modules. All the presented devices are designed in order to obtain the requested value of reliability for the overall system.

References

G.C. Cardarilli, P. Marinucci, A. Salsano, "Fault-tolerant Solid State Mass Memory for Satellite Applications", IEEE Instrumentation and Measurement Conference, St. Paul, Minnesota, USA, May 18-21, 1998TC'98.

G.C. Cardarilli, P. Marinucci, S. Bertazzoni, M. Salmeri, A. Salsano, "Design of Fault-tolerant Solid State Mass Memory", DFT'99, Albuquerque, NM, October 1999.

G.C. Cardarilli, P. Marinucci, A. Salsano, " Development of an Evaluation Model for the Design of Fault-tolerant Solid State Mass Memory ", accepted for the presentation to ISCAS 2000, Geneva, Switzerland.

M.P. Kluth, F. Simon, J.Y. Le Gall, E. Muller, "Design of a fault tolerant 100 Gbits solid-state mass memory for satellites", VLSI Test Symposium, 1996., Proceedings of 14th , 1996.

Fichna, T.; Gartner, M.; Gliem, F.; Rombeck, F. "Fault-tolerance of spaceborne semiconductor mass memories", Fault-Tolerant Computing, 1998. Digest of Papers. Twenty-Eighth Annual International Symposium on , 1998 .
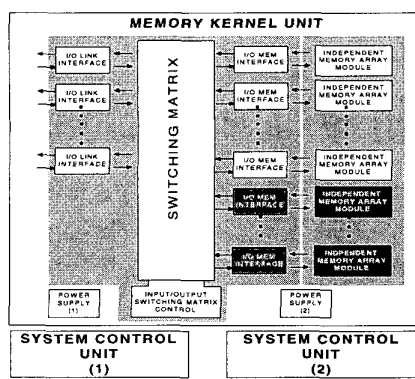
fig.1 – Global Architecture of SSMM
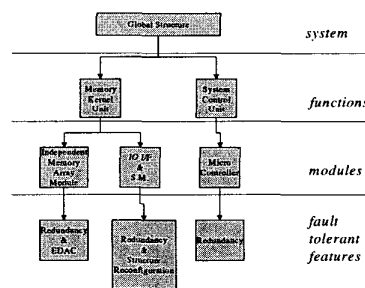
fig.2 - SSMM fault tolerant features
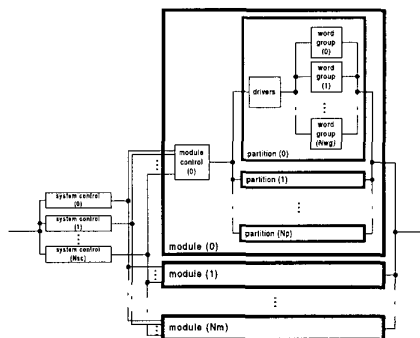
fig3 – Reliability structure of fault-tolerant semiconductor mass memory
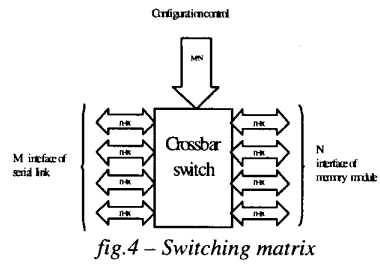
fig.4 – Switching matrix
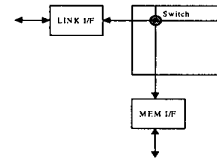
fig. 5 connection chain
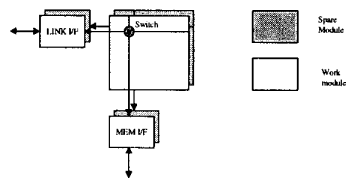
fig.6 connection chain with 1:1 redundancy
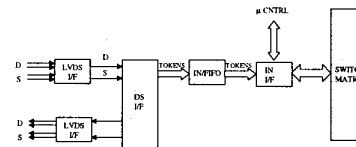
fig.7 – Unidirectional serial interface

(a)

(b)
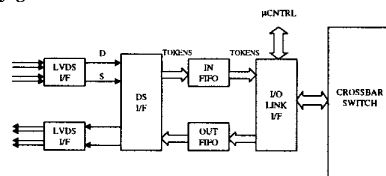
fig.8 – a) half-duplex bi-directional serial interface
b) full-duplex bi-directional serial interface
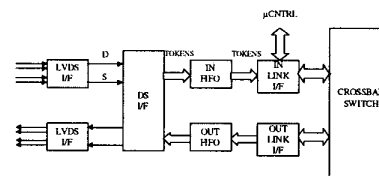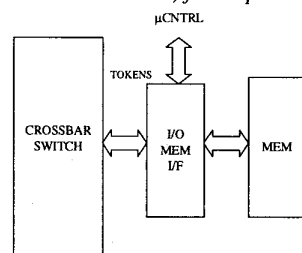
fig.9– interface to memory module
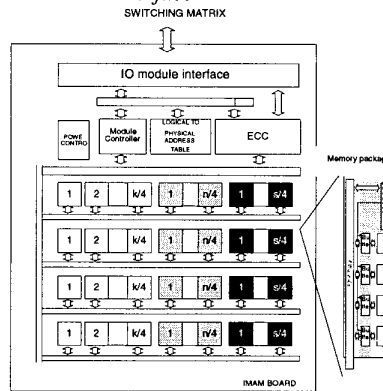
fig.10 – memory module layout

fig.11- Input parameters and constraints

fig.12 - Optimization results