Reliability of Single-Error Correction Protected Memories

Juan Antonio Maestro, *Member, IEEE*, and Pedro Reviriego, *Member, IEEE*

***Abstract*—Reliability is a critical factor for systems operating in radiation environments. Among the different components in a system, memories are one of the parts most sensitive to soft errors due to their relatively large area. Due to their large cost, traditional techniques like Triple Modular Redundancy are not used to protect memories. A typical approach is to apply Error Correction Codes to correct single errors, and detect double errors. This type of codes, for example those based on Hamming, provides an initial level of protection. Detected single errors are usually corrected using scrubbing, by which the memory positions are periodically re-written after a fixed (deterministic scrubbing), or variable period (probabilistic scrubbing). These traditional models usually offer good results when calculating the reliability of memories (e.g. through the Mean Time To Failure). However, there are some particularities that are not modeled through these approaches, to the best of our knowledge. One of these particularities is how double errors are handled. In a traditional approach, two errors in the same word produce always a system failure (only single errors can be corrected). However, if the two (or more) errors affect the same bit, either the second one reinforces the first one (keeping just a single error), or corrects it. In both scenarios, the resulting situation does not trigger a system failure, which has a direct impact on the reliability of the memory. In this paper, traditional reliability models are refined to handle the mentioned scenarios, which produces a more precise analysis in the calculation of mean time to failure for memory systems.**

***Index Terms*—Error correction codes, memory, reliability, single event upsets (SEU).**

ACRONYM[1]

SEU	Single Event Upset
MBU	Multiple Bit Upset
SEC	Single Error Correction
DED	Double Error Detection
METF	Mean Events to Failure
MTTF	Mean Time to Failure

NOTATION

N	number of bits per word (width)
M	number of memory words (size)
λ	event arrival rate per word
R(t)	reliability of a memory in t seconds
$\mathrm{r_r(t)}$	reliability of a single memory word in t seconds
$\mathrm{P_f}$	probability of failure in a scrubbing interval
$\mathrm{P_{nf}}$	$1 - \mathrm{P_f}$; probability of not failing in a scrubbing interval
$\mathrm{P_a(i)}$	probability of i events arriving in the same scrubbing interval
$\mathrm{P_{fa}(i)}$	probability that the i events arriving in the same scrubbing interval do not affect the same bit $P_{fa}(i) = 1 - (1/N)^{j-1}$
$\mathrm{P_f}$'(i)	probability of failure upon the i event
$\mathrm{P_f(i)}$	probability of failure for i events, assuming that no failure has been produced in the $i-1$ previous events $P_f(i) = P'_f(i) \cdot \prod_{j=2}^{i-1}(1 - P'_f(j))$
$\mathrm{P_m(i)}$	probability of i events affecting bits already altered by single event upsets
$\mathrm{t_s}$	scrubbing period

I. INTRODUCTION

WHEN a digital circuit operates in a radiation environment, several problems can arise, due to the presence of high-charged particles. A usual effect is obtained through Single Event Upsets (SEU), which produce soft errors in the system [1]–[5].

This result is particularly visible in hostile environments where there are physical phenomena that affect semiconductors in a negative way. Radiation [6]–[8] is one of these factors, and its influence in errors has been reported many times [1], [2]. Radioactive material is used in medical and military industries, but more research is being devoted to this problem in space applications [9]. Space applications are especially critical, because systems are not easily accessible, and therefore errors may produce the complete failure of a mission [10]–[12].

Among the different modules that can be found in digital circuits, memories are usually most affected by SEU [13]–[19]. Errors in memories usually affect a single cell, but they may also produce more important problems, as a row/column failure (if the error strikes on the row/column selector), or even a whole chip failure (if the error strikes on the chip selector). Memories are usually protected by codes that can correct single errors, and detect double ones in a given word, known as SEC-DED. This may be achieved by using redundancy, for example Hamming codes [20], [21]. With this, an extra cost overhead is introduced, which is the one associated to the redundant bits. Besides, adding extra hardware increases the probabilities of suf-

Manuscript received October 30, 2007; revised July 12, 2008; accepted July 20, 2008. First published November 07, 2008; current version published March 04, 2009. This work was supported by the Spanish Ministry of Science and Education under Grant ESP-2006-04163. Associate Editor: L. Cui.

The authors are with Universidad Antonio de Nebrija, 28040 Madrid, Spain (e-mail: jmaestro@nebrija.es; previrie@nebrija.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TR.2008.2006470

[1]The singular and plural of an acronym are always spelled the same.

fering a SEU, because the number of storage elements is also higher. Once an error has been detected using these redundant codes, a usual method to get rid of the error is to use scrubbing [22], [23]. This solution implies rewriting each memory word periodically, thus restoring its correct value. Scrubbing can be performed in a deterministic, or a probabilistic way. Deterministic scrubbing is achieved when the word checking is produced through a cyclic, constant process. Probabilistic scrubbing is obtained when this checking happens whenever a word is read by the program in execution.

Several studies have been conducted to assess the reliability of memories that use these protection mechanisms in the presence of soft-errors [23]–[27]. Most of the reported analyses of soft errors in memories are based on the usual reliability model. In this way, the Mean Time To Failure (MTTF) is used to assess the system reliability in an intuitive way [24]. However, all these studies assume that two consecutive errors on the same memory word would produce a double error (detectable but not correctable), and therefore a memory failure. But this is not certain because those two consecutive errors may in fact affect the same bit on that word, and therefore in this particular case two situations may happen: i) the second error keeps (enforces) the first one, causing a single-error situation (which is correctable); or ii) the second error flips back the initial error to its right value, causing a no-error situation. Whether the second error reinforces the first error or corrects it depends on the physical nature of the device, and the phenomenon that induces those errors.

The first situation has been described in [25], but it has not been totally developed due to the complexity of the presented formulation. The second situation, however, has not been addressed in literature, to the best of the authors' knowledge. Therefore, existing analyses do not consider the impact of those effects on memory reliability.

In this paper, previous analyses will be extended by incorporating the mentioned effects, and therefore extending the scope of these models. In Section II, the proposed models that include refinements to the traditional reliability approaches are presented for a single register first, and then for a whole memory system. The reliability of the proposed models is studied both when scrubbing is not used, and when it is used. The results in terms of the derived reliability indexes are summarized in Section III. Then, in Section IV, a wide set of simulation results are presented to validate the applicability of the proposed models, and to illustrate the accuracy of the proposed approximations. Finally, some conclusions are presented in Section V.

II. Proposed Refinements to the Soft Error Model

The following assumptions will be taken into account, which are also followed in the related literature [23]–[26].

- Soft errors arrive to the system following a Poisson process [28]. This is a typical assumption because this distribution is representative of real radiation environments.
- Soft errors on each memory cell are independent. Events that have happened in a certain part of the memory do not have an influence on the probability of error in other areas of the system.
- Soft errors are uniformly distributed across all cells. There are no areas in the memory where errors are more likely to happen.
- SEC-DED codes are used to protect the memory. In this way, there is a mechanism to detect & correct single errors. Therefore, to produce a failure, two or more errors are necessary in different bits of the same logical word.

Initially, the case of a single register $(\mathrm{M}=1)$ will be studied to properly model the new failure scenarios. Then the results are extended to a whole memory system.

A. Single Register $(M = 1)$

1) Nonscrubbing Case: The first step should be to calculate the MTTF as a statistic describing the reliability, $R(t)$.

The commonly accepted interpretation of reliability would dictate that a system has been reliable in time t if during that interval no errors or just one error has happened (as it can be corrected with the SEC-DED mechanism). Applying the Poisson formulation, this condition would be expressed as

$$R(t) = e^{-\lambda t} + \lambda t e^{-\lambda t} = (1 + \lambda t) e^{-\lambda t} \tag{1}$$

Integrating by parts, the expression used in traditional models is directly obtained as

$$MTTF|_{tradit.} = \frac{2}{\lambda} \tag{2}$$

This result is also easily obtained using the Mean Events To Failure $(METF)$ relationship with the MTTF for Poisson processes [29].

$$MTTF|_{tradit.} = \frac{METF|_{tradit.}}{\lambda} \tag{3}$$

in which the $METF$ is 2 (implying a double error, which is uncorrectable).

The assumption that 2 or more errors would make the system unreliable is in line with traditional approaches. However, the scenarios in which several errors on the same bit either i) make a persistent single error, or ii) correct the error, have to be taken into account to refine the model.

In the following, we study the situation in which several SEU hit the same bit, keeping the single error in it. To model this case, we use induction. First, let us consider the situation where two consecutive SEU happen. The traditional model would be represented by (2), where two events always result in a failure. Now, if the probability of both SEU hitting the same bit is considered, and given that there is no constraint on where the first SEU happens, the probability that the second one occurs in the same bit is $1/N$. Reciprocally, the probability that the second event happens in a different bit (and therefore producing a double error) is $(N-1)/N$. In this way, the probability of failure for two events would be

$$P_f(2) = \frac{N-1}{N} \tag{4}$$

Now, let us consider the case in which i events happen. Again, given that the first one has no constraint on which bit the failure happens, the probability that the next $i-2$ (all but the last one) hit the same bit is $(1/N)^{i-2}$. Finally, to produce the failure situation, the last event should hit a different bit, so that a double error situation is produced. This is represented by the same previous expression, $(N-1)/N$, which makes the probability of failure

$$P_f(i) = \left(\frac{N-1}{N}\right) \cdot \left(\frac{1}{N}\right)^{i-2} \quad (5)$$

This process is cumulative, and therefore the case of $i+1$ events is also similar to the case of i events, and can be derived in an easy way. As the number of events grows, the probability that all of them but the last one hit the same bit becomes lower. This probability decreases in a $(1/N)$ factor, which is a convergent term. Therefore, if the factor $1/N$ is applied to (5), the general term for $i+1$ can be derived as

$$P_f(i+1) = \frac{1}{N} \cdot P_f(i) = \left(\frac{N-1}{N}\right) \cdot \left(\frac{1}{N}\right)^{i-1} \quad (6)$$

Note that, through the inductive process, the probability for $i+1$ events (6) happens to be the probability for i events (5), after replacing i with $i+1$. This makes the derivation coherent, which proves that (5) is the general term of the model. Now, to calculate the $METF$, and because the number of events is not bounded (it can tend to infinite), the following expression is obtained. It represents the weighted probability of error (multiplied by the number of events), which is equivalent to the mean number of events to produce a failure.

$$\begin{aligned} METF|_{cumul.} &= \sum_{i=2}^{\infty} i \cdot P_f(i) = \sum_{i=2}^{\infty} i \cdot \left(\frac{N-1}{N}\right) \cdot \left(\frac{1}{N}\right)^{i-2} \\ &= N \cdot \left(\frac{1}{N} + \frac{1}{N-1}\right) \end{aligned} \quad (7)$$

Combining (3) with (7), the $MTTF$ of the system happens to be (similar expression to the one derived in [25])

$$MTTF|_{cumul.} = \frac{N}{\lambda} \cdot \left(\frac{1}{N} + \frac{1}{N-1}\right) \quad (8)$$

Notice that the $METF$ in this case (7) is greater than the $METF = 2$ for the traditional case. The reason for this is the effect of the cumulative process, which makes the number of events different from (greater than) the number of errors. In other words, several events in the same bit force a single error on that position. This result differs from the traditional model, in which the number of events is always equal to the number of errors induced in the system.

However, from the point of view of the memory, the number of *effective* events is still 2, because those accumulated in the same bits are not *perceived* by the system. These effective events seem to arrive at the memory with a lower rate; or in other words, the *effective* arrival rate, λ', is lower than the actual λ.

If we now apply the traditional model using the effective arrival rate λ', the $MTTF$ could be expressed as (being the effective $METF = 2$)

$$MTTF|_{tradit.}^{\lambda'} = \frac{2}{\lambda'} \quad (9)$$

Because the $MTTF$ in (9) has to be the same as the one calculated with the cumulative model in (8), we can deduce that

$$\begin{aligned} MTTF|_{tradit.}^{\lambda'} &= \frac{2}{\lambda'} = \frac{N}{\lambda} \cdot \left(\frac{1}{N} + \frac{1}{N-1}\right) \\ &= MTTF|_{cumul.}^{\lambda} \end{aligned} \quad (10)$$

To meet equality (10), the effective event arrival rate that the system *perceives* is

$$\lambda' = \lambda \cdot \frac{2}{N \cdot \left(\frac{1}{N} + \frac{1}{N-1}\right)} = \lambda \cdot \frac{2}{1 + \frac{N}{N-1}} \quad (11)$$

The conclusion of this deduction is that, when the cumulative case is considered in the system, the effect is the same as applying the traditional model, but with an arrival rate lower than the actual one. This intuitively produces an increase in the $MTTF$ of the system.

Now, we address the case in which a second error on a bit already affected by an error clears the initial error. The probabilistic analysis is very similar to the previous case, but now only an even number of events have to be taken into account (which is obvious if a second SEU always has to correct a previous one). Therefore, applying this constraint, and rewriting (5), the general term of the probability of failure for this case is

$$P_f(2i) = \left(\frac{N-1}{N}\right) \cdot \left(\frac{1}{N}\right)^{i-1}, \text{ and } P_f(2i+1) = 0 \quad (12)$$

The first part of (12) represents the case in which several pairs of consecutive events hit & correct a single bit; but the last pair hits different bits, producing a double (uncorrectable) error. The second part of this expression, however, represents an odd number of events in which the pairs of consecutive SEU again hit & correct singles bits, except the last one. Because at the end there is only a single error, this is correctable by the SEC codes, and therefore the probability of failure is zero.

If (12) is extended to an unbound number of (even) errors, then the *METF* of the system is (similar to (7))

$$\begin{aligned} METF|_{self.} &= \sum_{i=1}^{\infty} 2i \cdot P_f(2i) \\ &= \sum_{i=1}^{\infty} 2i \cdot \left(\frac{N-1}{N}\right) \cdot \left(\frac{1}{N}\right)^{i-1} \\ &= \frac{2 \cdot N}{N-1} \end{aligned} \quad (13)$$

Note that now i ranges from 1 to infinity because it represents the number of double events, and that is why the $METF$ is calculated using $2i$. It is not possible that an odd number of events can cause a failure. Then, combining (3) with (13), the expression of the $MTTF$ is

$$MTTF|_{self.} = \frac{N}{\lambda} \cdot \frac{2}{N-1} \quad (14)$$

As in the cumulative case, this situation can be seen as the traditional model with an effective rate lower than the actual one. In other words, the number of effective events is lower than actual events.

A similar deduction can be proposed. Considering that the $MTTF$ calculated with the traditional model (9) should be the same as the $MTTF$ calculated with the self-clearing model (14),

$$\begin{aligned} MTTF|_{tradit.}^{\lambda'} &= \frac{2}{\lambda'} \\ &= \frac{N}{\lambda} \cdot \left(\frac{2}{N-1} \right) \\ &= MTTF|_{self.}^{\lambda} \end{aligned} \tag{15}$$

Therefore, the effective arrival rate, λ', would be

$$\lambda' = \lambda \cdot \frac{N-1}{N} \tag{16}$$

When N is not too high, this factor is significant. This dependency with N can be explained as, for lower values of N, the probability of two errors hitting the same bit is larger.

2) Scrubbing Case: As mentioned in the introduction, scrubbing is used to rewrite memory words where an error has been detected (through the redundant bits), thus bringing them back to their correct value. Let us define t_s as the scrubbing period. If we assume that these periods are independent, then the $MTTF$ can be derived, as discussed in the next subsection, from the probabilities of failure, P_f, in a scrubbing period for both the cumulative, and self-clearing models. The probability of j SEU arriving in a scrubbing interval would be given by the Poisson distribution as

$$P_a(j) = \frac{(\lambda \cdot t_s)^j \cdot e^{-\lambda \cdot t_s}}{j!} \tag{17}$$

The probability that all the j SEU hit the same bit, therefore accumulating a persistent, correctable single error, is $(1/N)^{j-1}$. Therefore, the probability that the previous does not happen (which would produce a non-correctable error) is $1-(1/N)^{j-1}$. Weighting (17) with this factor, and adding all the probabilities from the case of two SEU to an infinite number, the probability of failure for the cumulative model would be

$$\begin{aligned} P_f &= \sum_{j=2}^{\infty} P_a(j) \cdot P_{fa}(j) \\ &= \sum_{j=2}^{\infty} P_a(j) \cdot \left(1 - \left(\frac{1}{N}\right)^{j-1}\right) \\ &= \sum_{j=2}^{\infty} \frac{(\lambda \cdot t_s)^j \cdot e^{-\lambda \cdot t_s}}{j!} \cdot \left(1 - \left(\frac{1}{N}\right)^{j-1}\right) \end{aligned} \tag{18}$$

Starting from the same assumptions, and through a similar deductive process, the probability of failure for the self-clearing model would be

$$P_f = e^{-\lambda t_s} \cdot \left[\sum_{j=1}^{\infty} \left(\frac{(\lambda \cdot t_s)^{2 \cdot j}}{(2 \cdot j)!} + \frac{(\lambda \cdot t_s)^{2 \cdot j+1}}{(2 \cdot j+1)!} \right) \cdot \left(1 - \left(\frac{1}{N}\right)^{j}\right) \right] \tag{19}$$

In the next subsection, the $MTTF$ is derived for a whole memory when scrubbing is used based on P_f. The $MTTF$ for $M = 1$ is a particular case of the general result, and therefore it is not considered independently.

B. Memory Systems ($M > 1$)

In the previous subsection, the considerations of cumulative, and self-clearing events have been presented. Both cases have been applied to a single memory word case, i.e. $M = 1$.

In this subsection, these models will be extrapolated to a whole memory system ($M > 1$).

1) Scrubbing Case: Let us start assuming that a scrubbing technique has been implemented. According to the traditional formulation, the $MTTF$ of this case would be

$$MTTF = \int_0^{\infty} r_r(t)^M \cdot dt \tag{20}$$

where $r_r(t)$ is the probability that a single register has not failed at time t. The problem is that, in this case, $r_r(t)$ is represented by a quite complex expression, difficult to handle in an analytical way.

$$\begin{aligned} r_r(t) = & \, r_r(t_s)^{floor\left(\frac{t}{t_s}\right)} \cdot e^{-\lambda \cdot rem(t,t_s)} \\ & \cdot \left(1 + \lambda \cdot rem(t,t_s) + \frac{1}{N} \cdot \frac{(\lambda \cdot rem(t,t_s))^2}{2!} \right. \\ & \left. + \frac{1}{N^2} \cdot \frac{(\lambda \cdot rem(t,t_s))^3}{3!} + \ldots \right) \end{aligned} \tag{21}$$

However, following the approximation described in [23], it can be simplified. The $MTTF$ would be approximated by

$$\begin{aligned} MTTF &= P_{nf}^M \cdot t_s \cdot \left(1 + P_{nf}^M + \left(P_{nf}^M\right)^2 + \left(P_{nf}^M\right)^3 + \ldots\right) \\ &= P_{nf}^M \cdot t_s \cdot \frac{1}{1 - P_{nf}^M} \end{aligned} \tag{22}$$

where P_{nf} is the non failure probability of a single register on a t_s interval, or, in other words, $1 - P_f$.

It can be proved that, if $MTTF \gg t_s$ (which is a reasonable assumption in most cases), the approximation is often good enough. In fact, the error is smaller that t_s as proven in [23]. To properly calculate the $MTTF$ for the cumulative or self-clearing models described in this paper, the suitable P_{nf} has to be used based on (18), and (19). With this consideration, a feasible approach to calculate the probability of failure in memories is obtained, which is valid not only for the traditional approach, but also for the improvements presented in this paper.

Let us now consider that the reliability of the model (for the cumulative case) is

$$\begin{aligned} (r_r(t_s))^M = & \, e^{-\lambda \cdot t_s \cdot M} \cdot \left(1 + \lambda \cdot t_s + \frac{1}{N} \cdot \frac{(\lambda \cdot t_s)^2}{2!} \right. \\ & \left. + \frac{1}{N^2} \cdot \frac{(\lambda \cdot t_s)^3}{3!} + \cdots \right)^M \end{aligned} \tag{23}$$

This implies that the system will not fail if there are zero, or one events; but it will not fail either if a second event hits the bit already affected by the first one (probability $1/N$), or if a third event arrives at the system under the same circumstances (affecting an already altered bit, which has a probability of $1/N^2$), etc.

Expression (23) can be approximated for $\lambda \cdot t_s \cdot M \ll 1$ as

$$\begin{aligned}(r_r(t_s))^M \approx & \left(1 - \lambda \cdot t_s \cdot M + \frac{(\lambda \cdot t_s \cdot M)^2}{2}\right) \\ & \cdot \left(1 + \lambda \cdot t_s \cdot M + M \cdot (M-1) \right. \\ & \left. \cdot \frac{(\lambda \cdot t_s)^2}{2} + M \frac{1}{N} \cdot \frac{(\lambda \cdot t_s)^2}{2}\right) \\ \approx & 1 - \left(\frac{N-1}{N}\right) \cdot M \cdot \frac{(\lambda \cdot t_s)^2}{2} \end{aligned} \quad (24)$$

Because this approximation implies keeping the quadratic term ignoring those with a higher order, the expression is also valid for the self-clearing model (both the cumulative, and self-clearing models coincide up to the quadratic term).

Integrating (24) to find the $MTTF$, we obtain

$$\begin{aligned} MTTF|_{cumul.,self.} \approx & \frac{t_s}{\left(\frac{N-1}{N}\right) \cdot M \cdot \frac{(\lambda \cdot t_s)^2}{2}} \\ = & \frac{2 \cdot N}{M \cdot (N-1) \cdot \lambda^2 \cdot t_s} \end{aligned} \quad (25)$$

The $MTTF$ for the traditional model introduced in [23] is

$$MTTF|_{tradit.} \approx \frac{2}{M \cdot \lambda^2 \cdot t_s} \quad (26)$$

Comparing (25) with (26), the mentioned ratio of $N/(N-1)$ is directly obtained.

Following with the vision presented in previous systems, this can be seen as the memory perceiving a lower number of effective events, or in other words, as if the effective arrival rate λ' is lower:

$$\begin{aligned} MTTF|^{\lambda}_{cumul.,self.} \approx & \frac{2 \cdot N}{M \cdot (N-1) \cdot (\lambda)^2 \cdot t_s} \\ = & \frac{2}{M \cdot \left(\sqrt{\frac{N-1}{N}} \cdot \lambda\right)^2 \cdot t_s} \\ = & \frac{2}{M \cdot (\lambda')^2 \cdot t_s} \\ = & MTTF|^{\lambda'}_{tradit.} \end{aligned} \quad (27)$$

where

$$\lambda' = \lambda \cdot \sqrt{\frac{N-1}{N}} \quad (28)$$

Comparing (28) with (11), and (16), the effect introduced by $M > 1$ can be observed: λ' does not depend linearly on N, but through its square root. This is logical because, when $M > 1$, the probabilities of several events occurring in the same bit decrease, which makes the effective arrival rate higher than in the $M = 1$ case.

2) Nonscrubbing Case: Once the scrubbing case has been studied, the model with no scrubbing will be analyzed next.

When no scrubbing is applied to the memory, the $MTTF$ for the traditional model is given by

$$MTTF = \frac{1}{\lambda \cdot M} \sum_{i=2}^{\infty} \left[i \cdot P'_f(i) \prod_{j=2}^{i-1} \left(1 - P'_f(j)\right) \right] \quad (29)$$

where $\lambda \cdot M$ is the arrival rate extended to the whole memory, and the summation is the $METF$ of the system (the calculation of $MTTF$ is straightforward through (3)). $P'_f(i)$ is the probability of failure upon the arrival of the i-th SEU that for the traditional model is

$$P'_f(i)\big|_{tradit.} = \frac{i-1}{M} \quad (30)$$

This represents the probability that one of the memory words already with an error suffers a new SEU, producing an uncorrectable double error. From (29), we can get an upper bound for the other two failure models by replacing $P'_f(i)$ with

$$P'_f(i)\big|_{cumul.,self.} = \frac{N-1}{N} \cdot \frac{i-1}{M} \quad (31)$$

This implies that, to produce a failure due to a double error, a second event has to occur in the same memory word as a previous one, $(i-1)/M$, but on a different bit, $(N-1)/N$. Otherwise, the single error would persist (cumulative), or would disappear (self-clearing).

This, as it was explained before, can be seen as a lower number of effective events in the new models, which would produce a lower probability of failure. Combining (30) with (31),

$$P'_f(i)\big|_{cumul.,self.} = \frac{N-1}{N} \cdot P'_f(i)\big|_{trad.} \quad (32)$$

Another detail that should be pointed out is that expression (31) has been simplified when using the number of events i. If there have been k previous SEU affecting any erroneous bit, the number of perceived events by the system will be lower than the real one, and therefore the actual number of errors for the cumulative approach would be

$$P'_f(i) = \frac{N-1}{N} \cdot \sum_{k=0}^{i-2} \frac{(i-1)-k}{M} \cdot P_m(k) \quad (33)$$

$P_m(k)$ represents the probability that k events have affected bits previously hit by SEU. This produces $i-1-k$ actual errors in the system (or $i-1-k$ perceived events).

But because these $P_m(k)$ are low, and decrease with a $1/N$ factor, they can be ignored, except for $P_m(0) \approx 1$, reaching expression (31). This is more accurate when $M \gg 1$, as in this case most failures would occur for a large number of events, making the impact of $P_m(k)$ even smaller.

The reasoning for the self-clearing model is similar, but this time, an event on an already affected bit not only would not increase the error count, but it would also reduce it (the error would be corrected). However, the conclusion in this case is similar to the cumulative one: expression (31) is accurate enough for the model in (29).

According to the literature, a different approach to calculate the $MTTF$ is based on an approximation for the traditional model [23], which is accurate only for large values of M:

$$MTTF|_{tradit.} = \frac{1}{\lambda} \cdot \sqrt{\frac{\pi}{2} \cdot \frac{1}{M}} \quad (34)$$

This expression indicates that the probability of failure is directly proportional to the memory size, M. To use this approximation with the cumulative, self-clearing models, the effective probability of failure needs to be scaled according to (31), and (32). In other words, M needs to be divided by an $N/(N-1)$ factor, which is the ratio between the traditional, and new probabilities of failure. In this way, (34) can be modified:

$$MTTF|_{cumul.,self.} = \frac{1}{\lambda} \cdot \sqrt{\frac{p}{2} \cdot \frac{1}{M} \cdot \frac{N}{N-1}} \quad (35)$$

This, again, can be seen as the traditional case with a modified arrival date, λ':

$$MTTF|^{\lambda}_{cumul.,self.} = MTTF|^{\lambda'}_{tradit.} \quad (36)$$

$$\lambda' = \lambda \cdot \sqrt{\frac{N-1}{N}} \quad (37)$$

As expected, the effective arrival rate is lower than the actual one, and it is consistent with the one obtained for the scrubbing case in (28).

Approximation (35), as it was explained for (31), does not reflect the decreasing number of perceived events due to the possibility of several SEU affecting the same bit. In the same way, this effect may be neglected, due to its low impact.

III. Summary of Reliability Indexes

In this section, a summary of the reliability indexes calculated with the improved models are presented. This is a previous step before we present the simulation experiments to compare the traditional with the new reliability indexes.

A. Single Register ($M = 1$)

1) Nonscrubbing:

$$\text{Traditional:} \quad MTTF|_{tradit.} = \frac{2}{\lambda}$$

$$\text{Cumulative:} \quad MTTF|_{cumul.} = \frac{N}{\lambda} \cdot \left(\frac{1}{N} + \frac{1}{N-1}\right)$$

$$\text{Self-clearing:} \quad MTTF|_{self.} = \frac{N}{\lambda} \cdot \frac{2}{N-1}$$

2) Scrubbing:

$$\text{Traditional:} \quad MTTF|_{tradit.} \cong \frac{2}{\lambda^2 \cdot t_s}$$

$$\text{Cumulative:} \quad MTTF|_{cumul.} \cong \frac{2 \cdot N}{(N-1) \cdot (\lambda)^2 \cdot t_s}$$

$$\text{Self-clearing:} \quad MTTF|_{self.} \cong \frac{2 \cdot N}{(N-1) \cdot (\lambda)^2 \cdot t_s}$$

TABLE I
$MTTF$ (in Seconds): Simulated vs. Theoretical for All Models

	Simulated	Theoretical
Traditional (Eq. 2)	2	2
Cumulative (Eq. 8)	2.09	2.09
Self-clearing (Eq. 14)	2.18	2.18

B. Memory Systems ($M > 1$)

1) Nonscrubbing:

$$\text{Traditional:} \quad MTTF|_{tradit.} \cong \frac{1}{\lambda} \cdot \sqrt{\frac{\pi}{2} \cdot \frac{1}{M}}$$

$$\text{Cumulative:} \quad MTTF|_{cumul.} \cong \frac{1}{\lambda} \cdot \sqrt{\frac{\pi}{2} \cdot \frac{1}{M} \cdot \frac{N}{N-1}}$$

$$\text{Self-clearing:} \quad MTTF|_{self.} \cong \frac{1}{\lambda} \cdot \sqrt{\frac{\pi}{2} \cdot \frac{1}{M} \cdot \frac{N}{N-1}}$$

2) Scrubbing:

$$\text{Traditional:} \quad MTTF|_{tradit.} \cong \frac{2}{M \cdot \lambda^2 \cdot t_s}$$

$$\text{Cumulative:} \quad MTTF|_{cumul.} \cong \frac{2 \cdot N}{M \cdot (N-1) \cdot (\lambda)^2 \cdot t_s}$$

$$\text{Self-clearing:} \quad MTTF|_{self.} \cong \frac{2 \cdot N}{M \cdot (N-1) \cdot (\lambda)^2 \cdot t_s}$$

Note that except for the $M = 1$, and nonscrubbing case, the cumulative, and self-clearing models coincide. This is due to the approximations we applied in the model derivation, in which second order effects are discarded. In this situation, both approaches coincide from a macroscopic point of view.

IV. Model Validation

The purpose of this section is to validate the proposed models, and to illustrate the accuracy of the proposed approximations.

For the single register case, when scrubbing is not used, the expressions in (2), (8), and (14) have been checked by simulation. Let us consider $\lambda = 1$, and $N = 12$ (a typical value to protect 8-bit memories). Then, calculating the $MTTF$ according to the previous expressions, the results shown in Table I are obtained.

The results are consistent with the theoretical model, as the $MTTF$ is higher for the case in which successive SEU clear previous errors; while the $MTTF$ is lower for the traditional model, where none of the presented scenarios are taken into account.

For the case of memory systems when scrubbing is used, the theoretical results of the different models (traditional, cumulative, and self-clearing) have been checked through simulation for $\lambda = 1$, $t_s = 0.01$, and $N = 12$. The comparison of the model results vs. simulation is offered in Fig. 1, for the three cases. The first conclusion is that all the theoretical results are very close to the simulated models, which implies that the different approximations are good enough, and have an acceptable

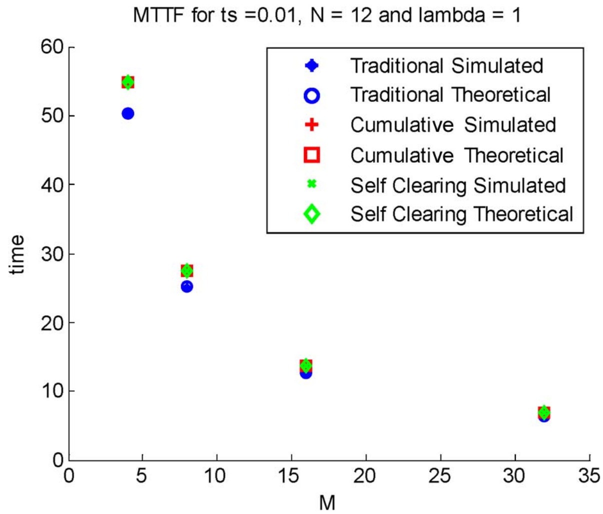

Fig. 1. Traditional, cumulative, and self-clearing models: theoretical results vs. simulated.

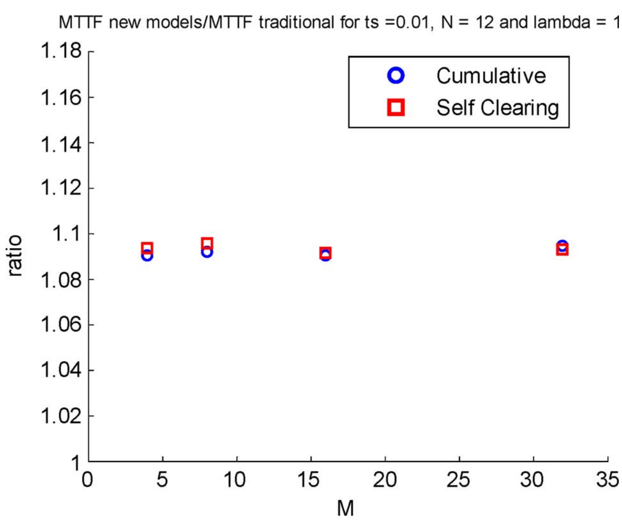

Fig. 2. Ratio of the $MTTF$ of the new models vs. the $MTTF$ of the traditional one.

relative error. Another important point is that both the cumulative, and self-clearing models provide a higher $MTTF$ to the system, or in other words, the reliability (undervalued through the traditional model) is estimated in a more realistic way. Notice that the cumulative, and self-clearing results are very close. This is because, when scrubbing is applied, most of the failures are produced by just two errors in the same t_s period, making the formulation of both models very similar in this case.

From Fig. 1, it seems that the difference between both the cumulative and self-clearing models, compared to the traditional one, decreases as M grows. However, this appearance is not true. The absolute difference does decrease, but not the relative one, which keeps constant as predicted by (25), and (26). In Fig. 2, the ratio $MTTF$ (new models)/$MTTF$ (traditional model) is depicted. It is clearly seen that this ratio (which marks the relative difference between both) always has a value around 1.09, which indicates that the relative difference is 9% approximately. This is consistent with the values predicted by (25), and (26).

TABLE II
$MTTF$ (IN SECONDS). TRADITIONAL MODEL: SIMULATED VS. APPROXIMATION

M	**Simulation**	**Approx. by (29)**
1	1.9978	2.0000
2	1.2515	1.2500
4	0.8061	0.8047
8	0.5305	0.5306
16	0.3567	0.3565
32	0.2429	0.2429
64	0.1673	0.1673

TABLE III
$MTTF$ (IN SECONDS). CUMULATIVE MODEL: SIMULATED VS. APPROXIMATION

M	**Simulation**	**Approx. by (29)**
1	2.0957	2.0833
2	1.3073	1.2934
4	0.8415	0.8324
8	0.5546	0.5502
16	0.3727	0.3704
32	0.2536	0.2528
64	0.1748	0.1742

TABLE IV
$MTTF$ (IN SECONDS). SELF-CLEARING MODEL: SIMULATED VS. APPROXIMATION

M	**Simulation**	**Approx. by (29)**
1	2.1814	2.0833
2	1.3404	1.2934
4	0.8540	0.8324
8	0.5603	0.5502
16	0.3755	0.3704
32	0.2547	0.2528
64	0.1754	0.1742

In the last set of experiments, we consider the case of a memory system when no scrubbing is used. Some simulation results will be offered to compare the estimates given by the proposed approximations with the experimental values. The tables below show the results for the different models obtained by simulation, and by the approximation given by (29) for different values of the memory size. For simplicity, we select an arrival rate of $\lambda = 1$.

From Tables II–IV, it can be seen that the approximation using (29) works well for all failure models, even for very small values (16 or larger) of the memory size. For sizes up to 16, the approximation deviates from the simulation results. One of the factors that contributes to this deviation is the $P_m(k)$ effect mentioned before. These probabilities have a larger impact for a low memory size, and ignoring them introduces an error in the models. This error is greater for the self-clearing case because, in this situation, the error count is more altered than in the other cases.

Some results for the approximations given by (34) and (35) are shown in Table V, where the memory sizes are greater because the approximations only work for large values of M, as mentioned before. In this case, the arrival rate, λ, is 0.1.

From these results, observe that the approximation works well when M is large.

TABLE V
$MTTF$ (IN SECONDS). SIMULATED VS. APPROXIMATION BY EQUATIONS (34), AND (35) FOR ALL THE MODELS

M	1024	2048	4096	8192
Traditional Sim.	0.3981	0.2804	0.1975	0.1394
Traditional Approx. Eq. (29)	0.3982	0.2802	0.1975	0.1393
Traditional Approx. Eq. (34)	0.3917	0.2769	0.1958	0.1385
Cumulative Sim.	0.4161	0.2927	0.2066	0.1455
Cumulative Approx. Eq. (29)	0.4156	0.2925	0.2062	0.1454
Cumulative Approx. Eq. (35)	0.4091	0.2893	0.2045	0.1446
Self-Clearing Sim.	0.4162	0.2927	0.2064	0.1455
Self-Clearing Approx. Eq. (29)	0.4156	0.2925	0.2062	0.1454
Self-Clearing Approx. Eq. (35)	0.4091	0.2893	0.2045	0.1446

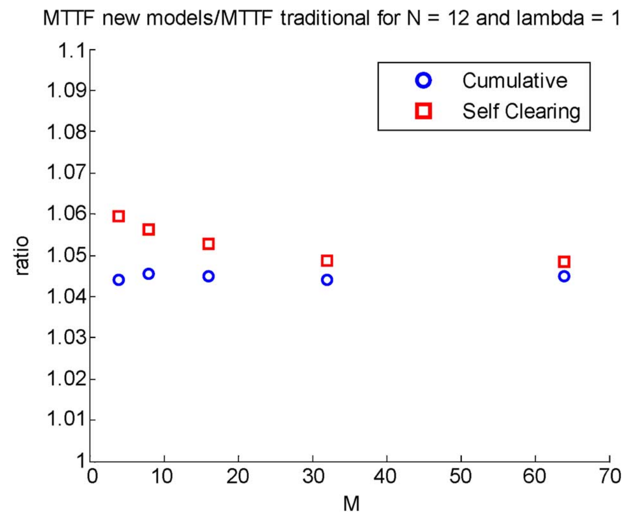

Fig. 3. $MTTF$ ratio (new models/traditional model) for both the cumulative, and self-clearing approaches. M values up to 70.

Finally, in both cases, the ratio of the $MTTF$ of the new models versus the traditional one gives a value close to the one predicted by dividing both approximations, (34), and (35):

$$MTTF(new)/MTTF(traditional) = \sqrt{\frac{N}{N-1}} \tag{38}$$

In our case, this ratio has a value of 1.0445. This is illustrated in Figs. 3, and 4. Notice that, for the smaller values of M (Fig. 3), the value of the ratio is not constant, because the approximations are less accurate in this case. However, when M is large enough (Fig. 4), this ratio remains constant at the mentioned value.

V. CONCLUSIONS

We refined traditional reliability models to handle novel scenarios called *cumulative* (when a previous error is reinforced by a following one), and *self-clearing* (when a previous error is corrected by a following one), which produces a more precise analysis in the $MTTF$ calculation for memory systems.

Considering these cases increases the $MTTF$ of the device. The models have been applied to a whole memory system, first with scrubbing, and finally without it. The obtained $MTTF$ results are more accurate than the ones offered by the traditional model, whose error is not negligible (up to 10% in some cases).

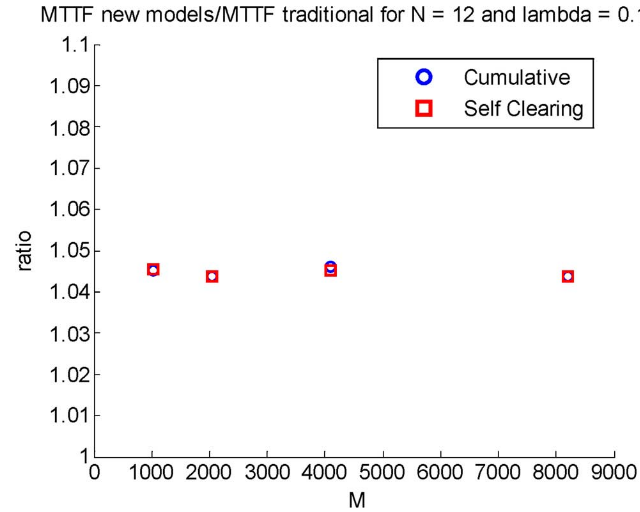

Fig. 4. $MTTF$ ratio (new models/traditional model) for both the cumulative, and self-clearing approaches. M values up to 10000.

Some interesting conclusions are:

- Because, in the cumulative, and self-clearing models, successive SEU may hit the same bit without increasing the count of errors, the number of events (SEU) does not have to be equal to the number of errors, as in the traditional model. In other words, the "perceived" number of events by the memory is lower than in the traditional case, which intuitively increases the $MTTF$. This can also be seen as having an effective arrival rate lower than the actual one.
- The ratio of the $MTTF$ of the new models vs. the traditional one is a constant, and independent of the size of the memory, when the memory is large enough (and the scrubbing period is fast enough, if scrubbing is applied).

REFERENCES

[1] R. D. Schrimpf and D. M. Fleetwood, "Radiation effects and soft errors in integrated circuits and electronic devices," in *World Scientific Publishing*. : , 2004, ISBN: 981-238-940-7.
[2] P. E. Dodd and L. L. Massengill, "Basic mechanisms and modeling of single-event upset in digital microelectronics," *IEEE Trans. Nuclear Science*, vol. 50, no. 3, pp. 583–602, Jun. 2003.
[3] R. C. Baumann, "Soft errors in advanced computer systems," *IEEE Design and Test of Computers*, vol. 22, no. 3, pp. 258–266, May/Jun. 2005.
[4] L. Schiano, M. Ottavi, and F. Lombardi, "Markov models of fault-tolerant memory systems under SEU," in *Records of the 2004 International Workshop on Memory Technology, Design and Testing*, Aug. 2004, pp. 38–43, issue 9-10.
[5] G. C. Cardarilli, A. Leandri, P. Marinucci, M. Ottavi, S. Pontarelli, M. Re, and A. Salsano, "Design of a fault tolerant solid state mass memory," *IEEE Trans. Reliability*, vol. 52, no. 4, pp. 476–491, Dec. 2003.
[6] J. F. Ziegler and W. A. Lanford, "The effect of sea level cosmic rays on electronic devices," *J. Appl. Phys.*, vol. 52, no. 6, pp. 4305–4318, Jan. 1981.
[7] C. A. Gossett, B. W. Hughlock, M. Katoozi, G. S. LaRue, and S. A. Wendler, "Single event phenomena in atmospheric neutron environments," *IEEE Trans. Nuclear Science*, vol. 40, no. 6, pp. 1845–1856, Dec. 1993.
[8] Y. Tosaka, H. Kanata, T. Itakura, and S. Satoh, "Simulation technologies for cosmic ray neutron-induced soft errors: Models and simulation systems," *IEEE Trans. Nuclear Science*, vol. 46, no. 3, pp. 774–779, Jun. 1999.

[9] A. D. Fogle, D. Darling, R. C. Blish, and E. Daszko, “Flash memory under cosmic and alpha irradiation,” *IEEE Trans. Device and Materials Reliability*, vol. 4, no. 3, pp. 371–376, Sept. 2004.

[10] J. E. Vinson, “Circuit reliability of memory cells with SEU protection [for space application],” *IEEE Trans. Nuclear Science*, vol. 39, no. 6, pp. 1671–1678, Dec. 1992.

[11] A. S. Brogna, F. Bigongiari, F. Bertuccelli, W. Errico, S. Giovannetti, E. Pescari, and R. Saletti, “SEU protected CPU for slow control on space vehicles,” *Second IEEE International Workshop on Electronic Design, Test and Applications*, no. 28-30, pp. 422–424, Jan. 2004.

[12] Y. Chen, D. Nguyen, S. Guertin, J. Bernstein, M. White, R. Menke, and S. Kayali, “A reliability evaluation methodology for memory chips for space applications when sample size is small,” in *Integrated Reliability Workshop*, Oct. 2003, vol. 2003, pp. 91–94, issue , 20-23.

[13] T. Sasada, S. Ichikawa, and T. Kanai, “Measurement of single-event effects on a large number of commercial DRAM,” *IEEE Trans. Nuclear Science*, vol. 53, no. 4, pp. 1806–1812, Aug. 2006, part 1.

[14] M. Nicolaidis, “Design for soft error mitigation,” *IEEE Trans. Device and Materials Reliability*, vol. 5, no. 3, pp. 405–418, Sept. 2005.

[15] T. C. May and M. H. Wood, “A new physical mechanism for soft errors in dynamic memories,” in *Proc. 16th Annual Int. Reliability Physics Symp.*, 1978, pp. 33–40.

[16] W. R. McKee, H. P. McAdams, E. B. Smith, J. W. McPherson, J. W. Janzen, J. C. Ondrusek, A. E. Hyslop, D. E. Russell, R. A. Coy, D. W. Bergman, N. Q. Nguyen, T. J. Aton, L. W. Block, and V. C. Huynh, “Cosmic ray neutron induced upsets as a major contributor to the soft error rate of current and future generation DRAMs,” in *34th Annual Proc. IEEE Int. Reliability Physics Symp.*, 1996, pp. 1–6.

[17] F. Vargas and M. Nicolaidis, “SEU tolerant SRAM design based on current monitoring,” in *Proc. 24th IEEE Int. Symp. Fault Tolerant Computing*, Austin, TX, Jun. 1994, pp. 106–115.

[18] E. Normand, “Single event upset at ground level,” *IEEE Trans. Nuclear Science*, vol. 43, no. 6, pp. 2742–2750, Dec. 1996.

[19] D. G. Mavi and P. H. Eaton, “Soft error rate mitigation techniques for modern microcircuits,” in *Proc. 40th Annual Reliability Physics Symp.*, 2002, pp. 216–225.

[20] R. W. Hamming, “Error detecting and correcting codes,” *Bell Syst. Tech. J.*, vol. 29, pp. 147–160, Apr. 1950.

[21] R. Hentschke, F. Marques, F. Lima, L. Carro, A. Susin, and R. Reis, “Analyzing area and performance penalty of protecting different digital modules with Hamming code and triple modular redundancy,” in *15th Symposium on Integrated Circuits and Systems Design*, 2002, pp. 95–100.

[22] J. M. Kontoleon and J. Andrianakis, “Reliability analysis of simplex and duplex memory systems with SEC and soft-error scrubbing recovery,” *International Journal of Quality & Reliability Management*, vol. 20, no. 5, pp. 620–636, 2003.

[23] A. M. Saleh, J. J. Serrano, and J. H. Patel, “Reliability of scrubbing recovery-techniques for memory systems,” *IEEE Trans. Reliability*, vol. 39, no. 1, pp. 114–122, Apr. 1990.

[24] R. M. Goodman and M. Sayano, “The reliability of semiconductor RAM memories with on-chip error-correction coding,” *IEEE Trans. Information Theory*, vol. 37, no. 3, pp. 884–896, May 1991, part 2.

[25] M. Blaum, R. Goodman, and R. McEliece, “The reliability of single-error protected computer memories,” *IEEE Trans. Computers*, vol. 37, no. 1, pp. 114–119, Jan. 1988.

[26] G. C. Yang, “Reliability of semiconductor RAMs with soft-error scrubbing techniques,” *IEE Proceedings Computers and Digital Techniques*, vol. 142, no. 5, pp. 337–344, Sept. 1995.

[27] P. Reviriego, J. A. Maestro, and C. Cervantes, “Reliability analysis of memories suffering multiple bit upsets,” *IEEE Trans. Device and Materials Reliability*, vol. 7, no. 4, pp. 592–601, Dec. 2007.

[28] F. Faure, “Injection de Fautes Simulant les Effets de Basculement de bits Induits par Radiation,” Ph.D. Thesis, Institut National Polytechnique de Grenoble, , Nov. 2005.

[29] W. Feller, *An Introduction to Probability Theory and Its Applications*. : John Wiley & Sons, 2001, ISBN: 0471257095..

Juan Antonio Maestro holds a M.Sc. degree in Physics (1994), and a Ph.D. degree in Computer Science (1999) from Universidad Complutense de Madrid. He has served both as a Lecturer, and Researcher at several universities, as Universidad Complutense de Madrid, UNED (Open University), Saint Louis University, and Universidad Antonio de Nebrija, where he currently manages the Computer Architecture and Technology Group. His current activities are oriented to the Space field, with several projects on reliability and radiation protection, as well as collaborations with the European Space Agency. He is the author of numerous technical publications, both in journals and international conferences. In addition, he has worked for several multinational companies, managing IT projects as a PMP, and organizing support departments. His areas of interest include High Level Synthesis and co-Synthesis, Signal Processing and Real-Time systems, Fault-tolerance, and Reliability. He is a member of the IEEE.

Pedro Reviriego received the M.Sc., and Ph.D. degrees (Honors) from Technical University of Madrid in 1994, and 1997, both in Telecommunications Engineering. From 1997 to 2000, he was an R&D Engineer at Teldat working on router implementation. In 2000, he joined Massana to work on the development of 1000BaseT transceivers. During 2003, he was a Visiting Professor at University Carlos III. From 2004 to 2007, he was a Distinguished Member of Technical Staff with LSI Corporation working on the development of Ethernet transceivers. He is currently with Universidad Antonio de Nebrija. His research interests are fault tolerant systems, performance evaluation of communication networks, and the design of physical layer communication devices. He has authored numerous papers in international conferences and journals. He has also participated in the IEEE 802.3 standardization for 10GBaseT. He is a member of the IEEE.