Error Detection Control System based on CheckSum using Orthogonal Systems of SAC functions

Dr. APOSTOLOS LEROS
Associate Professor
Department of Automation
Technological Education Institutes of Halkis
34400 Psahna, Halkis, Evia, Greece

Abstract: - In this paper a new approach for increasing the efficiency of checksum error detection is proposed. In order to decrease the probability of undetected errors, it is suggested to use Avalanche transformation of codes before the checksum calculation. Analytical estimations of undetected error probabilities are presented. Comparative analysis indicated high effectiveness of the suggested approach.

Key-Words: - Error Detection, checksum, SAC functions, Error Control System

1 Introduction

Checksum is one of the most widely utilized in practice means of detecting errors that appear when blocks of codes are being transferred in the channel or stored in memory [5]. In particular, this method of control is provided by the standard protocols which regulate the data transfer between the computer and modem, or between remote modems [7]. The error detection during the storage of data in the sectors of flopy disks is also achieved with the use of checksum. Nowdays the dynamic increase of information transfer relative to speed results in decreasing the reability of data transfer. So, the important problem of checksum utilization, increases the error detection reliability**.**

This requires the design of specialised means to increase the effectiveness of traditional methods for error detection.

2 Analysis of checksum error detection reliability

As previously mentioned, checksum is a way of error detections in data blocks. Suppose, that a data block contains k codes, D_1, D_2,…,D_k, each having n bits length. In comparison with other methods of error detection, such as cyclic redundancy checking (CRC), the important merit of checksum is simplicity and high speed.

The main disadvantage of checksum is the impossibility of detecting errors which appear in the same position in an pair of codes, ($\{D_i , D_j\}$, $i \neq j$, i=1,..,k, j=1,..,k), in the block. Traditionally checksum S of the data block is formed in such a way that each bit of checksum is XORed among the bits in all codes of the block.

$$S = D_1 \oplus D_2 \oplus ... \oplus D_k \quad (1)$$

The analysis of the practical use of the checksum [3,5] error detection method shows that the quantity of k codes in the controlled block substantially exceeds the n bits of length for each code, i.e., in practice n<<k.

Thus, from the practical point of view, it can be assumed that in one code, does not appear more than one error and multi errors appear in different codes of block. For the error appearance, we suppose the binomial error model [5], and letting p_b, be the probability that one erroneous bit has appeared in a transmitted data block, then the probability p_c of a single error during trasmittion of an n-bits code, is determined by the Bernoulli's formula: $p_c = \binom{n}{1} \cdot p_b \cdot (1-p_b)^{n-1} = n \cdot p_b \cdot (1-p_b)^{n-1}$.

The probability p_r of r errors occurring in the controlled block of data is determined by the expression:

$$p_r = \binom{k}{r} \cdot p_c^r \cdot (1-p_c)^{k-r} = \\ = \binom{k}{r} \cdot n^r \cdot p_b^r \cdot (1-n \cdot p_b \cdot (1-p_b)^{n-1})^{k-r} \quad (2)$$

In practice, the quantity of errors appearing in the trasmitted block does not exceed 3-4 erroneous bits. Therefore, it is justified to analyze the probabilities of 2 or 4 errors (even number of errors will be detected always) [3] beeing detected during transfer or storage of the data block.

Suppose that the number of the errors occurred during block transmission is two. Let's assume that the first of them occurred with the transfer of code D_1, and the second occurred with the transfer of the code D2. Also let's assume that with the appearance of the error, code D_1 is transformed in a new code H_1 and code D_2 respectively in code H_2. Moreover this

pair of the new codes differs only in one bit. With the usual formation of checksum, the errors will not be detected, if they occur in one and the same bit: i.e., with the fulfillment of conditions $D_1 \oplus D_2 = H_1 \oplus H_2$ or $D_1 \oplus H_1 = D_2 \oplus H_2$. Assume that the first error occurred in the j^{th} bit, $j \in \{1,\ldots,n\}$, then the probability, $P_{2c,}$ of the error not appearing in the same position bit of the second code is equal to:

$$P_{2c} = \frac{1}{n} \quad (3)$$

The probability P_{4c} of not detecting the same position bit in fourfold codes using the traditional checksum can be determined as follows.

Suppose that in a fourfold code taking them in two pairs, the same position bit error for each pair will not be detected, then let's examinate all possible variants for fourfold code error bit position in which such errors will not be deteced by the usuall chechsum method. The first error can be located in the jth bit position of one code of block. The second error can be located in the i^{th} bit position in any of the remaining codes, where $i \neq j$. The third error belonging in another code of the same data block must coincide with the position of the first or the second error. Therefore there are 2 possible bit-positions for the third error. Then it is clear that there is only one possible bit-position for the fourth error. Since there are n^2 possible versions of positions for the third and the fourth errors, only two of the fourfold errors will not be detected using the traditional checksum method. There is one case of bit-position for the four errors which will not be detected. In this case all four errors occur in the same bit position of the differenet codes of block. The probability of this case, three out of four errors in four codes to be coincident is $1/n^3$ because every possible bit position of the first error corresponds only to one position of all the other three errors taken multiplicatively from n possible cases for each code (n^3 possible cases). Consequently, the P_{4c} probability that the fourfold errors will not be detected using the traditional checksum is defined as:

$$P_{4c} = \frac{2}{n^2} + \frac{1}{n^3} \quad (4)$$

Similar methods for increasing the reliability of checksum block control have been proposed [5,6,7]. However, all these methods demand the utilization of additional check bits. Realization of these methods require complex computation and channel resources for the transfer of the additional check bits. Consequently, the main checksum advances of simplisity and high speed are lost.

More effective is the approach, based on the optimization of coding the checksum according to real probabilistic characteristics of the appearance of erros [3]. This approach does not require additional control check bits and its utilization has no influence on the rate of data tranfer.

3 Utilization of Avalanche transformation for increasing checksum data transmission control reliability

For practical optimization checksum coding we proposed to calculate modified checksum using Boolean Avalanche transformation of transmitted codes.

Such Boolean tranformations are orthogonal systems of n Boolean functions with n variables that satisfy Strict Avalanche Criterion (SAC).

A Boolean function $f(x_1,\ldots,x_n)$ defined on a set Z with binary elements consisting of all possible 2^n n-tuples of n variables, satisfies the Strict Avalanche Criterion (SAC), if there is a 50% probability that the complement of a single incoming n-tuple data bit affects the output of the Boolean function by 50%, as follows:

$$\forall j \in \{1,\ldots,n\}:$$
$$\sum_{x_1,\ldots,x_n \in Z} (f(x_1,\ldots,x_j,\ldots,x_n) \oplus f(x_1,\ldots,\overline{x_j},\ldots,x_n)) = 2^{n-1} \quad (5)$$

Usually these type of functions are being used in cryptographical algorithms and their design methods have been developed in [1,2,11].

If one of the n inputs of avalanche tranformation is changed then half of its outputs will be changed. This means, that there is an "avalanche amplifier" which by changing one incoming of the n-tuple data bit tranforms half of the outputs. Because every function of this system satisfies the Avalanche Criterion, these tranformations are called "avalanche".

We propose to use the Avalanche trasformation as "error amplifier" which prevents dual errors "masking" interaction in checksum. The avalanche amplifier in case a single bit error appears on the checksum will have the effect of decreasing the probability of an error to appear in the same bit position (masking the repetitive error on the same bit).

Let's denote with F(D) the result of avalanche tranformations of n-bits on a code D consisting of n-bits. For the checksum, S^m, on the avalanche tranformation of the code D we propose to calculate the XORed of the same bits avalanche

tranformations $F(D_1),F(D_2),\ldots,F(D_k)$ from all codes $D_1,D_2,\ldots,D_k$ in block:

$$S^m = F(D_1) \oplus F(D_2) \oplus \ldots \oplus F(D_k) \quad (6)$$

The proposed control scheme of utilizing the Avalanche tranformation in the checksum is shown in Fig.1

With the appearance of a single error during the transmition of the code D_q, $q \in \{1,\ldots,k\}$, this code D_q is transformed into code Hq, which differs from the initial code Dq only in one bit. Thus, defining with Ψ_q the code which affects the checksum and consists of "1" on the error bit and all other bits of "0", we can calculate the following:

$$H_q = D_q \oplus \Psi_q, \Psi_q = \{\psi_q^1, \psi_q^2, \ldots, \psi_q^n\},$$
$$\psi_q^j \in \{0,1\}, \forall j = 1,\ldots,n : \sum_{j=1}^{n} \psi_q^j = 1 \quad (7)$$

In view of the properties of avalanche tranformations, the code $F(H_q)$ of the error code H_q will differ from the avalanche tranformeted code $F(D_q)$ of the initial code Dq in half of each bits, i.e.,:

$$F(H_q) = F(D_q) \oplus \Delta_q, \Delta_q = \{\delta_q^1, \delta_q^2, \ldots, \delta_q^n\},$$
$$\delta_q^j \in \{0,1\}, \forall j = 1,\ldots,n : \sum_{j=1}^{n} \delta_q^j = \frac{n}{2} \quad (8)$$

Thus, if a single error appears then n/2 bits of the modified checksum will change. If a second error appears then another n/2 bits will change of the modified checksum bits. It is clear, that the probability of the masking interaction of n/2 erroneous bit pairs is less than the probability of the masking interaction of a single bit pair.

4 Analysis of effectiveness of using avalanche transformations for increasing checksum error detection reliability

For the evaluation of the effectiveness of this approach it is necessary to define and compare the probabilities of multi error detection of traditional and proposed schemes of checksum calculation.

Suppose that the number of the errors which occurred during the transmission of the block is two. Let's assume that the first of them did occur with the transfer of code D_1, and the second with the transfer of a different from the first code, D_2. Further let's assume that with the appearance of the error, code D_1 is transformed in H_1, and code D_2 in H_2. Let's define the fault probability, P_{2f}, of the fact that a dual error will not be detected with the proposed control sheme of the checksum. Then the emergent pair of errors will not be detected, if the following condition is satisfied $F(D_1) \oplus F(D_2) = F(H_1) \oplus F(H_2)$. This condition will be true only in case, when $F(D_1) \oplus F(H_1) = F(D_2) \oplus F(H_2)$. The last condition is true if after the errors have occurred during the transformation of D_1 and D_2 the positions of all n/2 bits changed. That is, $F(D_1)$ are equal to the positions of the n/2 changed bits in $F(D_2)$. The probability P_{2f} that the randomly selected n/2 positions in the n - bit code will coincide with the fixed n/2 positions is determined as follows:

$$P_{2f} = \frac{1}{\binom{n}{n/2}} = \frac{((n/2)!)^2}{n!} = \prod_{j=0}^{n/2-1} \frac{j+1}{(n-j)} \quad (9)$$

Thus, the probability of detecting dual errors during block tranformation using the proposed scheme of the proposed checksum control shceme, increases by t_2 times in comparion to the ordinary checksum scheme. The numerical value of the t_2 increase is determined by the formula:

$$t_2 = \prod_{j=1}^{n/2-1} \frac{n-j}{j+1} \quad (10)$$

For example, with n=8, the probability that the dual errors will not be detected is decreased by 8.7 times in comparion to the traditional checksum.

The values of the probabilities that the dual errors will not be detected using ordinary and modified checksum for 8 and 16 – bits length codes are show in the Table I.

Table I

n	P_{2c}	P_{2f}	$t_2 = P_{2c} / P_{2f}$
8	0.125	0.014286	8.75
16	0.0625	0.0000777	804.37

The results of Table I, coincide with the experimental results and demonstrate the high level of effectiveness of using the "single error amplifier". This case is often occurs in practice.

The effectiveness of the proposed approach is proportional to the codes length.

In a similar way it can be shown that the probability P_{4f} that the fourfold error will not be detected using the proposed scheme of the checksum can be defined by the following equation.

$$P_{4f} = \frac{\sum_{t=0}^{n/2} \left(\binom{n/2}{t}^2 \cdot \binom{n-2t}{n/2-t}\right) + 1}{\binom{n}{n/2}^3} \quad (11)$$

The values of the probabilities that the fourfold error will not be detected using ordinary and modified

checksum for 8 and 16 – bits length codes are show in the Table II.

Table II

n	P_{4c}	P_{4f}	$t_4 = P_{4c} / P_{4f}$
8	0.04296	0.001866	23
16	0.01123	$1.0122 \cdot 10^{-6}$	11094

Similarly the results of Table II, coincide with the experimental results and demonstrate the high level of effectiveness using a "single error amplifier". This fourfold case seldom occurs in practice.

From the results obtained from Table II it is obvious that for the fourfold error case the effectiveness of the proposed approach is proportional to the codes length. Thus, the theoretical and experimental investigations have proved that the efficiency and effectiveness of the proposed approach is high because it increases the checksum error detection reliability.

The proposed approach does not demand the use of additional check bits. The effect of an increase in the detection reliability of the checksum error is achieved due to the optimization of checksum coding. The utilization of the Avalanche tranformation makes it possible to increase the Hamming distance between the correct checksum and the checksum code which was destroyed by single error. This allows the significant decrease in the probability of multi error masking interaction which is not detected by the ordinary checksum method.

5 Hardware implementation

For the practical implementation of the proposed approach which increases the checksum error detection reliability it is necessary to synthesize a system of orthogonal Boolean functions which correspond to SAC. This problem may be solved using the methods proposed in [1, 2, 8, 9,11].

Another important implementation problem is the rapid calculation of the Boolean functions system.

This problem may be solved by using software and hardware. The most effective hardware way is FPGA implementation of the Avalanche transformation. Single-chip FPGA implementation allows parallel calculation of the Boolean system which consists of the Avalanche transformation [8, 10].

Analysis of the Avalanche transformations implemented in Altera FPGA devices has showed that the calculation rate is about a few tens nsec. Thus, the hardware FPGA implementation of the proposed approach for increasing the checksum error detection reliability offers the main advantage of high-speed error detection included in the proposed checksum control method.

6 Conclusion

The proposed approach for increasing the checksum error detection reliability is based on the utilization of the Boolean Avalanche transformation which plays the role of a "single error amplifier". This also allows a significant decrease in the probability of multi error masking interaction which is not detected by the ordinary checksum control method.

The theoretical estimations of the probabilities of non detected errors for different multiplicity have also been obtained.

Comparison analysis has demonstrated high level effectiveness of the proposed approach.

Probabilities that multi error will not be detected are significantly decreased in comparison to the ordinary checksum control.

In contrast to known approaches for increasing checksum error detection reliability the proposed one does not require additional check bit and its implementation does not influence the data transmission rate. The proposed approach is based on the optimization of checksum coding according to real probabilities characteristics of erros appearance. The developed aproach can be used to increase the reliability of error detection in channel data transmission of computer networks.

References:

[1] Bardis N.G**,** Mitrouli M., Markovskyy A.P, Some properties of Boolean functions and design of Cryptographically strong balanced Boolean functions, *RECENT ADVANCES IN STATISTICAL DESIGNS AND RELATED COMBINATORICS* University of Athens, July 7-9, 2003, Athens, Greece

[2] Bardis N.G, Bardis E.G., Markovskaja N.A., Polymenopoulos A., Design and Implementation of Boolean Balanced Functions Satisfying Strict Avalanche Criterion (SAC), *WSES Press – Problem in Applied Mathematics and Computational Intelligence*, ISBN: 960-8052-30-0, 2001, pp. 12-16.

[3] Di C., Proietti D., Richardson T., Telafar E., Urbanske R. Finite length analysis of low-density parity check ensembles for the binary erasure channel.//*IEEE Trans. Information Theory.-2002*,- Vol.48, No.6, - P.1570-1579.

[4] Fu F.W., Klove T., Wei V.K. On the Undetected Error Probability for Binary Codes.// *IEEE*

Transaction on Information Theory.-2003,- Vol. 49, No. 2,- P.382-391.

[5] Klove T., Korzhik V. Error Detecting Codes: General Theory and Their Application in Feedback Communication Systems. *Norwell, MA: Kluwer*, 1995. – 433 p.

[6] Konstantidinis S., Perron S., On a Simple Method for Detecting Synchronization Errors in Code Message.// *IEEE Trans.of Information Theory.- 2003*,- Vol.49, No. 5,- P.1355-1368.

[7] Saxena N.R., McCluskey E.J. Extended precision checksums. // *Proc.17-th Intern. Symp. Fault-Tolerant Comput.* : FCTS-17,- Pittsburgrh(USA).-1987.-P.142-147.

[8] Bardis N., Markovskyy A., El Xami Igiad, Rentko S., Method and Structure for Forming and Calculating Systems of Orthogonal Balance Boolean Functions, *Bulletin of National Technical University of Ukraine,* ISSN 0201-744X,ISSN 0135-1729,UDK 681.322, No.32, 1999, pp.90-98.

[9] Bardis N., Bardis E., Markovskyy A., Spyropoulos A., Design of Boolean Function from a Great Number of Variables Satisfying Strict Avalanche Criterion, *RESENT ADVANCES IN SIGNAL PROCESSING AND COMMUNICATIONS- IMACS/IEEE*, ISBN: 960-8052-03-3, pp. 107-112, 1999.

[10] N.G.Bardis, A. Polymenopoulos, E. G. Bardis, A. P. Markovskyy, N. E. Mastorakis, Hash Algorithms: a design for parallel calculations, *International Journal Computer Science*, 2003, Issue 1, Volume 1, ISSN: 1109-2777, 2002, pp. 48 - 56.

[11] M.Mitrouli, A.P.Markovskyy, Method for design of balanced Boolean functions satisfying strict avalanche criterion (SAC), *WSES Press, Recent Advances in Communications and Computer Science*, ISBN: 960-8052-86-6, 2003, pp.148-154.

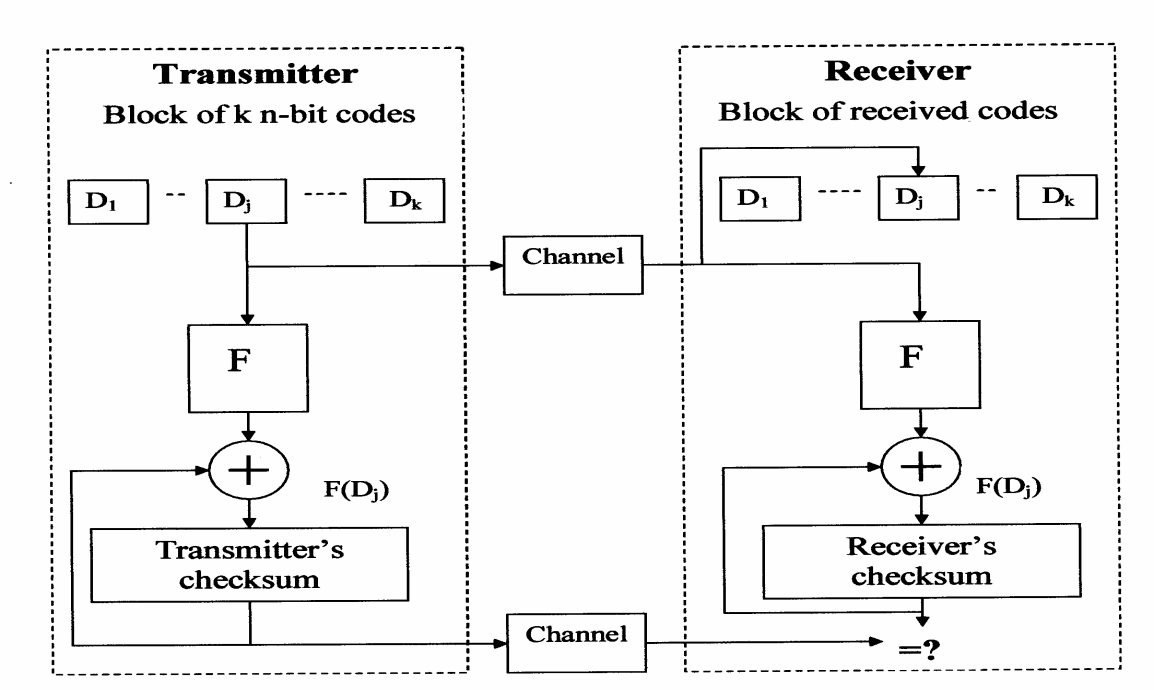

Fig. 1.