PERFORMANCE INCREASE OF ERROR CONTROL OPERATION ON DATA TRANSMISSION

Nikolaos G. Bardis, *Member, IEEE* , Athanasios Drigas and Oleksandr P. Markovskyi

***Abstract*— In this paper a new approach is proposed to increase the performance of the operation of error control on data transmission. Specifically, a hardware structure for parallel Cyclic Redundancy Check (CRC) calculation is developed to speed up the error control operation of data transmission. Based on a study of the properties of both CRC and Check Sum (CS) a new error detecting scheme is developed which combines CRC and CS. Also it is shown that the proposed error detecting scheme ensures high reliability and performance of the error control operation on data transmission in comparison to CRC alone.**

***Index Terms*— error detection, error control system, CRC, CS.**

I. INTRODUCTION

THE development of technologies for the transmission of information in the computer networks and in the telecommunication systems is inseparably connected with the problem of integrity and of ensuring high reliability during the process of error detection.

The dynamic increase in the speed of information transmission in the buses of the computer systems and the channels of computer networks brings about stringent requirements for the performance of the hardware implementation of the error control algorithm: it must ensure the realization of the operations which are connected with the rate of error detection for data transmission.

Thus, under the current and future technological conditions to achieve high effectiveness of the error control operation on data transmission it is necessary to intermix two fundamental components which are reliability and performance of error detection.

The increase of the requirements for the performance of the hardware implementation of the error control algorithm has the consequence of not increasing the effectiveness (speed and reliability) of the CRC which is the most common in practice sequential method for error detection [1], [8], [11].

One solution to the above problem of effective error control during data transmission is to develop a new approach and proper hardware implementation of error control, which would ensure reliability level of error detection close to CRC and also high performance of the error control operation based on hardware parallel processing [11].

Thus, the above problem of realizing the error control operation in data-transmission systems is practically very important and requires attention for the current and future technological developments in computer networks, in telecommunications systems, and in military applications. Especially, for the military we can mention the application in the weapon control systems where strong electromagnetic interferences may become sources of errors in one case during data transmission among the components of the system due to their nature of operation, and in an other case during channel transmission between the computer control weapon system components especially in ships and aircraft.

Manuscript received September 4, 2009. Performance Increase of Error Control Operation on Data Transmission Paper. N. G. Bardis, A. Drigas and O. P. Markovskyi

N. G. Bardis is adj. professor at Department of Mathematics and Engineering Science of the University of Military Education, Hellenic Army Academy and researcher at Net Media Lab of Institute of Informatics & Telecommunications, N.C.S.R. Demokritos, (e-mail: bardis@ieee.org).

A. Drigas is a Senior Researcher at N.C.S.R. Demokritos. He is the Coordinator of Telecoms and founder of Net Media Lab since 1996. (e-mail: dr@itt.demokritos.gr, http://imm.demokritos.gr).

O. P. Markovskyi is with the Department of Computer Engineering, National Technical University of Ukraine, (Polytechnic Institute of Kiev) (e-mail: markovskyy@mail.ru).

II. ANALYSIS OF THE CURRENT SITUATION FOR ERROR CONTROL IN THE HIGH-SPEED CHANNELS FOR DATA TRANSMISSION

In the current buses (USB, PCI) of computer systems the information is transmitted by blocks, which can be divided into symbols. Let the transmitted data block B consist of m-bits, i.e., $B=\{b_1,b_2,...,b_m\}$, $b_l \in \{0,1\}$, $l=1,...,m$, which is divided into

978-1-4244-6273-5/09/$26.00 ©2009 IEEE

$t = m/n$ symbols, each consisting of n bits. If we denote the j[th] symbol in the form, $X_j = \{x_1, x_2, ..., x_n\} = \{b_{(j-1)\cdot n+1}, b_{(j-1)\cdot n+2}, ..., b_{j\cdot n}\}$, $j = 1,2,...,t$, $\forall i \in \{1,2,...,n\} : x_i \in \{0,1\}$, then the block B can be presented in the form of $t = m/n$ symbols, i.e., $B = \{X_1, X_2, ..., X_t\}$, $X_j \in \{0,1,..., 2^{n-1}\}$.

To ensure the reliability of the data transmission two base technologies are used: One is the Forward Error Correction (FEC) with the application of correcting codes and the other is the Automatic Repeat Request (ARQ) which is implemented by repeating the transfer automatically in case of error detection [4], [9], [10]. The use of the FEC technology transmits the control code after each symbol is transmitted, while the ARQ technology transmits the control code only after the entire block is transmitted.

The main advantage of the ARQ technology is that the error detection is performed faster and it is simpler in comparison to the error correction which is performed with the FEC technology. With the use of the ARQ technology the size of the transmitted control information in the transmission line is substantially less.

In the wired connections for data transmission the rate of the appeared errors is relatively small and therefore for error-free transmission predominantly the ARQ technology is preferred [4], [9], [10].

Such lines of data transmission correspond to the theoretical model of binary symmetrical channel. According to this model the probability of error appearance with different multiplicity obeys the law of binomial distribution. In practice, for the transmission of the block, this means that the absence of errors is most probable and the probability of errors decreases with an increase in their multiplicity [5].

The most reliable method of block error control for data transmission is the CRC method. This method reveals all errors of odd multiplicity and all dual errors during the transmission of the data block [1, 7]. Apart from this, the CRC method guarantees the detection of errors which are localized within the block fragment with length less than the degree of the CRC polynomial. The probability of the errors of even multiplicity larger than two which are not revealed by the CRC method is 2^{-k}, where k is the degree of the CRC polynomial detection method.

The dynamic increase of speed for data transmission worsens the problem of implementing the CRC method, since this method principally works sequentially on the bits of the transmitted data block. The time t_{CRC} needed to accomplish the CRC control code for the transmission of a data block with m bits is $m \cdot (t_{xor} + t_{sh})$, where t_{xor} is the XOR summation time and t_{sh} is the shift time of the code. For the data transmission channel with spectrum modulation in which n bits are modulated by one channel signal the problem of CRC rate calculations is a very demanding one. For such channels the following condition must hold:

$$t_{xor} + t_{sh} < \frac{\tau}{n} \quad (1)$$

where τ *is the* rate of the signal transmission channel.

The increase of speed for error control using the CRC method can be achieved by working with several bits simultaneously within the same block [2], [3]. However, the complexity of the matrix transformation, which realizes this approach, grows exponentially with the increase in the number of bits used simultaneously.

Seldom, for the detection of errors during data transmission, the Check Sum (CS) method is used since it possesses smaller reliability in comparison to the CRC method. With the CS method it is possible to reveal all errors of odd multiplicity, but dual errors are not revealed with probability $P = 1/n$ [6].

The probability of not detecting the errors with even and larger than two multiplicity decreases with an increase in the number of the multiplicity. For the case of d multiple errors the probability of not detecting these errors is proportional to $1/\left(n^{d/2}\right)$. The relatively low reliability of error detection for the CS method of small but even multiplicity is caused by the ineffectiveness of the coding of the single errors [1], [7], [8], [12].

Known modifications for the CS method of error detection which use special coding of CS components to increase the reliability in case of small multiplicity are presented in [1], [3], [7], [8], [11], [12]. However, regarding the criterion of reliability these CS modifications are inferior to the CRC method.

The use of the CS method practically does not set limitations on the speed of the calculation of the control code since the structure of its operations allows their parallel hardware implementation. This provides the potential for error control in the transmission rate if $t_{xor} \le \tau$.

As a result the CRC and CS methods presented above each implemented alone do not ensure the high reliability of error detection and performance of the control operations which are required for the current and future rate of the data transmission.

The purpose of this paper is to develop a new method for error control, which satisfies both the high reliability of error detection and the fast performance of the error control operations based on parallel hardware implementation of the execution of control

functions during the rate of the data transmission. This new method consists of three approaches. The first one presented in section 3 is the parallel CRC calculation using precomputation tables. The second approach presented in section 4 is the combined use of both CRC and CS codes in which the error control is applied to sub blocks of the transmitted data block using the CRC method in every sub block and in parallel and the final control code is formed using the CS method. The third approach presented in section 5 is the combined use of both CRC and CS codes in which the error control is applied to symbols of the transmitted data block using a modification of the CS method for every symbol and the final control code is formed using the CRC method in parallel at the CS results of the symbols.

The following table summarizes all the basic nomenclature (indices) used through out the paper for the transmitted data block B:

m – length (bits) of the transmitted data block B;
n – length (bits) of every symbol;
t – number of the symbols,
d – multiplicity of errors,
h – number of sub blocks,
q – length (bits) of each sub block,
k – degree of the CRC polynomial.

III. PARALLEL CRC CALCULATION USING PRECOMPUTATION TABLES

The parallel processing which is connected with the calculation of the CRC control code increases exponentially the architecture complexity depending on the number of simultaneously workable single bits of information in each data block B [5], [11].

The large growth of technology of constant memories gives the possibility of real help for the rapid calculation of the CRC control code based on the use of precomputed tables. It is well known that for every binary *n* bit length code $C=\{c_1,c_2,...,c_n\}$ corresponds a polynomial $C(X)=c_1\cdot x^n+c_2\cdot x^{n-1}+...+c_{n-1}\cdot x+1$ over Galoise fields $GF(2^n)$. The operation plus "+" corresponds to logical operation XOR. Using the CRC method the polynomial $K(X)$ that corresponds to the control code of the transmitted data block *B* is calculated as the remainder of the division of the polynomial $B(X)=b_1\cdot x^m+b_2\cdot x^{m-1}+...+b_m$ to the selected base CRC polynomial *G(X), i.e.,* $K(X)=B(X)\bmod G(X)$. For example, the base polynomial $G(X)$ of the standard CRC-16 is $G(X)=x^{16}+x^{15}+x^2+1$. According to the property of the division of polynomials on Galoise fields, $K(X)$ can be presented in the form of XOR of all the remainders of the divisions of the nonzero components of $B(X)$ to the base polynomial $G(X)$ as follows:

$$K(X)=b_1\cdot x^m \bmod G(X)\oplus \\ \oplus\, b_2\cdot x^{m-1}\bmod G(X)\oplus ...\oplus b_m \bmod G(X) \quad (2)$$

The above expression (2) can be used as a theoretical base for designing an architecture for parallel calculation of the CRC control code.

The control code *K* of the data block *B* is decomposed into *h* parts, each of size *q*-bits, which are the sub-block $S_1,S_2,\ldots,S_h$, where q=m/h. Every sub-block S_i includes bits of the data block B at intervals every h bits: $S_i=\{b_i,b_{i+h},\ldots,b_{m-h+i}\}$, $i\in\{1,\ldots,h\}$. Correspondently, the S_i bits appear with the rate τ/h, which is by *h* times less than the channel rate τ.

For example in case h=4 the architecture of calculating the CRC control code with parallel calculation of 4 predetermined control codes is presented in Figure 1. The proposed architecture contains a counter of the number *h* of the sub-blocks called CNSB, an h -bits length Fragment Register (FR), *h* sub-blocks of table memory *T1,..., Th*, *h* logical AND gates each containing *k* elements, an XOR block element with h+1 inputs, and finally one Control Code Register (CCR) which forms the control code.

The cumulative size of the table memory as presented in the proposed architecture is of 2^m cells, each of size k – bits.

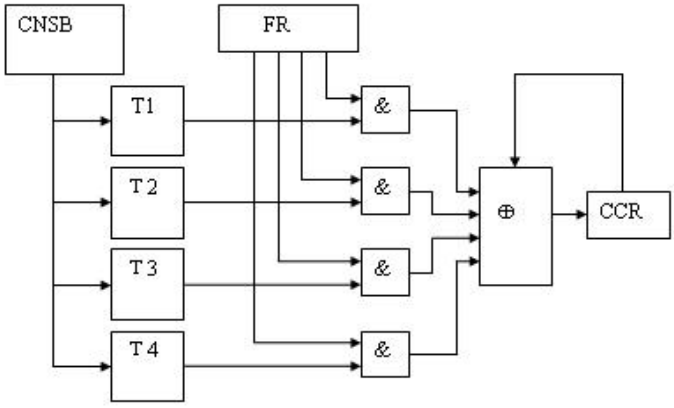

Fig. 1: Architecture of parallel calculation of the CRC control code

The total memory capacity of tables in proposed architecture is 2m k-bits word.

IV. COMBINED USE OF CRC AND CS CODES USING SUB BLOCKS OF THE TRASMITTED DATA BLOCK

The objective of this research is to combine the use of CRC and CS control codes in order to achieve high performance with reliability for error detection and control calculation during data transmission.

In this approach the control code for the data block B is decomposed into h parts, each of size q- bits, which are the sub-block $S_1,S_2,\ldots,S_h$, where q=m/h and thus $S_i = \{b_i, b_{i+h}, ..., b_{m-h+i}\}$, $i \in \{1,...,h\}$. Then the bits for each sub-block S_i are transmitted in the communication line with rate τ/h, which as previously mentioned is h times less than the rate τ of the transmitted data block B.

It is noted that the indices h and m for the number of sub-blocks and bits of the data block *B*, respectively, satisfy the relation h<m. With k we denote the degree of the selected base CRC polynomial G.

The structure of the data block B with m=256 bits and its decomposition into h=4 sub-blocks each with q=64 bits is shown on Figure 2.

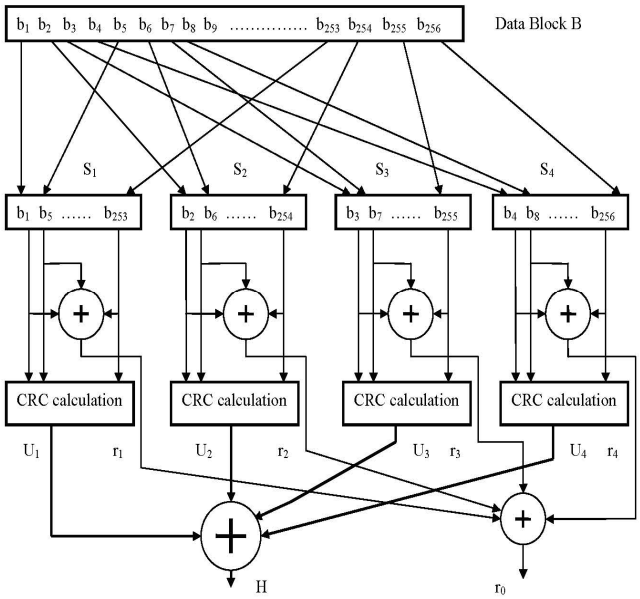

Fig. 2: Architecture of decomposing data block B into sub-blocks

As presented in figure 2 above, for each sub-block $S_1,S_2,\ldots,S_h$ of the transmitted data block B we calculate the particular CRC codes $U_1,U_2,\ldots,U_h$ which are the *k*-bits remainders of the division between each sub-block polynomial $S_1(X),S_2(X),\ldots,S_t(X)$ by the base CRC polynomial $G(X)$.

We also calculate the bit parity: $r_j=b_i \oplus b_{i+h} \oplus \ldots \oplus b_{m-h+i}$ for every transmitted sub-block S_i. The particular control code Q_i for each i^{th} sub-block S_i is formed as the concatenation of its CRC-code U_i with its parity bit r_i, i.e., $Q_i=U_i||r_i$, giving its size of k+1 bits. The other part of the control code D_B of the data block *B* is formed as the concatenation of the k - bits of the code *H* and the bit parity r_0: $D_B = H||r_0$. The code *H* is formed as the XOR of all the $U_1,U_2,\ldots,U_h$ and the parity r_0 is formed as the XOR of all the $r_1,r_2,\ldots,r_h$ of the sub-blocks.

$$\begin{aligned} H &= U_1 \oplus U_2 \oplus ... \oplus U_h, \\ r_0 &= r_1 \oplus r_2 \oplus ... \oplus r_h \end{aligned} \tag{3}$$

Since the $Q_1,Q_2,\ldots,Q_h$ codes are independent of each other these can be hardware implemented in parallel. The number h of the sub-blocks is defined differently every time depending on the rate of the data transmission:

$$t_{xor} + t_{sh} < \frac{\tau}{h} \tag{4}$$

The time t_D to form the (k+1) - bits of the control code D_B of the data block B from (3) is form by the following expression:

$$t_D = t_{xor} \cdot \log_2 h \tag{5}$$

The control code D_B of the block B in the receiver end is calculated during the transmission of both the data block B and the control code D_B sent by the transmitter.

The degree of reliability of the above combined CRC and CS approach can be determined as follows. Since in the combined control code D_B of the data block *B* exists the parity bit $r_0 = r_1 \oplus r_2 \oplus ... \oplus r_h = b_1 \oplus b_2 \oplus ... \oplus b_m$, then the error in odd multiplicity is always detected. The case of double error appearing in two bits in sub-block S_i, according to the CRC properties [1] gives one change in the corresponding remainder U_i. Consequently, the control code D_B change, which means that such type of errors are always detected.

In the sequel we will show that all the rest errors during the transmission of the data block *B* are not detected with probability $2^{-(k+1)}$. At the appearance of an error during the transmission of some sub-block of data S_i, this sub-block changes in the form of $S_i \oplus E_i$, where E_i is the error vector of size *q* - bits with the particular bit in error set to 1 and the rest bits not in error set to 0.

In case the i^{th} sub-block S_i is transmitted without error then E_i=0. According to the properties of the remainder of the division by prime polynomial [5], for each sub-block S_j the following holds:

$$(S_j \oplus E_i) \bmod G = S_j \bmod G \oplus E_j \bmod G$$

In this way any change ΔH of the *H* control code can be presented as follows:

$$\Delta H = E_1 \bmod G \oplus E_2 \bmod G \oplus ... \oplus E_h \bmod G \tag{6}$$

According to the properties of the remainder of the division by prime polynomials [5] the power *q* is by far bigger than 2^k, i.e., $2^k > q >> k$. The binary code ΔH, of the remainder of this

division has g different tupplets (e.g., strings of bits) each taking with equal probabilities values in the interval from 0 to 2^k-1.

The result of the XOR logical operation between ΔH with the k-bits parity code r_0 has values 0 or 1 with equal probabilities. The k-bits parity code r_0 changes at the appearance of an error with probability 0.5.

Thus at the appearance of an error in the transmission of data block B, its control code D_B changes by value $\Delta D_B = \Delta H \| \Delta r_0$ which has size (k+1) – bits, each bit having value 0 or 1 with equal probability. The probability for the control code D_B of the data block B not to change at the appearance of an error, i.e., ΔD_B=0, has the value of $2^{-(k+1)}$, which is what we were intended to prove.

In this way the probability of not detecting all the errors of even multiplicity except the double errors in some sub-block S_i is $2^{-(k+1)}$. This means that all the errors apart from the double errors are detected with the proposed architecture of error control approach with probability value larger than the one achieved by the CRC code alone.

To guarantee the detection of all the double errors, the proposed combined CRC and CS architectural error control approach for the transmission of data block B can be modified in the following way. The control code D_B of the data block B is the result of the AND logical operation of the H, r_0 and L codes, i.e., $D_B=H\|r_0\|L$, where the log_2h - bits size $L = r_1 \cdot 1 \oplus r_2 \cdot 2 \oplus r_3 \cdot 3 \oplus ... \oplus r_h \cdot h$ and the numbers 1,2,..,h are the binary representation of the number for each even sub-block.

At the appearance of double errors during the transmission of data block B created by one bit of different sub-blocks S_l and S_t, where l and t$\in$ {1,…,h}, l$\neq$t, change their r_l and r_t parity bits, respectively. The ΔL change of the code L is the XOR result of all the l and t binary numbers, i.e., $\Delta L=l \oplus t$. From the above it is obvious that the XOR result of all the changed ΔD_B codes of the data block B is not equal to 0. This means that the probability that the errors in codes of even multiplicity will always be detected. Since every bit of the sub-block particular control code is equal to one with probabilities 0.5, then it is obvious that every k+1+$\log_2 h$ bit control code of the data block B will also be equal to one with probabilities 0.5, too. This means that the probability of not detecting the error of even multiplicity larger than 2 is equal to $2^{-(k+1+\log_2 h)}$ [1].

In this way, the reliability of the proposed combined CRC and CS architectural error control approach for the transmission of the data block B is larger than the reliability of the CRC alone, as well as the performance of the error control are h times larger than the performance of the CRC alone.

V. COMBINED USE OF CRC AND CS CODES USING SYMBOLS OF THE TRANSMITTED DATA BLOCK

The second approach of implementing the combined CRC and CS for error control uses n-bits symbols $X=\{x_1,x_2,...,x_n\}$ of the data block B. These symbols are transformed by the linear Boolean functions $\lambda_i(.)$, $i=1,...,s$, giving the code Y=$\{\lambda_1(X),\lambda_2(X)\ldots,\lambda_s(X)\}$ of size $s=1+log_2n$ bits long. The control code of the transmitted data block B is calculated as the remainder of the divisions of the polynomial $V(X)$ by the selected CRC base polynomial $G(X)$, where the code V is formed as the concatenation of all t codes Y_t: $V=Y_1||Y_2||...||Y_t$.

All the linear transformations $\lambda_i(.)$, $i=1,...,s-1$ (except the last one $\lambda_s(.)$) are chosen so that every binary variable $x_1,x_2,...,x_n$ enters as a linear coefficient in at least one linear transformation $\lambda_1,\lambda_2,...,\lambda_{s-1}$ with the restriction that if one variable is used in one specific linear transformation then this variable is not used in any of the others. This restriction is satisfied by the system of log_2n linear transformations of Binary Coder with n inputs. The last linear transformation $\lambda_s(.)$ is formed as the XOR of all the $x_1,x_2,...,x_k$ variables , i.e., $\lambda_s=x_1 \oplus x_2 \oplus ... \oplus x_k$.

As an example for 8 variables x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 the linear functions λ_1, λ_2, λ_3 *and* λ_4 are formed as follows (the first 3 functions are the functions for the 8 input Binary Coder: $\log_2 8=3$ and the last one is the XOR of all the variables). First we form the 3-bits code xxx to indicate which particular x_i, i=1, 2, 3,…,8, variable will be used in each linear functions λ_i. Next we calculate the decimal values of all the possible combinations xx**1** of the 3-bits code (001=1, 011=3, 101=5, and 111=7) and increase them by 1 to point to the corresponding x_i to be used in linear functions λ_1 , i.e., $\lambda_1=x_2 \oplus x_4 \oplus x_6 \oplus x_8$. We calculate the decimal values of all the possible combinations x**1**x of the 3-bits code (010=2, 011=3, 110=6, and 111=7) and increase them by 1 to point to the corresponding x_i to be used in linear functions λ_2, i.e., $\lambda_2=x_3 \oplus x_4 \oplus x_7 \oplus x_8$. We calculate the decimal values of all the possible combinations **1**xx of the 3-bits code (100=4, 101=5, 110=6, and 111=7) and increase them by 1 to point to the corresponding x_i to be used in linear functions λ_2, i.e., $\lambda_3=x_5 \oplus x_6 \oplus x_7 \oplus x_8$.

Finally, we form $\lambda_4=x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8$ (the XOR of all the $x_1,x_2,...,x_8$ variables).

When one or more errors appear in an odd number of the X_t symbols of the data block B then the s^{th} most significant bit of its (parity) code $Y_t = \lambda_s(X_t)$ changes. When two bits in the same symbol X_t are in error then at least one of the linear transformations $\lambda_1,\lambda_2,...,\lambda_{s-1}$ changes.

In this way the proposed linear transformation provides always the change of code Y_t in odd or even number of symbols X_t. If the degree k of the selected CRC base polynomial is larger than n symbols, i.e., X_t: $k > log_2n$, then the change of code Y_t will always bring a change in the CRC control code.

This means that the proposed approach always allows the detection of errors of odd multiplicity and double errors which

appear in the transmission of the data block B. Also in the proposed approach the reliability in the line of the data transmission which corresponds to the theoretical model of binary symmetrical channel is near the reliability provided by the CRC.

Also the calculation of the error control code is executed faster compared to the CRC. The error control in the rate of data transmission should be executed in less complexity compared to (1):

$$2 \cdot t_{xor} + t_{sh} < \frac{\tau}{\log_2 n} \quad (7)$$

As an example the time to implement the error control in a 32-bits bus PCI (n=32), is decreased 5 times compared to the CRC.

VI. Conclusion

The above analyses have shown that the efficiency of error control during the data transmission with the use of CRC, which is the most practical way of error detection, does not allow in contemporary very high speed channels of electronic systems and communications for the error control to be implemented at the rate of data transmission.

To increase the efficiency of error control during the data transmission using the CRC code an architecture is proposed which can be hardware implemented in parallel based on precomputed memory tables.

The study has showed that the possibility of increasing the efficiency of error control in data transmission using the proposed combined CRC and CS architectural error control approach is feasible which also allows for the parallel calculation in hardware level. The two presented combined CRC and CS architectural error control approaches result in the detection of three different types of errors which ensure high performance due to parallel implementation with a degree of reliability very close to that of CRC alone.

However, expressions (4) and (7) validate the theoretical increase of speed for error control detection. To obtain a more reliable estimation of the effectiveness of the three proposed approaches hardware simulations have been implemented in Altera FPGA architectures using VHDL. These simulations have shown that the real increase in performance is 20-30 % lower than the corresponding theoretical one.

Overall the proposed combined CRC and CS architectural error control approaches ensure high reliability level and high performance of error control in the rate of the data transmission which can be used in high speed information systems and networks especially in military applications where the needs are for high level security on data integrity in different communication systems.

Acknowledgment

The authors are grateful to Apostolos Leros, Associate Professor of Automation Department at the Technological Education Institute (T.E.I) of Chalkida (Greece), for his support during the writing of this paper.

References

[1] Bardis E.G., Markovskyy A.P. “Utilization of of Avalanche Transformation for Increasing of Echoplex and Checksum Data Transmission Control Reliability” // 2004 International Symposium on Information Theory and its Applications (ISITA-2004).-Parma, Italy, Okt 10-13, 2004.- pp.656-660.

[2] Albertengo G.A., Sisto R.J. “Parallel CRC Generation” // IEEE Micro, Vol.11, № 10, 1990- pp.84-91.

[3] Fletcher J. An Arithmetic Checksum for Serial Transmissions // IEEE Transaction on Communication.-. Vol. 30.- № 1.- 1983,pp.76-85.

[4] Klove T., Korzhik V. “Error Detecting Codes: General Theory and Their Application in Feedback Communication Systems”. Norwell, MA: Kluwer, 1995. – pp. 433.

[5] Kounavis M.E., Berry F.L. “A Systematic Approach to Building High Performance Software-Based CRC Generators” // 10th IEEE Symposium on Computers and Communications (ISCC'05), - 2005.- pp. 855-862.

[6] Partrige C., Hughes J., Stone J. “Performance of Checksums and CRCs over Real Data” // Proc. SIGCOMM’95 Conf., ASM, 1995- pp.68-76.

[7] Bardis Nikolaos, “Coding of checksum components for increasing the control reliability of data transmission for military applications", WSEAS TRANSACTIONS on COMMUNICATIONS, Issue 7, Volume 11, ISSN 1109-2742, November 2008, pp. 1122-1131.

[8] Bardis N.G, “Echoplex Error Control System using Avalanche Transformations”, WSEAS TRANSACTIONS on COMMUNICATIONS, Issue 2, Volume 3, ISSN 1109-2742, April 2004, pp: 741 – 745.

[9] Shu Lin and Daniel J.Costello, Jr, “Error Control Coding”, Prentice-Hall, Inc., Englewood Cliffs, N.J.07632, ISBN 0-13-283796-X, 1983, pp. 603.

[10] Klove Torleiv, “Codes for Error Detection, Serial on Coding Theory and Cryptography” – Vol.2, World Scientific, 2007, pp. 201.

[11] Bardis N.G, Bardis E.G., Markovskyy A.P., C.Economou, “Hardware Implementation of Data Transmission Control based on Boolean Transformation”, WSEAS TRANSACTIONS on COMMUNICATIONS, Issue 7, Volume 4, ISSN 1109-2742, July 2005, pp: 363 – 371.

[12] Doukas Nikolaos, Bardis Nikolaos, “Effectiveness data transmission error detection using check sum control for military application”, A Series Of Reference Books And Textbooks archive Proceedings of the 10th WSEAS international conference on Mathematical methods, computational techniques and intelligent systems, Corfu, Greece, ISBN ~ ISSN:1790-2769, 978-960-474-012-3, 2008, pp 498-502.