A Calculator Model for the Estimation of Noise Immunity of Trans-Ionospheric Communication Channels, Based on the Theory of Residue Number Systems

Vitaly Grankin
North-Caucasus Federal University
Russian Federation

vta0@yandex.ru

Gennady Linets
North-Caucasus Federal University
Russian Federation

kbytw@mail.ru

Roman Taranov
North-Caucasus Federal University
Russian Federation

roman_boxx@mail.ru

Department of Information Systems & Technologies

Abstract

In this article we try to propose a new model of an application-specific integrated circuit (ASIC) for calculation of the error probability in the trans-ionospheric channel. The new ASIC based on using of residue number system and this model investigates advisability of developing such hardware. While travelling through the ionosphere the signals of the space communication systems are subject to modifications in amplitude, phase and polarization. Thats which leads to intermittent reception, high error rate and weak noise immunity. In prospect by using of the analytical error probability, we can set to transmitter frequency and timings parameters of a signal to do preemptive correction and avoid weak noise immunity.

1 Introduction

The problem of determining the noise immunity of trans-ionospheric channel is a very important one because in the modern world space communications plays a major role in many economic activities. Its also important in applications of military and scientific fields. While travelling through the ionosphere the signals of space communication systems are subject to modifications in amplitude, phase and polarization [Shev15]. That is which leads to intermittent reception, high error rate and weak noise immunity. This negative influence increased especially in periods of ionospheric disturbance and more effects on trans-ionospheric channel. In papers [Pash06]

In: S. Hölldobler, A. Malikov, C. Wernhard (eds.): *YSIP2 – Proceedings of the Second Young Scientist's International Workshop on Trends in Information Processing, Dombai, Russian Federation, May 16–20, 2017*, published at http://ceur-ws.org.

considered, that to increase noise immunity of communication systems, correction of timings and frequency of signal must be calculated in real time, but in modern conventional systems it is almost impossible.

It is known that the propagation time of the signal from geostationary orbit is theoretically not be less than 240 ms, and consider that signal processing and switching, can make delay above 400 ms [Yon97]. Delay is critical in real-time systems and presents problems for a latency-sensitive applications such as voice communication, and under the described conditions, pause between voices will be at least 500 ms, by the way for a public telephone allowable delay is considered to 150-250 ms [Cis10]. In case of ionospheric disturbances, noise immunity of system will reduced dramatically and communication session will be lost.

Prospective communications systems have to monitor ionosphere state continuously in real time mode by passive listening of radio tracks and have to set correct frequency and timing parameters of signal before the conversation begins. Thus can avoid communication lost, especially in case of ionospheric disturbances. However, an insertion of corrections in a real-time mode introduces high demands on calculator speed.

Considering of the total delays of - propagation, processing, calculations, and corrections, required that an overall delay was minimal, and the only reserve of its reduction is hidden in computation process. It is known that cost-effective way to improve calculation speed is a parallel calculations.

In this paper described a model of an application-specific integrated circuit (ASIC) for an application in modern space communication systems. Scheme of the ASIC based on natural mathematical parallelism of residue number system (RNS).

2 Problem Analysis

Noise immunity plays a major role in communication systems, which characterizes endurance from many negative factors such as interference, Rayleigh fading, frequency-selective fading etc. Analysis of space communication field determines that as the noise immunity factor may be the error probability in received signal. Most systems require that value of the error probability would not be greater than permissible $P_{er} \leq P_{eral} = 10^{-5}$ [Gar09, Jac14, Sre16]. Currently this factor evaluated by statistical methods by counting an errors in received data symbols. Such an approach has several disadvantages, such as sufficiently large time intervals with data acquisition and hence a large inertia of the processing of statistical data, which leads to the fact that the data are not relevant in conditions of ionosphere changes and spacecraft motion.

In the paper [Pash06] proposed that the ionospheric fading are characterized by Rayleigh and Rice distributions, and the most interesting equations, describes a frequency-selective fading (FSF) and intersymbol interference (ISI) of signal. Also proposed an analytical evaluation method:

$$P_{er} = \psi(h^2, \gamma^2, \eta_c, \eta_m) \tag{1}$$

where h^2 is the relationship of signal noise ratio in antenna, γ^2 is the coefficient of Rayleigh fading characteristic, $\eta_c = \psi(F_0/F_k), \eta_m = \psi(1/T_sF_k)$ coefficients of energy fading in condition of frequency-selective fading (FSF) and intersymbol interference (ISI) of signal respectively, duration of signal T_s, spectral width F_0, coherence bandwidth F_k.

Coefficients of energy fading described [Pash06]:

$$\eta_c = \left(1 + \frac{1}{2\pi}\left(\frac{F_0}{F_k}\right)^2\right) erf\left(\frac{\pi F_k}{F_0}\right) + \frac{1}{\pi\sqrt{\pi}}\left(\frac{F_0}{F_k}\right) exp\left(-\left(\frac{\pi F_k}{F_0}\right)^2\right) - \frac{2}{\pi\sqrt{\pi}}\left(\frac{F_0}{F_k}\right), \tag{2}$$

$$\eta_m = \frac{1}{2\pi^2}\left(\frac{1}{T_sF_k}\right)^2 erf\left(\pi T_sF_k\right) - \frac{1}{\pi\sqrt{\pi}}\frac{1}{T_sF_k} exp\left(-(\pi T_sF_k)^2\right). \tag{3}$$

The disadvantage of this method is the high latency computing of the coefficients η_c, η_m, due to the complexity of calculation of elementary and special functions.

The error function (the Gauss error function) is a special function defined as:

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{t^2}\, dt \tag{4}$$

this integral cannot be evaluated in closed form in terms of elementary functions, which lead than no path to direct computing. Perhaps integral form can be represented as a Taylor series expansion, however, accuracy quite small; in interval of $x > 4$ it takes a series of 54 degrees for the accuracy of given task. Speed of convergence of such series decreases to the edges of $erf(x)$.

An analysis showed that the calculators based on different processing methods have different performance and hardware costs and their implementation is ambiguous. Considering of high cost and complexity of developing practical hardware, we can conclude for need in a computer modeling of calculator, as necessary comparison on speed and hardware costs. For example, serial adder has a greater delay but has a small hardware costs. On the other hand, a parallel adder or carry-lookahead adder is substantially less computation delay but much higher hardware costs. For the full carry-lookahead adder, costs are comparable with costs of implementing of the entire device.

Calculators based on theory of conventional and modular arithmetic are not equal for basic principles and its hard to compare. Lets consider the main differences between conventional numeric system (CNS) and residue number system (RNS) in the construction of hardware calculators:

- In the RNS are not determined operations of division and comparison, and we need in using of complicated methods.

- Some operations such as multiplication in RNS requires costly resource scaling, on other hand in CNS scaling is performed by trivial bit shift.

- After a series of the order change operations such a multiplication, in the RNS in order to avoid overflow should be built calculators with significantly overweight range and must use basis extension methods.

Therefore, the comparison of calculators in terms of word size is not correct, since the fundamental differences between numeric systems. Word size should be selected by the desired task, accuracy and numeric range.

3 Problem Statement

Suppose for the problem statements, we need to evaluate the noise immunity factor of trans-ionospheric channel and provide the calculation of the coefficients of energy loss (2, 3) for calculating of the error probability (1) of signal.

We need:

1. to carry out approximation of error function (4) with a given accuracy in the presentation of elementary functions in RNS,
2. to develop a mathematical model of calculator of noise immunity factor of trans-ionospheric communication channels. Calculator must be based on the representation of elementary functions in the RNS,
3. to make comparisons and evaluate hardware costs.

4 Problem Solutions

4.1 Error Function Approximation

In this paper, we used a Chebyshev polynomial, to reduce calculation delays in evaluation of the error function. Polynomial approximation is a preferred way for direct computing of the functions evaluation in way of using of RNS. Polynomial calculation provides choosing of accuracy on a predetermined range. To reduce approximation range we can use odd property of the error function $erf(-x) = -erf(x)$. The error function has no singularities and asymptotically approaches to 1, if required accuracy achieved at the edges, range should be limited. $erf(10) - erf(4) \approx 1.5 \cdot 10^{-8}$ it is enough for this task. In summary, the error function can be written in the approximate form:

$$erf(x) \approx \begin{cases} a_0 + a_1 x_1 + a_2 x_2 + ..., & x \geq 0, \\ -erf(-x), & x < 0. \end{cases} \tag{5}$$

Figure 1 shows example of approximation of $erf(x)$ function by using of Chebyshev polynomial with 4 degree in interval $x \in [0, 2]$.

In article [Gra15] proposed a representation and computation of the elementary function in RNS. By the transition from polynomial calculation to table computations by using RNS properties, achieved improvement of performance and decreased hardware costs. This method may be applied for evaluation exponential and error functions (4, 5) in RNS. For modules choose we used the criterion of potential losses [Mez15]. Comparison of models word sizes, given in Table 1.

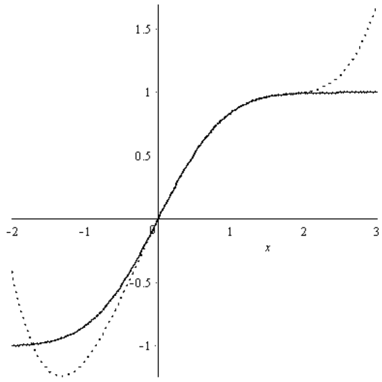

Figure 1: Example of Approximation by Using of Chebyshev Polynomial

Table 1: Word Sizes

	RNS	CNS
Word size	53 bits	32 bits
Effective size	≈ 48.36 bits	32 bits
Decimal digits	≈ 14.56	≈ 9.63
Range	$0 - 3.61 \cdot 10^{14}$	$0 - 4.29 \cdot 10^{9}$
RNS modules	3,5,7,11,13,17,19,23,29,31,59,61	
Modules size	2,3,3,4,4,5,5,5,5,5,6,6	

4.2 Developing of a Model

Modeling of a computing device is ambiguous. Various factors such as temperature and voltage are influence on the device performance [Sod07, Ham13, Oct06]. In addition, the performance of the devices implemented on the FPGA, depends on its architecture and project size [Alt08]. We choose length of propagation delay in computing process as criteria of efficiency of methods. Length measured as number of serial gates. Another criteria is a hardware cost, measured as number of total gates needed for implementation. Knowing the length l of propagation delay in number of serial gates and gate time t, define calculation time T:

$$T = l \cdot t \tag{6}$$

At the moment, there is no analytical expression of evaluation of delays and hardware costs of traditional elements such as: adder, multiplier, divider, calculating elementary functions.

General block diagram of calculating of noise immunity factor of satellite communication channels shown in figure 2.

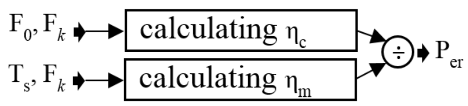

Figure 2: General Block Diagram of Calculating of Noise Immunity Factor

Implementation of the scheme on figure 2 in conventional numeric system required the following models of: adder, multiplier, divider, elementary function unit.

We used 4-bit carry-lookahead adder to accelerate transfer of carry bit. This scheme is the preferred in "delay/cost" criteria [Vah06]. We used a Wallace tree as a multiplier with high performance and low cost [Mit08], and traditional sequential division, in connection with the lowest cost.

Implementation of the scheme on figure 2 in residue number system required the following models of: modular adder, modular multiplier, elementary function unit, base expansion unit, scale unit, converter of CNS-RNS.

Length l or the path of propagation delay or gate delay expressed as the number of sequentially connected gates, shown in table 2 and 3 (for RNS). Hardware costs or number of gates for implementation of units, shown in table 2 and 3 (for RNS).

Table 2: Length l of Propagation Delay and Hardware Costs c of CNS Units

Adder		Matrix mul		Mul by Walles tree		Mul tree		Divider		Elementary functions	
l	c	l	c	l	c	l	c	l	c	l	c
12	337	373	10478	125	5984	61	11471	416	493	79	23680

Table 3: Length l of Propagation Delay and Hardware Costs c of RNS Units

Adder		Multiplier		Base extension		Scale		Elementary functions		Converting CNS-RNS	
l	c	l	c	l	c	l	c	l	c	l	c
5	1354	7	2533	75	11965	87	16386	89	16779	55	52806

Consider formulas for estimation of a model. Conventional adder:

$$\begin{aligned} l_s &= 12, \\ c_s &= 8 \cdot c_{sum} + 1c_{ls}(c_e + 3) + c_{carry}, \end{aligned}$$

where l_s length for convetional adder, c_s it hardware costs, c_{sum} numbers of elements of full 4-bit adder, c_e cost of carry scheme, c_{carry} cost of carry.

Conventional matrix multiplier:

$$\begin{aligned} l &= 12 \cdot (A - 1) + 1, \\ c &= A \cdot C_{sum32} - 1, \end{aligned}$$

where A is a word size (number of bits), C_{sum32} is a hardware costs of an adder.

Conventional multiplier, Walles tree:

$$\begin{aligned} l_m &= 4 \cdot A - 7 \\ c_m &= 6 \cdot A^2 - 5 \cdot A \end{aligned}$$

where A is a word size.

Conventional divider (implemented by multiple ticks scheme in reason of high costs of sequential implementation)

$$\begin{aligned} l_d &= A \cdot (l_{sum32} + 1) \\ c_d &= C_{sum32} + 1 + C_{cary} + 4 \cdot A \end{aligned}$$

where A is a word size, l_{sum32} length of adder, C_{sum32} is a cost of an adder, C_{carry} is a cost of carry bit.

Conventional elementary function calculator, represented by a parallel-connected comparators with common bus:

$$\begin{aligned} l_f &= l_{sum32} + 3 + S, \\ c_f &= (C_{sum32} + A + 1) \cdot S, \end{aligned}$$

where l_{sum32} is a length of an adder, C_{sum32} is costs of an adder, A is a word size.

Modular adder implemented by the scheme of carry-lookahead adder with tables of residues. It is compromise scheme in hardware costs and calculation speed. It formula is:

$$\begin{aligned} l_{sum} &= l_s + l_t, \\ csum &= \sum C_i + 2^{m_i+1} + 1, \end{aligned}$$

where l_s is a length of carry-lookahead adder, l_t is a delay of a table, C_i - costs of i-adder, m_i is a word size of i-modulus.

Modular multiplier implemented by a scheme of "a fast RNS galois field multiplier" [Rad90]:

$$\begin{aligned} l_{mul} &= l_m + 2, \\ c_{mul} &= \sum 3 \cdot (2^{m_i} + m_i) + C_{m_i}, \end{aligned}$$

where l_sum is a length of modular adder, m_i is a word size of i-modulus, C_{m_i} is a cost of modular adder of i-modulus.

Base extension implemented by a Garner's method[Gra16]:

$$\begin{aligned} l_{ext} &= l_{sum} \cdot (A_m - 1) + A_m + 8, \\ C_{ext} &= \sum C_{sum} + N \cdot (2^{a_i} + a_i) + n_m \cdot 32, \end{aligned}$$

where l_{sum} is a length of modular adder, A_m is a total word size of RNS, C_{sum} is a length of modular adder, a_i is a word size of i-modulus, N is a number of conversion tables, n is a number of adder by modulus.

Modular scaling implemented by a method of [Che05] and it formula is:

$$\begin{aligned} l_{scl} &= l_{ext} + l_{mul} + l_{sum}, \\ c_{scl} &= c_{ext} + c_{mul} + c_{sum} + c_{tab} + A_m, \end{aligned}$$

where l_{ext} is a length of base extension, l_{mul} is a length of modular multiplier, l_{sum} is a length of modular adder, c_{ext} is a costs of a base extension, c_{mul} is a costs of a modular multiplier, c_{sum} is a cost of modular adder, c_{tab} is a cost of conversion table, A_m is a total word size of RNS.

Elementary functions implemented by method of [Gra15] and it formula is:

$$\begin{aligned} l_{func} &= l_{scl} + 2, \\ c_{func} &= c_{scl} + \sum 2^{a_i} + a_i) \end{aligned}$$

where l_{scl} is length of modular scaling, c_{scl} is a costs of modular scaling, a_i is a word size of i-modulus,

Conversion from CNS to RNS is implemented by method of adding by the tree with 8 adders:

$$\begin{aligned} l_{tr} &= 11 \cdot l_{sum}, \\ c_{tr} &= (8 \cdot 4 + 4 + 2 + 1) \cdot c_{sum}, \end{aligned}$$

where l_{sum} is a length of modular adder, c_{sum} is a costs of modular adder.

Calculus of energy coefficients η_c, η_m in CNS:

$$\begin{aligned} l_{\eta_c} &= 3 \cdot l_s + 7 \cdot l_m + 2 \cdot l_f, \\ c_{\eta_c} &= 3 \cdot c_s + 7 \cdot c_m + 2 \cdot c_f, \\ l_{\eta_m} &= l_s + 8 \cdot l_m + 2 \cdot l_f, \\ c_{\eta_m} &= c_s + 8 \cdot c_m + 2 \cdot c_f, \end{aligned}$$

where l_s is a length of adder, l_m is a length of multiplier, l_f is a length of function, c_s is a costs of adder, c_m is a costs of multiplier, c_f is a costs of function.

Calculus of energy coefficients η_c, η_m in RNS is implemented with modular scaling for avoid of overflow and it formula is:

$$\begin{aligned} l_{\eta_c} &= 3 \cdot l_{sum} + 7 \cdot l_{mul} + 1 \cdot l_{scal} + 2 \cdot l_{func} + 2 \cdot l_{scal}, \\ c_{\eta_c} &= 3 \cdot c_{sum} + 7 \cdot c_{mul} + 3 \cdot c_{scal} + 2 \cdot c_{func}, \\ l_{\eta_m} &= 1 \cdot l_{sum} + 8 \cdot l_{mul} + 1 \cdot l_{scal} + 2 \cdot l_{func} + 2 \cdot l_{scal}, \\ c_{\eta_m} &= 1 \cdot c_{sum} + 8 \cdot c_{mul} + 3 \cdot c_{scal} + 2 \cdot c_{func}, \end{aligned}$$

where l_{sum} is a length of modular adder, l_{mul} is a length of modular multiplier, l_{scal} is a length of modular scaling, l_{func} is a length of modular funnction, c_{sum} is costs of modular adder, c_{mul} is costs of mofular multiplier, c_{scal} is a costs of modular scaling, c_{func} is a costs of modular function.

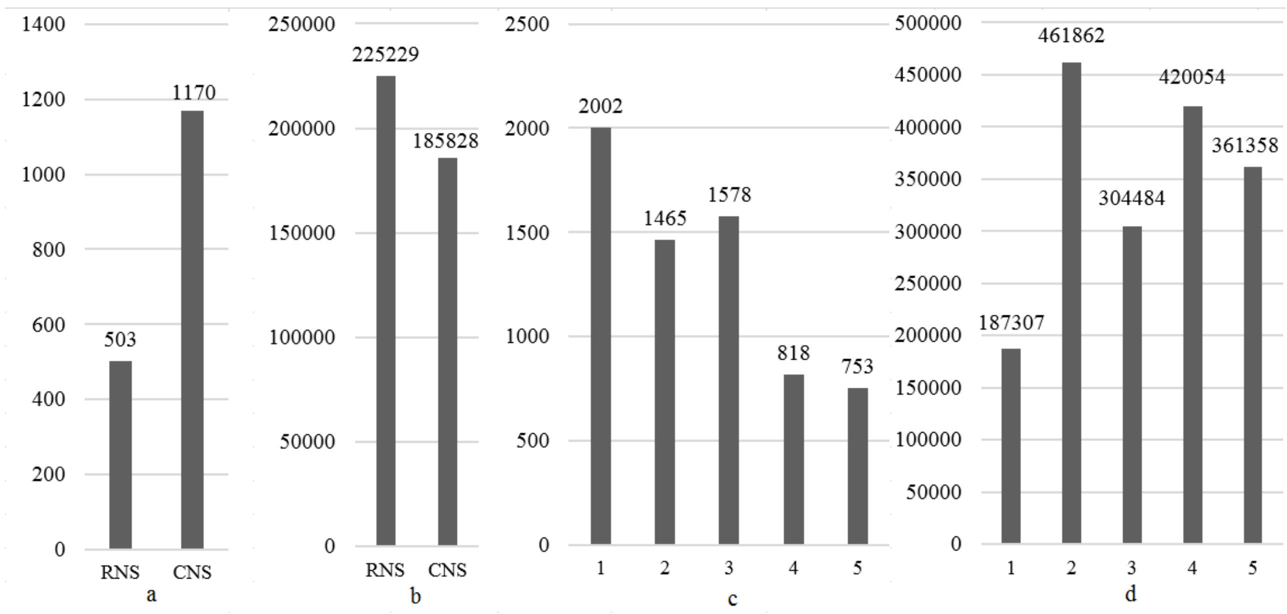

Figure 3: Length of Propagation Delay (a, c) and Hardware Costs (b, d) of Calculators (a, b) of Coefficients of η_c, η_m and (c, d) Noise Immunity Factor Calculator

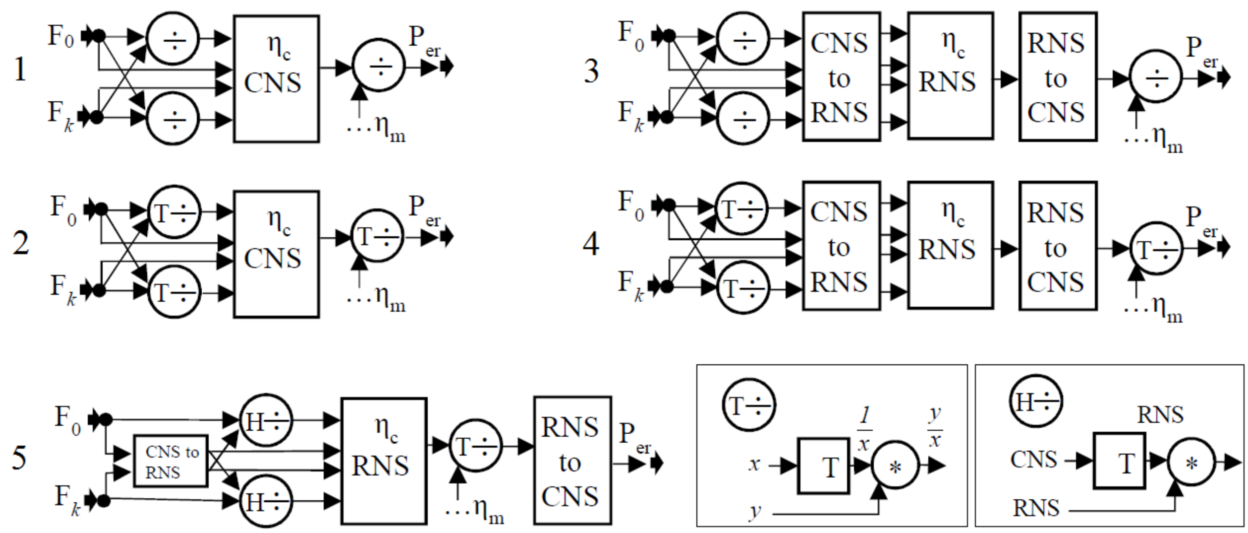

Figure 4: Block Diagram of P_{er} Calculator Schemes

Length l of propagation delay and hardware costs of coefficients of energy loses calculator is shown in figure 3 in sections a and b. Section c and d is for models of noise immunity factor calculator.

Implementation schemes of P_{er} calculator given in figure 4, that relevant with figure 3 (propagation delay in sections c and hardware cost in d).

5 Conclusion

1. Taken an approximation of error function by using of Chebyshev polynomial with required accuracy 10^{-7}.

2. Developed new mathematical models of the error probability calculator of trans-ionospheric channel. Models based on conventional and RNS representation of elementary functions.

3. By the modeling of noise immunity factor calculator, established that application of the theory of modular

arithmetic improves computational speed. Hybrid scheme (RNS with CNS) provides increasing of computational efficiency is more than 2 times, while increasing in hardware costs by 93%. This result explained by the high efficiency with using of modular structures in the given task.

4. On the basis of inductive inferences derived formula, that defines the length of the propagation path of signal in computation and hardware costs for CNS and RNS.

5. The analysis showed that calculator implemented by using of conventional numeric system has the lowest hardware cost (Fig. 3d position 1). However, calculating speed of that calculator is relatively small (Fig. 3c position 1). Practical interest is represented by a calculator with the lowest length of propagation delay (Fig. 3c position 5), constructed according to the hybrid scheme (by using CNS and RNS). The increase in performance is due to the increase in hardware costs, compared to the scheme that uses the traditional approach.

References

[Shev15] Shevchenko V., Pashitsev V., Chipiga A., Toporkov K. *Prediction of noise immunity of satellite by monitoring ionospheric scintillation index.* Infocommunication Technologies, Vol.13(4):365–375, 2015.

[Pash06] Pashintsev V., Maslov O. *Models of transionospheric radio channels and noise immunity of space communication systems.* Appendix to the journal "Infocommunication Technologies", Samara: PGATI, 4:357, 2006.

[Yon97] Yongguang Zhang, Dante De Lucia, Bo Ryu. *Satellite Communications in the Global Internet: Issues, Pitfalls, and Potential.* The Internet: the global frontier, The Seventh Annual Conference of the Internet Society, 1997.

[Cis10] Cisco systems *Voice quality.* http://www.cisco.com/c/ru_ru/support/docs/voice/voice-quality/5125-delay-details.html San-Francisco, 2010.

[Gar09] James L. Garrison *A method for detecting ionospheric disturbances and estimating their propagation speed and direction using a large GPS network.* Radio science, Vol.42, doi:10.1029/2007RS003657, 2007.

[Jac14] James L. Garrison, See-Chen G. Lee, Jennifer S. Haase, Eric Calais *Statistics of ionospheric disturbances and their correlation with GNSS positioning errors at high latitudes.* Space Weather Space Clim, A27 DOI:10.1051/swsc/2014024, 2014.

[Sre16] V. Sreeja *Impact and mitigation of space weather effects on GNSS receiver performance / Sorting and Searching.* Journal of the Asia Oceania Geosciences Society, (AOGS)20163:24 DOI: 10.1186/s40562-016-0057, 2016.

[Gra15] Grankin V. *Development of a tabular method of calculating of the elementary functions in RNS.* Vestnik of NCFU, Stavropol, 1(46):24–30, 2015.

[Mez15] Grankin V., Mezentceva O. *Substantiation of criteria for modules selection of RNS for implementation of compute nodes on the FPGA.* Vestnik of NCFU, Vol.3, (48):8–16, 2015.

[Sod07] G. Sodini. Judy Hoyt, Jing Kong Ke Lu, Ivan Nausieda, Palakodety R. *Microelectronic Devices and Circuits / Lecture 13 Digital Circuits (III) CMOS CIRCUITS.* http://web.mit.edu/6.012/www/, MIT, 2007.

[Ham13] Hambley A.R. *Electrical Engineering: Principles and Applications.* Pearson Education Inc., Upper Saddle River, New Jersey, 2013.

[Oct06] Octavian Florescu *Introduction to Microelectronic Circuits.* http://inst.cs.berkeley.edu/˜ee40/su06/ Berkeley, EE40, 2006.

[Alt08] Altera *OpenCore Stamping and Benchmarking Methodology.* Altera, 2008.

[Vah06] Vahid Frank *Digital Design.* John Wiley and Sons Publishers. ISBN 0-470-04437-3:296-316, 2006.

[Mit08] *Introductory Digital Systems Laboratory.* MIT, DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE, 6.111, 2008.

[Rad90] Damu Radhakrishnan *A fast RNS galois field multiplier*. Department of Electronic, DOI: 10.1109/ISCAS.1990.112619. Circuits and Systems, IEEE International Symposium on Source: IEEE Xplore, 1990.

[Gra16] Grankin V. *Analytical form of the Garner method for expanding of basis of the RNS*. Vestnik of ASTU, Astrakhan, 3:53–60, 2016.

[Che05] Chervyakov N., Mezentseva O., Lavrinenko I., Sivoplyasov D. *Method of expanding the dynamic range of a modular neurocomputer*. Neurocomputers: development & application, 7:64–69, 2005.