NATURAL DEVELOPMENT OF THE RESOURCES IN DESIGN AND TESTING OF THE COMPUTER SYSTEMS AND THEIR COMPONENTS

Julia Drozd[1], Alex Drozd[1], Svetlana Antoshchuk[1], Vyacheslav Kharchenko[2]
[1]Odessa National Polytechnic University, Odessa, Ukraine
[2]National Aerospace University named after N.E. Zhukovsky "KhAI"
drozd@ukr.net, svetlana_onpu@mail.ru, V.Kharchenko@khai.edu

Abstract

This work addresses to problems in design and testing of the computer systems and their components, which are considered in content of natural development of the resources for solving these challenges. The resources are examined as target and natural. The target resources contain the models, methods and means. The natural resources are considered like their particularities. Development of the resources is examined taking into account particularities of Universe which demonstrates itself as parallel, approximate and existing in on-line modes. Development of the models on example of arithmetic operations improvement from complete up to truncated ones is considered. The natural way of the methods development is shown examining the process of expansion of the results preparation method which is parallel and approximate.

Keywords*—computer system; development of resources; truncated operation; results preparation method.*

1. Introduction

The computer technologies have passed a way of prompt development during surprisingly short historical period. Inside the Universe the artificial computer world is created. The got experience allows not only to assess the computer world and the gone way, but also to make the certain forecasts of the further development. Such analysis can be executed representing development as process for solving a set of challenges in synthesis and the analysis. Concerning to the computer world they are problems and tasks of designing and an estimation of the received decisions accordingly [1, 2].

The decision of a task provides performance of set of the works during limited time with reception of reliable results. It becomes possible with use of the certain resources. Performance of set of the works, for example, calculations, during limited time demands the decision of a task with the certain productivity. Required reliability of results is provided at achievement of the certain degree of their adequacy to Universe. Resources include all necessary for solving a challenge: models, methods and means. Therefore development of the computer world can be considered as development of its resources.

In the given work an attempt to track development of resources is made. On belief of authors this development occurs naturally: resources are structured under particularities of Universe which is shown itself as both parallel and approximate [3].

Particularities of Universe are demonstrated by the example of the computer world.

Growing both parallelism and limited accuracy of decisions can be considered, analyzing development of personal computers. They have passed a way from series-parallel calculations to parallel, and also from hardware processing the approximate data on coprocessors with non-obligatory delivery up to use of sets of internal floating point pipelines in structure of the central and graphic processors [4, 5].

In section 2 both target and natural resources for solving the challenges are considered. Development of models on the example of perfection of the arithmetic operations is shown in section 3. Section 4 is devoted to questions of development of the methods, including a method of results preparation.

2. Resources for solving the challenges

The resources used for solving a task, further refer to target as well as the task carry a special-purpose character.

Target resources can be classified on models, methods and means. Models and methods concern to an information part of resources, and means belong to technological one. Models are our representations about the Universe and its components. Methods describe the transformations which are carried out with resources. Means provide realization of these transformations. Means include materials used for it and tools. The initial data concern to materials for performance of calculations as well as a code for programming a chip, this chip, the electric power, time, etc. The computer-aided design system can be considered as the example of the tool.

Target resources are a cost-based part of the decision of a task. First of all it is related to technological resources. Payment of information resources can enter cost of the tools developed with use of valuable models and methods. Information resources can be also freely distributed as are paid by work of the previous generations of researchers.

Decision of a challenge can use not only paid target resources but also the natural resources for which it is not necessary to pay as they are given to target resources as their particularities.

Gratuitous character of natural resources makes their attractive. At the same time, for use of natural resources it is necessary to learn to reveal them, and also to understand, that are they.

Two kinds of natural resources are most known. There are natural information redundancy and natural time redundancy used in on-line testing of digital components [6, 7].

Definition of natural resources as particularities of the target resources essentially expands set of their kinds. As a rule, the challenge is solved stage by stage. Particularities of the target resources involved at the previous stages can be used as natural resources at the following stages.

It is necessary to note, that all target resources develop under the control of the same particularities of Universe. Therefore target resources in process of natural development draw closer and, as it will be shown further, demonstrate natural resources becoming consistent.

For example, target resources for maintenance of fault tolerance and productivity of computer systems develop in the same direction of duplicating of operational elements and perfection of functions of a choice of result from parallel branches of calculations with use of technologies of reconfigurable computing [8, 9].

3. Development of models

The person thinks using models which are in permanent process of structurization under realities (or particularities) of our Universe. Such perfection of models can be considered by the example of development of the arithmetic operations.

Initially the arithmetic operations were carried out using consecutive or series-parallel codes with calculation of complete result. Frequently some of the arithmetic operations were executed in structure of the arithmetic-logic unit, using the same equipment.

Performance of arithmetic operations was improved at both system and circuit levels.

At a system level the paralleling of computing processes has led to performance of each arithmetic operation on single computing unit.

At a circuit level the basic way of development is connected with processing of operands in parallel codes using simultaneous devices, such as iterative array multiplier and divider.

Simultaneously with paralleling of calculations there was a process of their displacement in a direction of processing of the approximate data.

Various requirements to accuracy and to a range of representation of the approximate data have led to development of floating point formats where word sizes of a mantissa and an exponent provide necessary accuracy of calculations and volume of a range accordingly [10].

For the exact data accuracy and volume of a range are rigidly connected among themselves by word size of a binary code.

Feature of floating point formats is single accuracy when the mantissa of result inherits word size of a mantissa of an operand. It has led to creation of the new arithmetic operations calculating results with limited word size [11].

For two-place operations the result of complete operation is approximated with loss of a low half of calculated result. It follows from the record of the approximate floating point number which contains an operation of multiplication. Therefore multiplication in various forms is present at all operations with mantissas, and results of these operations inherit properties of product, in particular, the double size in comparison with the size of an operand.

Data processing with single accuracy creates in simultaneous devices a condition for transformation of complete operations into truncated ones which allow reducing volume of calculations.

In figure 1 the product conjunctions array (PCA) for the complete and truncated multiplication of binary codes $A\{1 \div n\}$ and $B\{1 \div n\}$ of mantissas $A = A\{1 \div n\} \cdot 2^{-n}$ and $B = B\{1 \div n\} \cdot 2^{-n}$ is shown ($n = 8$).

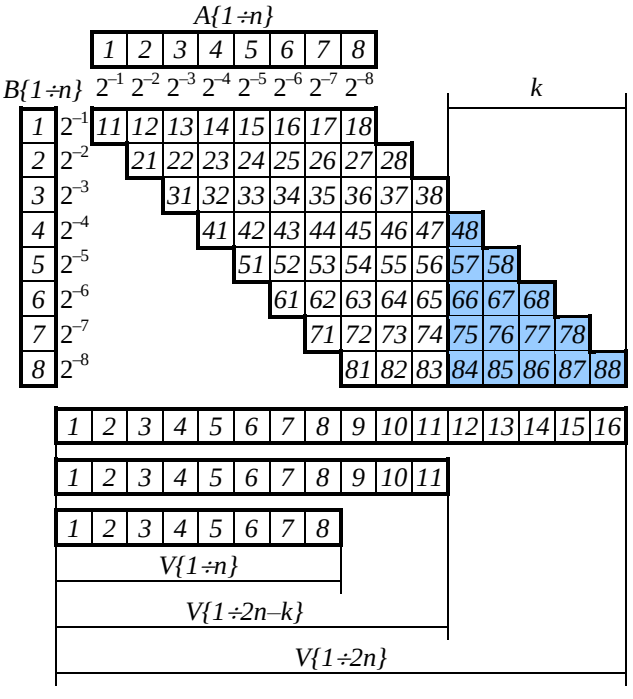

Figure 1: Performance of the complete and truncated multiplication

On the PCA the binary code $V\{1 \div 2n\}$ of a mantissa $V_F = V\{1 \div 2n\} \cdot 2^{-2n}$ of complete product is calculated.

The bits of a mantissa having weights from 2^{-1} up to 2^{-8} are numbered, since the first, high bit in a direction of reduction of weights. Conjunctions of are designated by the two-bit codes composed from numbers of bits forming them of multiplicand and a factor.

The known method of the truncated multiplication divides the PCA into two parts: low and high. Low part contains the low $k = n - \log_2 n$ columns eliminated from calculations. The truncated product $V\{1 \div 2n - k\} \cdot 2^{-(2n-k)}$ by high part of the PCA is calculated. The high n bits of the truncated product compose approximated result $V_0 = V\{1 \div n\} \cdot 2^{-n}$, and rightmosts $V\{n + 1 \div 2n - k\} \cdot 2^{-(2n-k)}$ are discarded [12].

Value k is estimated from a condition of preservation of single accuracy when the low non-calculated part of the PCA does not exceed weight of the younger bit of the approximated n-bit result.

The truncated operation almost twice simplifies simultaneous devices and reduces time of executed calculations [13].

It is necessary to note, that in on-line testing the simultaneous devices for processing mantissas are characterized by probability of an essential error. Such errors are caused by faults in true bits of the approximated result, in particular, in eliminated bits of calculated product of mantissas.

Reliability of on-line testing methods (in checking the results) is estimated by the following formula [14]:

$$R = P_E P_D + (1 - P_E)(1 - P_D), \qquad (1)$$

where P_E is a probability of an essential error;

P_D is a probability of error detection.

Traditionally on-line testing of arithmetic operations is carried out using a residue checking by modulo three which provides detection of all errors caused by the most probable faults of simultaneous devices, i.e. $P_D = 1$.

In this case the formula (1) will be transformed to the following form: $R = P_E$.

The top bound of probability P_E is the attitude

$$P_{E\,MAX} = n / n_C,$$

where n can be considered like the top bound of amount of the exact bits in approximate result;

n_C is size of the calculated result.

For the complete and truncated operations of mantissas multiplication the size of the calculated result $n_C = 2n$ and $n_C = 2n - k$. It determines the top bounds of an essential error probability for these cases as $P_{E\,MAX} = 0{,}5$ and $P_{E\,MAX} = n / (2n - k)$, accordingly.

Hence, proceeding from estimation of the top bound $P_{E\,MAX}$ of probability P_E, reliability of on-line testing methods for complete operation does not exceed $0{,}5$, and almost twice raises for the truncated operation.

Residue checking of truncated operation executed in simultaneous iterative array multiplier of mantissas is considered in [15].

Similarly to multiplication both truncated operation of division and shift in a floating point addition have been developed [16, 17].

Thus, change of complete operation into truncated ones shows structurization of arithmetic operation model under parallelism and non-accuracy of the Universe.

Such natural development of a resource gives it a set of advantages. Simultaneous simplification and acceleration of the device is indicative, i.e. two parameters which were traditionally opposed each other are in common improved: expenses of the equipment and speed.

The considered example demonstrates activation of the natural resources when initial contradictions between target resources are removed in process of their natural development.

The natural way of the arithmetic operation development is shown in figure 2.

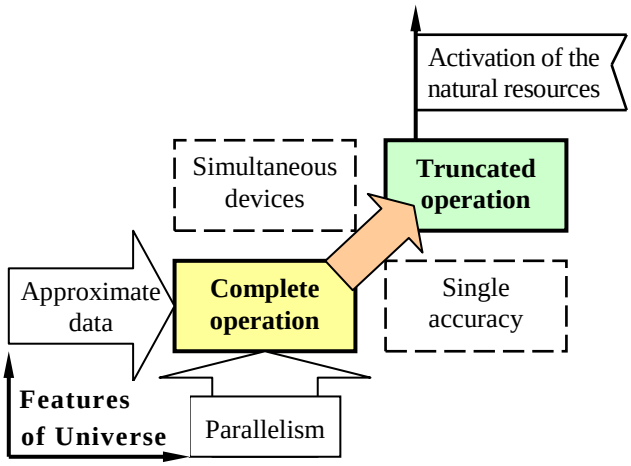

Figure 2: Natural development of the arithmetic operations

4. Development of methods

Natural development of resources is stimulated by the method of carrot and stick. A stick is natural selection, and carrot is gifts like realization of natural resources.

As example the success of technology CUDA using the graphic processor for performance of parallel calculations can be considered [18]. It based on the high level of multisequencing the calculations, focused on simultaneous processing of many thousand streams of the data, and also the approached character of calculations with use of floating point formats. Such natural development of the processor gives repeated increase of productivity and decrease of energy consumption in comparison with the traditional decisions realized in the CPU [19].

The example of technology CUDA again shows elimination of the traditionally established contradiction between productivity and energy consumption.

The important role in natural development of resources is played by a method of the results preparation which under various names actively captures space of decisions.

The method of the results preparation allows starting to solve a task before obtaining of all initial data, simultaneously (in parallel) with their formation. This determines approximate way for solving a task firstly receiving a set of possible results (Stage 1). One result is selected from set of possible results on receipt of the missing data by using of them (Stage 2) [20].

The use of the initial data (part 1 and part 2) in the two stages of the method execution is shown in figure 3.

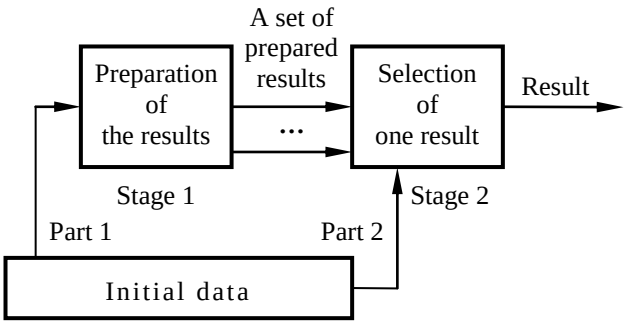

Figure 3: Execution of the results preparation method with use of the two parts of initial data

All modern co-design of digital systems and their components is based on a method of the results preparation. For example, every FPGA (Field Programmable Gates Array) chip is initially preform for set of projects, and the chip programmed under one project is preparation of results (for various input data) in the tables which have been written down in memory of LUT (Look-Up Table) [21].

Besides the most simple and high-speed (on half of bits of the address) the realization of a memory which is carried out on architecture 2,5 D, also is an example of using a method of results preparation [22].

In blocks of memory with structure 2,5 D hardware expenses for decoding of the 12-bit and 16-bit address are reduced by the method of results preparation above 21 and 85 times accordingly [23].

Due to structuring into a reality of world development FPGA-projects receive the features allowing to provide at a high level a set of characteristics: productivity of calculations and reliability of their results, universality, efficiency of designing, adaptability to manufacture, flexibility of decisions, and the most important advantage which is the combination of achievable levels testifying to their mutual consistency [24].

The specified advantages explain the wide-spread area of the challenges solved with use of FPGA-projects. For example, development of safety-critical instrumentation and control computer systems, which provide functional safety of objects of the raised risk (in power, on transport, in space, defensive and others areas) belongs to such challenges [25].

Preparation of results is used in personal computers at the decoding of instructions which are carried out on the pipeline device. Thus the sequence of instructions which is accepted to execution on linear sites of the program is prepared. Use of two decoding pipelines allows preparing instructions for two branches of the program missing from a point of check of a condition of branching. After performance of an instruction of checking a condition to execution the sequence of instructions of a corresponding branch of the program is accepted [26].

The software also is designed using the method of preparation of the program modules which get out necessarily.

Among the particularities of the Universe actively inherited by modern challenges and target resources, it is necessary to examine also on-line mode. Many processes, including testing of digital circuits or program components, were considered as carried out outside of on-line mode. Now both software and hardware products get on the market if pass testing in the certain deadlines, for example, 2 months. To satisfy this condition for the software products numbering (by virtue of a high level of parallelism) many thousand modules, testing is carried out in the limited volume with use of the approximated methods [27].

5. Conclusions

Models, methods and the means making target resources for solving the challenges in co-design and testing of computer systems and their components pass a natural way of development, structuring themselves under particularities of the Universe, including its parallelism, limited accuracy and on-line mode.

Such development of models has been considered on example of perfection of the arithmetic operations which have passed a way from exact complete operations up to approximated truncated ones. It became possible at paralleling calculations up to a level of their realization in simultaneous arithmetic devices, such as the iterative array multiplier or divider.

Simultaneous improvement of characteristics which traditionally were considered inconsistent is important. The use of the truncated operation in simultaneous device simultaneously reduces hardware overhead and operation time and also raises reliability of on-line testing. At use of technology CUDA concurrently productivity of calculations is raised and consumption of energy is reduced. The method of results preparation simultaneously simplifies procedure of results reception and reduces operation time.

Results pass a way of natural development from exact and correct up to reliable and approximate, i.e. adequate to the approximated Universe.

With increase of a level of decisions parallelism the exact methods giving representation about the Universe in its separate points cede the place to more adequate approximated methods which allow to see and estimate the Universe more widely (in areas).

Natural resources are particularities of the target resources and are shown in process of their natural development in elimination of contradictions in the decision of challenges.

The basic conclusion is the expediency of studying of Universe particularities and compliance with it during the decision of challenges, including problems in co-design and testing of computer systems and their components.

References

[1] G. Micheli, M. Sami, *Hardware / Software Co-Design*, Amsterdam: Kluwer, 1996.

[2] M. Nicolaidis and Y. Zorian, "On-line testing for VLSI – a compendium of approaches," *Electronic Testing: Theory and Application (JETTA)*, vol. 12, pp. 7 – 20, 1998.

[3] A. Drozd, S. Antoshchuk, "New on-line testing methods for approximate data processing in the computing circuits," *in Proc. 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, Prague, Czech Republic, 15 – 17 Sept, pp. 291 – 294, 2011.

[4] M. Guk, *Processors Intel: from 8086 to Pentium II*, SPb: Piter, 224 p., 1997.

[5] J. Owens, D. Luebke, N. Govindaraju, M. Harris, J. Kruger. A. Lefohn, T. Purcell, "A Survey of General-Purpose Computation on Graphics Hardware, *Computer graphics Forum*, vol. 26, no. 1, pp. 80 – 113, 2007.

[6] J. Savchenko, *Digital Tolerant Devices*, Moskow: Soviet radio, 176 p., 1977.

[7] A. M. Romankevich, V. N. Valuyski, V. A. Ostafin, *Structural Time Redundancy in Control Circuits*, Kiev: High School, Head publishers, 160 p., 1979.

[8] A. A. Ushakov, V. S. Kharchenko, V. V. Tarasenko, "Fault Toletant Embedded PLD-Systems: Structures, Simulation, Design Technologies," *in Proc. 10th Intern. Conf. MIXDES 2003*, Lodz, Poland, 26 – 27 June, pp. 546 – 551, 2003.

[9] R. Baxter, S. Booth, M. Bull, G. Cawood, K. D'Mellow, Xu Guo, M. Parsons, J. Perry, A. Simpson, A. Trew, "High-Performance Reconfigurable Computing – the View from Edinburgh," *in Proc. of NASA/ESA Conference on Adaptive Hardware and Systems AHS – 2007*, Edinburgh, UK, pp. 211 – 217, 2007.

[10] D. Goldberg, What Every Computer Scientist Should Know About Floating-Point Arithmetic," *ACM Computer Surveys*, vol. 23, no 1, pp. 5 – 18, 1991.

[11] W. Kahan, *IEEE Standard 754 for Binary Floating-Point Arithmetic*, Lecture Notes on the Status of IEEE 754, Elect. Eng. & Computer Science University of California, Berkeley CA 94720-1776, May 1996.

[12] Z. L. Rabinovich, V. A. Ramanauskas, *Typical Operations in Computers*, Kiev: Technika, 264 p., 1980.

[13] A. Ya. Savelyev, *Applied Theory of Digital Machines*, Moskow: High School, 272 p., 1987.

[14] A. Drozd, M. Lobachev, J. Drozd, "The problem of on-line testing methods in approximate data processing," *in Proc. 12th IEEE International On-Line Testing Symposium*, Como, Italy, 10 –12 July, pp. 251 – 256, 2006.

[15] A. Drozd, M. Lobachev and W. Hassonah, Hardware Check of Arithmetic Devices with Abridged Execution of Operations," *in Proc. European Design and Test Conf*, Paris, France, p. 611, 1996.

[16] A. Drozd and M. Lobachev, "Efficient On-line Testing Method for Floating-Point Adder," *in Proc.* of *IEEE Design, Aut. and Test in Europe*, Munich, Germany, pp. 307 – 311, 2001.

[17] A. Drozd, M. Lobachev and J. Drozd, "Efficient On-line Testing Method for a Floating-Point Iterative Array Divider," *in Proc.* of *IEEE Design, Aut. and Test in Europe*, Paris, France, p. 1127, 2002.

[18] NVIDIA CUDA Compute Unified Device Architecture. Programming Guide / Version 1.0, NVIDIA Corporation, 113 p., 2007.

[19] http://old.computerra.ru/interactive/423392/

[20] A. V. Drozd, M. V. Lobachev, J. V. Drozd, *Dedicated Architectures of Computers*, Learning aid, Odessa: Science and technique, 120 p., 2004.

[21] Netlist Optimizations and Physical Synthesis. Qii52007-2.0. Quartus II Handbook. Vol. 2. Altera Corporation, 2004.

[22] E. P. Ugryumov, *Digital Circuitry Engineering. Learning aid*, Learning aid, 3rd Edition, *SPb: BHV-Peterburg*, 800 p., 2004.

[23] A. Drozd, V. Kharchenko, S. Antoshchuk, J. Drozd, M. Drozd, J Sulima, *On line testing of the safe instrumentation and control systems*, A. Drozd and V. Kharchenko (Edits), National Aerospace University named after N.E. Zhukovsky "KhAI", 614 p., 2012.

[24] Design Optimization for Altera Devices. Qii52005-2.0. Quartus II Handbook. Vol. 2. Altera Corporation, 2004.

[25] E. S. Bachmach, A. D. Herasimenko, V. A. Golovir, V. S. Kharchenko, Yu. V. Rozen, A. A. Siora, V. V. Sklyar, V. I. Tokarev, *FPGA-based NPP I&C Systems: Development and Safety Assessment*, V. S. Kharchenko, V. V. Sklyar, (edits), RPC Radiy, National Aerospace University "KhAI", SSTC on Nuclear and Radiation Safety, 188 p., 2008.

[26] M. Guk, *Hardware of IBM PC: Encyclopaedia, 2rd Edition*, SPb: Piter, 928 p., 2003.

[27] O. V. Pomorova, T. A. Govorushchenko, "Analysis of Software System Quality Valuation Techniques and Means," *Radioelectronic and Computer Systems*, vol. 6, pp. 148 – 158, 2009.