

Forth in Russia

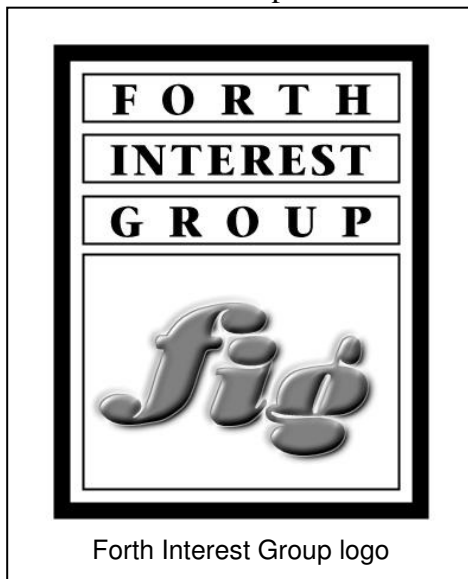
Sergey Baranov

St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
(SPIIRAS)

SNBaranov@gmail.com

This paper provides an extended version of a presentation made by the author at the International Conference “Development of Computing and Software in the States of the Former Soviet Union and Russia” SoRuCom-2011, held at the Yaroslav the Wise Novgorod State University (Velikiy Novgorod, Russia) on 12-16 September 2011 [27]. The purpose is to highlight the major milestones of Forth expansion in Russia since its early days, the current status of the Russia Forth community, and derive certain lessons learnt with an outlook into the future.

Forth became known in the USSR since the end of 1970-ies. After its appearance in the US – the first official publication dates to 1974 – this language and associated technique of



Forth Interest Group logo

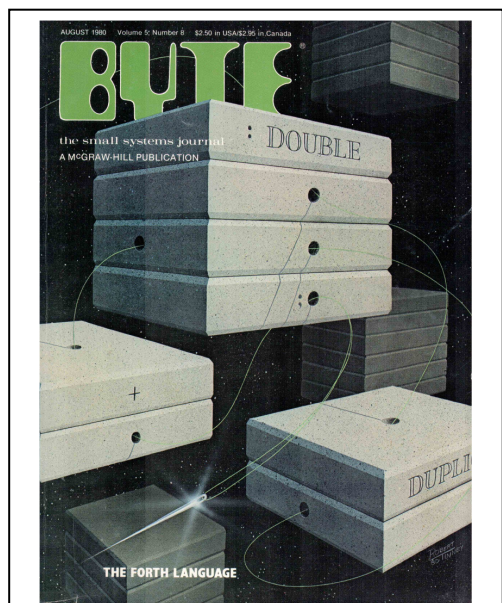
programming quickly won acknowledgement as a fast and efficient means for creating meaningful applications for microprocessor machinery, where efficiency of memory footprint and small program size were often vital. The Forth Interest Group [2] was soon founded, which is active still now, with the purpose to standardize this language and make it popular among software developers.

At that time a rapid growth of microprocessors was in place, a widespread Soviet one was K580IK80 – a sound copy of Intel 8080. At the Computer Center of the Leningrad State University (now the St. Petersburg State University) a team was established with an assignment to develop software for a new Soviet microcomputer with this processor under a contract with customers from industry. The team was headed by Associated Prof. Boris Katsev, PhD; he was a well-known specialist in computing

machinery, with talent, authority and organizing skills. Shortly before, he joined the University faculty after terminating his career in computer industry. Prof. Katsev staffed the team with young researchers and engineers of the Computer Center and faculty, the author being among them.

Contracts with leading Moscow industrial institutions NITSEVT and NIISCHEMASH for developing software of computer terminals with K580IK80 as the core processor were won for the University through Prof. Katsev's efforts. The mentioned institutions just started to develop such terminals for manufacturing for the whole Soviet Union. At that time the major computer facility of the Computer Center was a new mainframe ES 1030 (an analog of IBM/360) and old original Soviet ones M-220 and M-222, ending their life cycle. To test and debug software for K580IK80 the team decided to develop a cross-system for the K580 family, which included an assembler and a byte-code emulator.

PL/I was selected as an instrumental language for developing the cross-assembler, development took over a half-year. The resulting source code seemed to be enormous at that time (over 1000 lines in PL/I). All tasks running in parallel partitions of the IBM/360 compatible have to be shut down in order to provide the PL/I compiler with all available memory (less than 512K bytes at that time) for compilation of the cross-compiler code in one extended partition.



The Forth language special issue of
BYTE (August 1980)

Just at that time the team came across a copy of an article in the Dr. Dobb's Journal with a listing of an Intel 8080 assembler in Forth which took only 54 lines of text, one third of which being a table with recognizable mnemonics of Intel 8080 assembler instructions. Especially striking was the authors' claim that this was a complete assembler encompassing all modes and features of the Intel 8080 instruction set! The team spent considerable effort to clarify and understand how the assembler was done (with the CREATE-DOES> constructs); however, as soon as we got it, the power and beauty of this approach was greatly appreciated. As there were no other texts on Forth available at that time, the challenge was in grasping how this suite of Forth words worked as expected from just this listing. Only much later we accessed a special Forth issue of BYTE in 1980 [3] with many bright samples of how Forth may be used in various cases.

The most remarkable feature of Forth is its mechanism of introducing in a very elegant manner new basic language constructs, which render specifics of the task under consideration in the most appropriate way. Actually, Forth proposed a meta-language mechanism which allowed for creating new abstract data types along with their implementation at any level of abstraction, up to the machine code level. The latter allowed for reaching the maximal speed of code execution which was so important for many applications. For example, the fundamental notions of a variable and a constant may be quite elegantly introduced in just one line of text as:

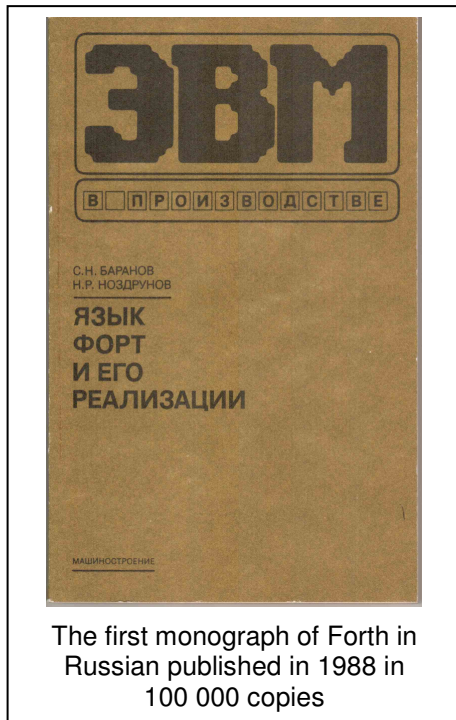
```
: VARIABLE CREATE 0 , DOES> ; : CONSTANT CREATE , DOES> @ ;
```

Having acquired that even fundamental control flow structures – branches and loops – may be so easily and simply expressed in the Forth core, we started to create quite exotic and unusual (at that time) control structures and experiment with them: switches, backtracking, exceptions handling and throwing, etc., based on the idea of a vector code field and manipulations with the return address and self-modifications of the running code.

The team immediately started to develop an implementation of Forth for ES 1030 (an IBM/360 compatible) – the only available computer at that time. An implementation in assembler was done remarkably fast, it was then used to bootstrap a Forth system from its baseline written in Forth; this source code was later published as an appendix to the first monograph on Forth in Russian [4], specifically aimed at industrial applications. At the same time another implementation of Forth was developed for the terminal ES-7970 with the K580 microprocessor [5], along with a number of utility applications for it.

The next phase started with implementing rather complicated projects in Forth. One of them was incorporated in the PhD thesis of Vyacheslav Kirillin “An Instrumental System for Developing Language Means of Microprocessor Machinery”, proved at the Leningrad State University in 1985. In particular, it contained descriptions of portable compilers from Pascal and Basic into K580, which worked on the terminal ES-7970 on top of its Forth system. Further popularization of Forth was contributed by Prof. Joseph Romanovsky at the Mathematical Department of the Leningrad State University [6]. A computer class was organized in form of a series of over 20 terminals connected to a powerful (at that time) ES mainframe (an IBM/370 compatible), where students could study Forth and experiment with it, surfing through an on-line Forth manual developed by Igor Agamirzian, Sergey Baranov, Vyacheslav Kirillin, and Nikolay Nozdrunov. While working at the terminal, students could create their own Forth programs, run them, and observe the results in parallel with reading manual sections. After studying Forth, it was much easier for students to learn PostScript and other interpretative programming languages provisioned at the curriculum.

At that period only the names of relevant technical journals were known in Russia: BYTE, Datamation, Dr. Dobb's Journal, and Forth Dimensions. We learnt about annual Forth conferences, held at the Rochester University, N.Y., by Institute for Applied Forth Research, Inc. However, to get access to these journals or to attend this conference seemed to be unrealistic. We



learnt that since 1983 the Institute published the Journal of Forth Applications and Research, that SIGForth (Special Interest Group) on Forth was established within the Association for Computing Machinery (ACM) with its periodicals SIGForth Bulletin and Newsletters, while in Europe annual euroForth conferences were regularly held by industrial companies and R&D institutions interested in Forth. A colleague of us, Alexander Sakharov, who worked at that time at the Library of the Academy of Sciences of the USSR, managed to provide within several years a subscription to the Journal of Forth Applications and Research, thus making this journal available to the interested people in Leningrad at least for reading at this open public library.

We found and contacted a team developing a dedicated Forth-processor [7] at the Institute of Cybernetics of the Estonian Academy of Sciences. The project was financed within a special contract with industry. As the leading engineer of the team Alexander Astanovsky came from Leningrad, we soon established common relations and interests of this group with our team in the Leningrad State University. Due to the efforts of these Estonian colleagues, a number of our R&D papers appeared in the collections of articles “Programming for Microprocessor Machinery” regularly published by the Institute of Cybernetics in Tallinn. In 1982 a Soviet conference on Forth was organized by Matti Tombak at the Tartu State University in Tartu, Estonia.

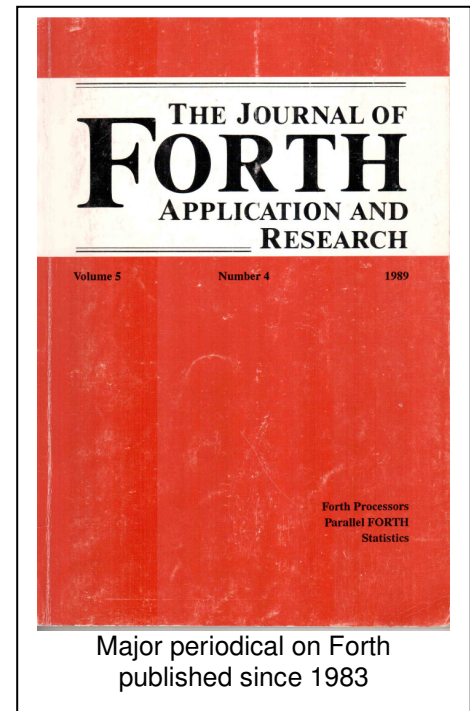
Approximately in 1980 within the framework of the Commission on Technology of the State Committee on Science and Technology headed by Prof. Igor Velbitsky of the Institute of Cybernetics of the Ukrainian Academy of Sciences, a Working Group on Microprocessor Machinery was formed, headed by Dr. Raivo Raud of the Institute for Cybernetics of the Estonian Academy of Sciences. In this Working Group a Forth division was quickly established with active participation and contributions from Vsevolod Kotlyarov, Sergey Baranov, Grigory Pogosiants, and Alexander Liberov.

Appearance of the already mentioned monograph [4], which incorporated accumulated experience of developing Forth systems, became a noticeably milestone on the Forth path in the USSR. Its first print was made in 50 000 copies; however, after receiving many requests from practitioners, the public house Mashinostroyenie (Machine-building) made another print of additional 50 000 copies, which turned out to be a rare case in its publishing practice!

A number of Soviet Forth systems, based on the standards fig-Forth and Forth-83, had already been known by that time: Forth-SM (S.Katsev, I.Shendrikov), Forth-Tartu (R.Viainaste, A.Yuurik), Forth-K580 (V.Kirillin, A.Klubovitch, N.Nozdrunov), Forth-ES (S.Baranov), Forth-Iskra-226 (G.Lezin), Forth-M6000 (V.Patryshev), Forth-BESM-6 (I.Agamirzian), Forth-Elbrus (A.Soloviev), Forth-Agat (A.Trofimov), to name just a few, which clearly demonstrates a great interest to this language and the respective programming technique among the Soviet software community. In the mentioned monograph, principles of Forth were systematically explained and demonstrated with the source code of a Forth core for the IBM/360 instruction set. Later, with appearance of personal computers, Forth systems for IBM PC and compatibles under MS-DOS were developed and widely deployed: Astro-Forth (I.Agamirzian) and Beta-Forth (S.Baranov).

In parallel with implementations of Forth as per se, various dialects of the language were developed, mainly for control applications. One of them – the Comfort system – was developed by a team at the St. Petersburg State Polytechnic Institute (V.Kotlyarov, N.Morozov, A.Pitko, S.Kireyev) for two families of 16-bit microcomputers Elektronika S5 and Elektronika 60, which were successfully employed in industrial control systems of various classes. In the Leningrad Construction Bureau of the state company “Svetlana” a chip for Forth and Comfort was developed and manufactured.

At the peak of “perestroika” (re-building) in the USSR in April 1988, through enormous efforts of Boris Katsev and Nikolay Nozdrunov the coop “Forth-Info” was established and registered in Leningrad, at that time it was one of the first coops in the area of programming and computing machinery. Employees of the laboratory of system programming of the Department of



mathematics and mechanics of the Leningrad State University with experience in Forth formed its core. Their direct task was creating and further developing of new programming techniques based on Forth. A noticeable outcome of the coop first three years was developing the 16-bit microprocessor Dofin-1610 for real-time control systems. The processor turned out to be the fastest one in its class of 16-bit processors in the USSR at that time. It displayed performance 50 times higher than its closest analog i8086 and was produced in small parties at the state company “Integral” in Minsk, Belorussia, mainly thanks to Prof.Katsev’s connections in the industry. Later the coop was transformed into an innovative and technology company “TechnoForth”.

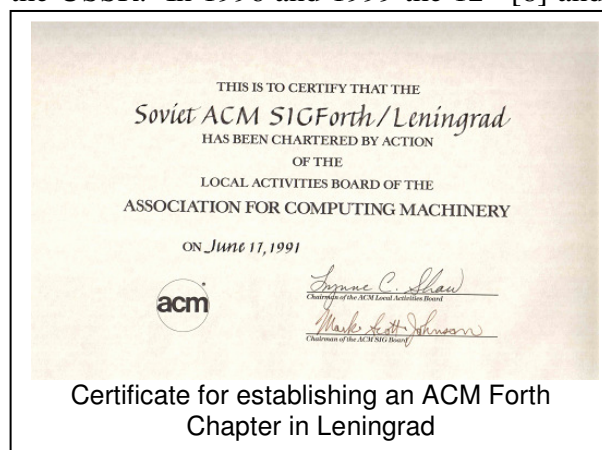
It so happened that the monograph [4] caught attention of Forth activists in the USA and the author received an invitation to come to the Rochester Forth Conference in July 1989. There were many technical hurdles in organizing this trip, which was handled through the Presidium of the USSR Academy of Sciences in Moscow; however, all was over thanks to the efforts and good will of the American colleagues. Within several following years thanks to established and strengthened connections, Soviet Forth-people could participate in the annual Forth conferences in the USA and Europe with their contributions and presentations of their accomplishments, exchanging experience and new ideas in this domain with the world “Forth elite”. In 1992 with support from the coop “Forth-Info” a group of students from the Leningrad State University was brought to Rochester, and after that they were invited as interns for summer practice in several US companies interested in Forth. That was genially new experience in this transition period of the Russian history! Companies in US and Europe, including MMS, MPE, Delta-T, Silicon Composers, and others demonstrated tangible interest to the “Russian experience” and established partnership with Soviet organizations and specialists.



Participants of Rochester conference in 1989



At that time conferences on Forth and its applications started to be regularly held in the USSR as part of activities of the Working Group on Microprocessor Machinery. Some of them were sponsored by the coop “Forth-Info” and some were held under financing from the state budget. Return visits of Director of the Forth Institute in Rochester Larry Forsley and Prof. Nicholas Solntseff of the McMaster University in Hamilton, Canada, took place. Their itineraries included visiting institutions in Leningrad, Moscow, and Novosibirsk. In 1991 the first issue of the Russian journal “Forth in Research and Development” under the aegis of the Leningrad State University; unfortunately with no continuation because of financial reasons due to the collapse of the USSR. In 1996 and 1999 the 12th [8] and 15th [9] annual conferences EuroFORTH were held in St. Petersburg, Russia, in the hotel “Rus”, organized by SPIIRAS with leading contribution from S.Baranov, I.Podnozova, E.Ignashkina, M.Kolodin, and M.Gassanenکو. In 1996 Michael Gassanenکو proved his PhD thesis “Mechanisms of Code Execution in Open Extendible Systems Based on Threaded Code” and then continued his research in this direction [10].



Certificate for establishing an ACM Forth Chapter in Leningrad

In 1991 a local ACM chapter on Forth was created in Leningrad, which worked for several years. Due to these activities, the chapter received subscriptions of all ACM periodicals (over 30 titles) that were deposited at the Library

of the Academy of Sciences with free access to the general public. This helped the Library to maintain the completeness of its repository when the state funding for purchasing technical literary nearly stopped because of the USSR collapse.

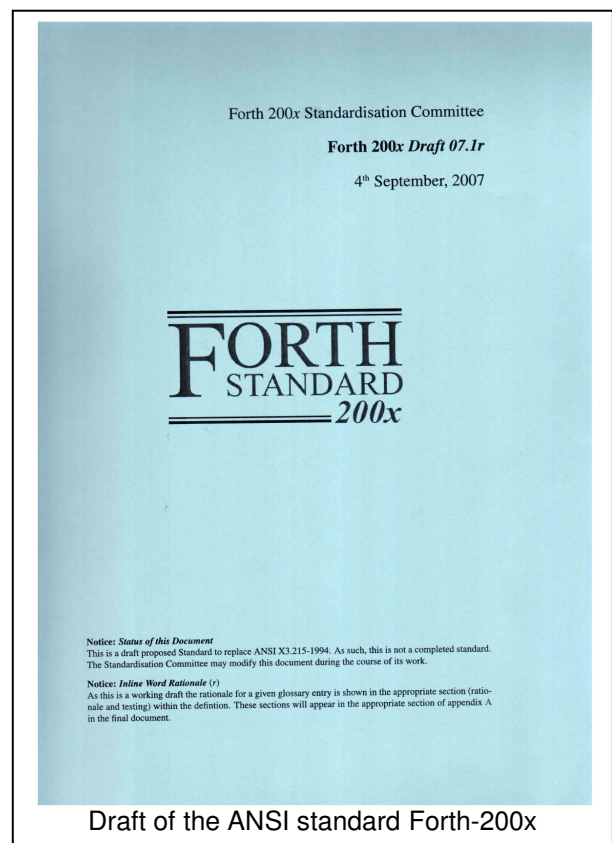
Later on, translations of books on Forth and its practical applications started to appear in Russia, such as [11], [12], [13], [14]. They stimulated further R&D in this area in Russia. For example, after learning about hardware implementation of cellular automata, which T.Toffoli and N.Margolus worked with at MIT [12], S.Baranov developed its software implementation of a cellular automata machine on top of his Beta-Forth system. Due to thorough programming of the main kernel, the resulting implementation Beta-CAM [16] displayed acceptable performance at the very first IBM PC with the processor i8080, which caused a surprise of those specialists when they saw its demonstration, inviting the author to visit MIT after the Rochester conference of 1991. Similarly, A.Kutuzov developed an expert system IBM PC based on the approach of C.Townsend [13], the system was later used to teach students at the St. Petersburg State Polytechnic University.

The second Russian monograph on Forth was that by V.Diakonov [15] published in 1992 at 30 000 copies, in 1993 followed by 50 000 copies of the monograph by Yu.Semyonov [17] where experience of the Institute for Theoretical and Experimental Physics in this area in this area was described. The valuable feature of this monograph was its appendices with the source code of a Forth interpreter for the processor Intel 80286 in the macro assembler MASM, implementing the standard fig-Forth, and a number of applications in Forth.

At the end of 1990-ies Andrey Cherezov [18] implemented his SP-Forth which is still in use in a number of Russian developments. A dedicated site of the Russian Forth community was established and is being supported [19], it stimulates further R&D and offers new ideas and solutions based on Forth.

Fallout of the USSR and transition of the former Soviet Union states to market economy made corrections to the status of Forth in Russia. The number of Forth enthusiasts and Forth addicts decreased, as the lion's share of programmers became employed in software companies with C/C++ and Java as their major instrumental languages. It's noteworthy that Java uses the same two-level structure of its code as Forth does: the source text is first translated into an intermediate representation (byte-code), which is interpreted by a Java virtual machine at run time. The Forth strength which allowed it to quickly fill-out its proper niche in software development for microprocessor machinery – direct access to all processor resources – at the same time turned out to be its vulnerability with respect to software security and safety. The Forth ideology – *everything allowed!* – opens doors to unwanted self-modifications of the executable code with penetration of software viruses and reduces code portability to other platforms. Java partially resolves this issue forbidding direct execution of machine instructions at the language level by protecting its assembler kernel from direct access from running applications via a complicated mechanism of their certification, while Forth can only rely on self-discipline and good will of the programmers.

Nevertheless, Forth developments continued to go on in Russia and in the rest of the world. Appearance of the Forth ANSI standard [20] which replaced the previous de facto standard Forth-83, became a strong argument for accepting Forth by the software industry. The new standard resulted from enduring efforts of the Technical Committee X3J14 which eliminated previous limitations of 16-bit address space and introduced the necessary ordering in the structure of the language. Since the beginning of 2000-ies



the standard is being revisited by a team of enthusiasts with support from companies which continue to use this language. Annual EuroForth conferences became the forum for their meetings, including the 29th one EuroForth 2013 in Hamburg [21].

One of the projects aimed at implementing the complete ANSI standard for IBM PC under MS Windows was carried out in 1994-1995 under a contract with Motorola, Inc. by the joint-stock company IBS created at SPIRAS and headed by S.Baranov at that time. V.Kirillin, A.Klubovitch, and D.Preobrazhensky who participated in this project, later became authoritative specialists in Java and its implementations. The project curator from Motorola was A.Sakharov; before that he left for US and was employed by Motorola.

At the end of 1980-ies S.Baranov developed a Forth-based technology of porting large size legacy programs to other platforms, which included automated building of a compiler of the source language from its grammar representation in formal regular expressions considered as texts in Forth [22]. With this technology a known system of symbolic computations SAC-2 written in a special algorithm description language ALSDES was successfully ported to IBM PC from CDC-6000 and IBM/360 mainframes. Due to additional strong type control, included into the porting technology, 2 errors in SAC-2 algorithms were found which went unnoticed for many year of using this system on mainframes. Based on the results of this and other adjacent research works, in 1990 S.Baranov proved his Doc.Sci thesis “A Forth-Based Technology for Porting and Implementing Large Computer Algebra Packages”.

Application of the Forth technology in school informatics may be also considered as an important accomplishment. In the beginning of 1990-ies an implementation of full Logo for the Russian school PC “Elektronika UKNTS” [23] was carried out under a contract with the St. Petersburg branch of the Institute for New Technologies. This product was used for some time in a number of schools in St. Petersburg and Moscow before a massive migration of IBM PCs occurred. Due to Forth features all computer graphics of Elektronika UKNTS worked remarkably fast, in spite of limited processor performance and small memory size. To minimize memory requirements of this implementation, a special mechanism of detaching the finished software product from its instrumental Forth system and making it a minimized stand-alone one [24] was developed. Starting from the main Forth word of the application (similar to the function `main` in C), only words from the Forth core were selected, which were referenced to from this word in the process of automatic construction of their transitive closure. Moreover, if dynamic search in the Forth vocabulary was not anticipated, then the vocabulary entries were deprived of their headers, making the entries “truncated” and consisting on the code field and parameter field only. Thus, the resulting applications became remarkably small – just 8K for a complete implementation of the Logo language.

Using the same approach, a version of the school micro PC Elektronika 31 was created at the already mentioned Construction Bureau of the state company “Svetlana”, the operating system and a programming system in Basic being implemented in the Forth dialect Comfort.

Interesting ideas in the same area of interpretative programming languages were developed by A.Baehrs in Novosibirsk. He proposed a notion of the “working mix” [25] which became a conceptual basis for developing software for the work station “Mramor”. His PhD thesis presented for viva in 1993 was unanimously qualified as a Doc.Sci work by examiners and members of the dissertation council, so his Doc.Sci viva took place in Moscow in the next year of 1994. On an advice from A.Baehrs and with his support a detailed analysis of the Forth phenomenon was made and published [26].

Summarizing, one can say that for nearly 40 years Forth continues to exist and attract talented programmers with its options “to do everything” with high quality, little effort, and quite fast. In spite of reduced share of implementations for embedded applications, it continues to find its champions and supporters and allows them to succeed in their developments in the current market environment.

References

- [1] Moore C.H. FORTH: A New Way to Program a Mini-Computer. – Astronomy and Astrophysics Supplement, 1974, vol.5. – P.497-511
- [2] Forth Interest Group – <http://www.forth.org/index.html>
- [3] BYTE, Vol.5, No 8, August, 1980

- [4] Baranov S.N., Nozdrunov N.R. The Forth Language and its Implementations. – Leningrad.: Mashinostroyeniye, 1988. – 156 p. (In Russian)
- [5] Baranov S.N., Kirillin V.A., Nozdrunov N.R. Implementation of Forth for the Display Terminal ES-7970. – In the collection: “Programming of Microprocessor Machinery”. Tallinn: Institute for Cybernetics, 1984. – P. 41-49 (In Russian)
- [6] Burago A.Yu., Kirillin V.A., Romanovsky J.V. Forth – a Language for Microprocessors. Leningrad.: Znaniye, 1989. – 36 p. (In Russian)
- [7] Astanovsky A.G., Lomunov V.N. A Processor Oriented to Forth. In the collection: “Programming of Microprocessor Machinery”. Tallinn: Institute for Cybernetics, 1984. – P. 50-67 (In Russian)
- [8] EuroForth 1996. – <http://www.forth.org/bournemouth/euro/index.html>
- [9] EuroForth 1999. – <http://www.forth.org.ru/~mlg/ef99/EF99repo.html>
- [10] Gassanenko M.L. A One-Stack Implementation of Backtracking for Forth. – Proceedings of SPIIRAS. 2002. Issue 1. Vol. 1. St. Petersburg.: Nauka, 2002. – P. 211–223 (In Russian)
- [11] Brodie L. Starting Forth. An Introduction to the Forth Language and Operating System for Beginners and Professionals. – Moscow: Finance and Statistics, 1990. – 352 p. (In Russian)
- [12] Toffoli T., Margolus N. Cellular Automata Machines: A New Environment for Modeling. – Moscow: Mir, 1991. – 280 p. (In Russian)
- [13] Townsend C., Feucht D. Designing and Programming Personal Expert Systems. – Moscow: Finances and Statistics, 1990. – 314 p. In Russian)
- [14] Kelly M., Spies N. Forth: A Text and Reference. – Moscow: Radio and Telecom, 1993. – 320 p. (In Russian)
- [15] Diakonov V.P. Forth Systems for PC Programming. – Moscow: Nauka, 1992. – 352 p. (In Russian)
- [16] Baranov S.N. Cellular Automata on a PC. Priroda, 1992, №9. – P.17-23. (In Russian)
- [17] Semyonov Yu.A. Programming in Forth. – Moscow: Radio and Telecom, 1991. – 241 p. (In Russian)
- [18] Site of Andrey Cherezov – <http://www.enet.ru/win/cherezov/> (In Russian)
- [19] Site of the Russian Forth-community – <http://www.forth.org.ru/news/> (In Russian)
- [20] American National Standard for Information Systems. Programming Languages. Forth. – <http://www.openfirmware.info/data/docs/dpans94.pdf> – 210 p.
- [21] EuroForth 2013. – <http://www.complang.tuwien.ac.at/anton/euroforth/ef13/>
- [22] Baranov S.N. Implementation of the MINISAC System for Symbolic Computations in Forth. – In collection of papers: Mathematical Methods of Constructing and Analysis of Algorithms Leningrad: Nauka, 1990. – P.3-15 (In Russian)
- [23] Baranov S.N., Preobrazhensky D.S. Logo in Forth. – Prolog, 3(5), 1993. – P.6-10 (In Russian)
- [24] Baranov S.N., Software Product Alienation in Beta-Forth. – In book: Problems of Software Engineering. St. Petersburg: SPIIRAS, 1992. – P.139-147 (In Russian)
- [25] Baehrs A.A. On Object-Oriented Aspects and Organization of the Architecture of Software Systems. – In collected articles “Actual Problems of Software Engineering”. Leningrad, LIIAN, 1989. – P.4-15. (In Russian)
- [26] Baranov S.N., Kolodin M.Yu. The Forth Phenomenon. – In book: System Informatics, issue 4, Novosibirsk: ISP, 1995. – P.193-271 (In Russian)
- [27] SoRuCom 2011 – <http://sorucom.novgorod.ru/>

About the Author



Sergey Baranov graduated with honor the Leningrad State University in 1972, worked at this university, at SPIIRAS, Motorola, St.Petersburg State Polytechnic University; PhD since 1978, Doc.Sci since 1991, Prof. since 1993. Currently works at SPIIRAS as a Chief Research Associate, teaches students at 3 major St. Petersburg Universities, and performs consulting at Motorola Mobility LLC. Major scientific interests are software engineering, compilers, analysis and verification of software specifications, formal methods, and symbolic

computations.