

## CHAPTER 6. THE MARS PROJECTS

### 6.1 Introduction

In this chapter we study the MARS computers and the organizations involved in their development. The MARS (Modular, Asynchronous, Extendable Systems) multiprocessor computers were part of the Soviet Union's START program, created in part as the Soviet answer to the Japanese Fifth Generation efforts. Centered in Novosibirsk, the work represents a very high-profile project carried out within the USSR Academy of Sciences (AN SSSR). Carried out under the patronage of G. I. Marchuk, chairman of the State Committee on Science and Technology (GKNT) and later President of the USSR Academy of Sciences, the START program as a whole and the MARS project in particular enjoyed some of the most favorable conditions within the Academy of Sciences—save perhaps in institutes with dual Academy/ministerial subordination—for developing new computers. While not necessarily typical of Academy efforts in computer development, therefore, the MARS project does represent something of a best-case. Many of the problems experienced in MARS development, particularly those regarding environmental factors, were shared by other Academy projects, but to an even greater degree.

Like most Academy of Sciences projects, the MARS computers incorporate a strong research component. The machines were viewed not only as vehicles for providing the scientific community with high-performance computing capability, but also as a means of experimenting with new architectural approaches. We therefore expect to see a different set of factors guiding this research than work done chiefly for industrial use. At the same time, being applied research, the work should strongly reflect the opportunities and constraints of the technological, organizational, and political context within which it was carried out. The MARS project can contribute to our understanding of high-technology R&D within the Academy of Sciences both prior to and during the reform period.

The MARS research took place within a changing organizational environment. The groundwork was laid within the Computing Center of the Siberian Department of the USSR Academy of Sciences (VTs SO AN SSSR). Much of the implementation work was carried out in the same facilities, but within a new organization, the START temporary scientific-technical collective (VNTK). Following the termination of START, a new—although more traditional—Academy of Sciences institute, the Institute of Informatics Systems (ISI), was created based on many of the Novosibirsk laboratories which had participated in START. Additional organizations reflecting a variety of organizational types were created as the *perestroika* reforms progressed. The organizational evolution surrounding the MARS project provides insights into the forces influencing organizational development within the Academy of Sciences during the Soviet Union's last decade and the relationship between technologies and the organizations within which they are developed.

This chapter is organized around the phases of development of the organizations involved in MARS research. Following a brief history of research at the Computing Center of the SO AN SSSR, we examine both organizational and technological developments during START's formation, the years of its existence, and during the period following its termination through the dissolution of the Soviet Union.

## **6.2 History of VTs SO AN SSSR Research**

The Siberian Department of the USSR Academy of Sciences, the first territorial department of the AN SSSR, was established in 1957 to provide a strong research base for the rapid development of Siberia. All Academy of Sciences research institutes east of the Urals were subordinated to the SO AN SSSR, and many new institutes were created. From 1957 to 1975 the chairman of the SO AN SSSR was M. A. Lavrent'yev; from 1975 to 1980, G. I. Marchuk; and from 1980 to the present, V. A. Koptug.

Half of the scientific personnel and research institutes are concentrated in the Akademik City (*Akademgorodok*) in Novosibirsk. The *Akademgorodok* is home to over twenty Academy research institutes, over 100 subsidiaries and design bureaus of branch scientific institutes, and—during the 1980s—a combined staff of over 70,000 individuals. The accumulation over the years of organizations associated with branch science has come to be known as the “implementation belt.” Its purpose is to facilitate the transfer of scientific results into production [Soan82].

Under the leadership of Marchuk and his successor, A. S. Alekseyev, the Computer Center of the SO AN SSSR became one of the major computer science research facilities in the Soviet Union. Marchuk, Alekseyev and others such as A. P. Yershov (one of the fathers of Soviet computer science) and V. Ye. Kotov had important voices in high-level computer-related policy commissions and committees within the Academy of Sciences and the State Committee on Science and Technology.

The principal areas of research of the Computer Center were the development of mathematical models of the atmosphere and oceans; methods of numerical analysis; computer applications for control processes in enterprise and territorial automated management systems (ASU); research and development of theoretical foundations for information processing; systems programming; development of techniques for mathematical computer-aided simulation; and the development of collective-use computing facilities shared by numerous *Akademgorodok* institutes. The methods and models enabled research in weather forecasting, general circulation of the atmosphere and oceans, climate theory, nuclear reactor analysis, and especially seismology [Soan87].

Until the late 1970s, virtually no work was done on designing and building computers. Much theoretical work, such as that of Vadim Ye. Kotov on the modeling of concurrent systems, and systems software development was carried out. The latter included

automated programming systems (Al'pha), multi-language translation systems (BETA), the first Soviet experimental time-sharing systems (AIST-0), and early translators for such languages as Algol-60, Al'fa-6, EPSILON, ALMO [Met173; Yers80b]. During the 1970s and 1980s the VTs SO AN SSSR was the principal organization in an effort to develop the *Sibir'* regional network, part of the larger academic network *Akademset'*, and became very involved in the development of such automated management systems (ASU) as Barnaul and view/Sigma [Alek83; Bobk78; Eko79; Mche85].

Prior to the late 1970s, the laboratories which were to become involved in START had done little applied work, and almost none in the area of high-performance computing. Following a change in the USSR Academy of Sciences' charter in 1977 encouraging more applied research [Fort90, 50], G. I. Marchuk began to conceive of his organization's becoming a pioneer in high-performance computing. To proceed past a paper design, projects within the Academy of Sciences had to gain technical, financial, and political support. Marchuk had close ties with V. S. Burtsev, the director of ITMVT at that time, as did Kotov. The three of them had served together on an Academy of Sciences commission which addressed supercomputing issues. Over the course of many conversations with Marchuk, Kotov, and others, Burtsev agreed to provide some technical support for a development team from Novosibirsk. As Marchuk moved up through the scientific bureaucracy, from director of the Computing Center of the SO AN SSSR to president of the SO AN SSSR (1975) and chairman of the GKNT (1980), he became increasingly able to push proposals into the top levels of government. As a result, it became increasingly feasible to carry out applied high-performance development in Novosibirsk.

In 1978, Marchuk and Kotov published a conceptual framework for a modular, asynchronous, extendable system (MARS) computer. The projects with origins in this work, discussed in section 6.6.2, marked the Computer Center's first serious attempts at applied

computer development and formed one of the cornerstones of the START program discussed below.

### **6.3 The MARS Conception**

The basic principles of MARS and some of the reasoning behind them were first spelled out in [Marc78; Marc78b] and were repeated in various forms over many years in publications about MARS research. A recent article published in the West explaining the origins of the MARS philosophy and design goals is [Koto91].

In [Marc78], Marchuk and Kotov presented their analysis of key trends in the development of computer technology, and identified one possible approach to the development of computers of the next generation. The principal trend, in their view, was a broadening of the sphere of application of computer technology, the compound or systemic character of the problems to be solved, and, consequently, the transition from single units of machines with a traditional (Von Neumann) architecture to computing systems with a variety of configurations and a wide range of capabilities and purposes [Marc78, 4,9,12]. Given this trend, a very relevant problem was the architecture of systems oriented towards multiple modes of use, from time-sharing to real-time processing, to information retrieval, etc. [Marc78, 14]. Extendable systems which could be adapted to a variety of application domains were of particular interest [Koto86b, 277; Koto91].

Providing specialization and adaptability to application domains would require “the selection of a base set of specialized units for information storage and processing, the creation of subsystems and devices which are programmable to specific operation modes and algorithms, and the development of the principle of reconfigurability of the system as a whole” [Marc78, 15]. At the basis of the architecture should be a theory of the analysis and synthesis of computational structures.

A second major trend was the miniaturization of the component base, making it possible to incorporate multiple microprocessors as a basic component of a computing system. It was felt that this trend would also result in fundamentally new means of system design, as the design of such components increasingly took on the characteristics of the design of large and complex systems [Marc78, 11].

A third trend was the steady improvement of the man-machine interface through higher-level programming languages.

Kotov and Marchuk felt that the architecture of promising computers of the future would be a logical extension of those ideas and principles which to one degree or another were already to be found in machines in existence at that time. Section 2 of [Marc78] identifies a host of Soviet and Western machines which are grouped according to architectural features relevant to the discussion. Their survey touches on general-purpose multiprocessors (Burroughs 7700, UNIVAC 1108, El'brus), vector-pipeline systems (Star-100, ASC), SIMD architectures (SOLOMON, ILLIAC-IV, PEPE, STARAN), and homogeneous computing systems with programmable interconnect systems. They discuss distributed memory and associative memory approaches, and asynchronous computational methods such as data flow.

The MARS Conception reflects an effort to fuse many of these ideas into a unified computing system, maximizing the strengths of each approach and minimizing the weaknesses. The key architectural principles, discussed in greater detail below, were [Marc78, 5,33-41]:

- parallelism (both in processing, data access, and control);
- decentralization of information processing and data flow;
- asynchronous interaction of devices and processes;

- hierarchical structure (both functionally and in terms of the components), with multiple virtual layers in the system;
- specialized systems components, hardware implementation of complex data processing functions;
- self-identification of data and processes (tagged architecture);
- modularity, reconfigurability.

**Parallelism.** Collectively, the machines surveyed exhibit parallelism at a number of different levels: parallel processing, parallel access to memory, and parallel control. Parallel processing can take the form of parallelism of iterations, independent operations on a set of objects, parallel processing of interacting branches, asynchronous parallelism, etc. Kotov and Marchuk tried to fuse many of the ideas into a unified computing system; a goal of the MARS project was to incorporate many of these kinds of parallelism in the same system, in an integrated fashion.

**Decentralization.** To accommodate growth in the number of processors or other devices without overloading certain system resources, some form of decentralized processing would be necessary, enabling subsystems to function autonomously on local data and communications. Marchuk and Kotov felt that the optimal variation would be the combination of the principles of centralized information processing with the capability of widespread decentralization [Marc78, 34].

**Asynchronous control.** They felt that asynchronous control was “necessary” for implementing the complex interaction of a highly parallel and (partially) decentralized system. Components would interact with each other through shared resources such as memory, buffers, communications channels, etc., but would not have direct control over each other. Kotov’s earlier work [Koto66] had developed ideas of a centralized asynchronous computation with shared memory. Other researchers, including Torgashev and My-

asnikov in the Soviet Union, and Miller, Dennis, and Rumbaugh in the West had worked on ideas of decentralized asynchronous processing in their work on recursive and data flow machines [Glus74; Denn68; Rumb75]. Seeking again to fuse many different threads of research into a single system, Marchuk and Kotov suggested that the optimal solution would incorporate the strengths of the centralized and decentralized approaches, minimizing their differences. As a result, the MARS model combined various asynchronous computational models.

**Hierarchical structure.** The computational models in effect at different levels differ in the nature of the activation conditions used. In particular, control mechanisms could be divided into three categories [Marc78b, 19-20]: unconditional, conditional, and data flow. These control mechanisms could be modeled using an extended Petri net notation<sup>1</sup> and trigger functions. Trigger functions are boolean expressions indicating readiness of some system component, be it an operation, expression, module, etc. [Koto66]. At the level of expressions with scalars and vectors, the data flow model is used. At the level of larger fragments (statements, program modules) which share common memory, an asynchronous control-driven mechanism based on trigger functions is used [Koto86b, 279].

Marchuk and Kotov divided the world of computational processes into two categories: user, or application processes, and systems processes. To handle the complexity of the contemporary systems used to run these processes better, they felt it useful to view a system as a series of nested “virtual machines” in which each layer represents a virtual

---

<sup>1</sup>Petri nets are a formalism for representing systems with concurrency or parallelism. A Petri net is a graph with two types of nodes—places and transitions. Places are drawn as circles while transitions are drawn as bars (or rectangles in [Marc78b]). Directed arcs (arrows) connect places to transitions and transitions to places. For each transition, the directed arcs define its input places (arc from place to transition) and its output places (arc from transition to place). A Petri net is executed by defining a marking and the firing transitions. A marking is a distribution of tokens to the places of the Petri net. A transition is enabled whenever all of its input places have one or more tokens. A transition fires by removing one token from each of its input places and adding one token to each of its output places.

machine which ‘‘runs’’ on a lower level virtual machine. They drew this concept from the work of Dijkstra on the THE multiprogramming system [Dijk68], and pointed to the IBM VM (Virtual Machine) operating systems as another example.

The functional hierarchy could also be reflected in the hardware itself, however. There were a number of examples of such systems. The central processors of the CDC 6600 and the Burroughs B6700 consisted of a number of subprocessors and registers. The ILLIAC IV design originally called for four array processors, each consisting of 64 processing elements. Marchuk and Kotov felt that some form of hierarchical organization was necessary to implement highly parallel computation most effectively.

**Specialized systems components.** Before the introduction of RISC concepts, the trend in instruction sets had been towards increasing numbers of instructions of increasing complexity. Marchuk and Kotov also felt that the most appropriate way to adapt a computer to specialized application domains was through the creation of application-specific, complex operations implemented either in hardware or in microcode. These measures, it was felt, would raise the level of the machine language, increasing programming effectiveness and reliability; decrease the portion of computation executed in software, improving performance; and increase the reliability and modularity of the entire system. Marchuk and Kotov point to a number of systems with hardware or microcode implementation of high-level operations, including the Burroughs B5700/B6700, FORTRAN and SNOBOL machines, and machines including hardware support for operating systems and database operations [Marc78, 38-39].

**Tagged architectures.** Tagged architectures, allowing low-level specification of data types, information about origins of operations, labels, state of readiness of data, etc. could be used to provide considerable control at low levels in the architecture. They

could provide greater structural flexibility and simplify programming and process control. These ideas had already been implemented in the Burroughs and El'brus machines.

**Modularity and reconfigurability.** To maximize the average performance of the system across a wide spectrum of applications, Marchuk and Kotov held that a rigid, uniform architecture could not be used. Such systems, while achieving high performance on certain kinds of problems, ran slowly when there was a mismatch between the nature of the problem and the architecture of the machine. To overcome this difficulty, Marchuk and Kotov would rely on the principles of modularity and reconfigurability. Rather than rely on one type of specialized system, a full MARS configuration would include different types of subsystems, each oriented towards a different class of problem. Furthermore, it should be possible to modify the structure of systems to match the characteristics of a given application. The challenge, of course, was not to lose performance overall as a result of this flexibility. The key was the integration of the core MARS principles in a manner that was without internal contradictions. They felt that these conventions should be reflected not only in the hardware and architecture, but in the software as well.

The belief was that the conventions not only were compatible, but could all be integrated into a coherent whole, multiplying the effectiveness of each [Koto91, 44]. Three basic principles which would facilitate this were [Marc78b, 6-7]: (1) a unified set of rules and means of composing systems (and programs for them) from modules of various levels; (2) evolution of the functional capabilities of the system and its ability to adapt through hardware constructs, virtualization, and specialized modules; and (3) a unified principle of asynchronicity for organizing the base system, and the computational processes and programs.

MARS represents a series of experiments seeking to further the knowledge of concurrency and the assumptions of the design philosophy through the construction of a number

of artifacts exhibiting different designs and application paradigms. In the MARS projects, in keeping with the basic philosophies, a multiprocessor is viewed as a “(statically) reconfigurable structure of loosely and/or tightly coupled based processing modules” [Koto91, 37]. Two key projects were the MARS-M and the MARS-T. In addition to the overarching philosophies just mentioned, each of these projects exhibited a number of more specific design goals.

Kotov, G. I. Marchuk, and Yu. L. Vishnevskiy decided in 1980 to start an implementation of the MARS ideas. They felt the basic ideas had been sufficiently worked out that they could begin designing a concrete architecture. Having persuaded V. S. Burtsev to agree to support the project by making ITMVT facilities and tools available, they felt such a project was feasible. Largely thanks to Marchuk’s efforts the GKNT agreed to fund initial design work which was carried out between 1980 and 1985.

## **6.4 The Pre-START Years (1983-1985)**

### **6.4.1 Formation of START**

The START program and the research carried out within it were born out of the confluence of three major streams of activity: research in concurrent architectures and artificial intelligence in progress in the Soviet Union during the early 1980s, the Japanese Fifth Generation Project, and changes in Soviet legislation which made it possible to organize research and development in new ways. In the years following the announcement of the Japanese Fifth Generation Project in 1981, Soviet researchers, particularly in the area of artificial intelligence, began discussing ways of incorporating their efforts into a Soviet counterpart to the Japanese program. A new form of organization called a “Temporary Scientific-Technical Collective” (VNTK), legalized in 1983 while such discus-

sions were taking place, became the vehicle in which the “Soviet Fifth Generation Project” was carried out.

The Japanese Fifth Generation Project placed artificial intelligence research at the core of a program oriented towards the development of a new generation of computers. Anticipating that information processing systems of the future would play crucial roles in increasing the productivity of office workers, cultivating information as a national resource comparable to food and energy, assisting in saving energy and resources, and coping with an aging society, the Japanese initiated a program to develop computers with the following characteristics: increased intelligence and ease of use to better assist man; the ability to process information conversationally using everyday language; the ability to store knowledge and carry out learning, association, and inference functions; etc.

[Moto82]. The Japanese announced their plans in October, 1981, at the International Conference on Fifth Generation Computing Systems in Tokyo. The Institute for New Generation Computer Technology (ICOT), formed on April 14, 1982, was at the core of the project [Feig84, 10]. Only about forty researchers worked at ICOT itself, but over 150 others in Japanese industry worked under their direction [Feig84, 27]. The anticipated total budget for the ten-year project was \$850 million [Feig84, 115].

Some of the earliest discussions about the creation of a comparable Soviet program took place between Aleksandr S. Narin’yani, Viktor M. Bryabrin and Enn Kh. Tyugu early in 1983. Each of these individuals had been engaged in AI research for many years, and felt that it was necessary to “do something,” to organize a higher-level, focused program of research in AI. Narin’yani, Bryabrin, and Tyugu worked together on preliminary drafts for the formation of a new organization, but drew Vadim Ye. Kotov into the discussions early on. Unlike the other three, Kotov had strong ties to Guriy I. Marchuk, who at this time was chairman of the GKNT.

Kotov proposed emphasizing the architectural aspects of the program and using artificial intelligence problems as benchmarks. He approached Marchuk who, after some discussion, agreed that the proposal was worth trying to push through policy-making channels [Afan87]. Kotov saw an opportunity to acquire additional support for MARS and worked this project into the proposal. Narin'yani, Bryabrin, Tyugu and Kotov spent much of the summer of 1983 in the GKNT headquarters in Moscow drafting a more formal proposal for higher-level authorities.

Within the Soviet science and technology community and in bodies such as the Coordinating Committee on Computer Technology<sup>2</sup> of which Marchuk was chairman there was considerable discussion about possible responses to the creation of national computing projects in Japan and Western Europe. While some proposed large-scale, long-term projects with massive government funding, others, including the founders of START, proposed more modest approaches for a number of reasons. Large-scale computing programs in the past, such as that to develop a national system of automated management systems (ASU) had not lived up to the initial promises [Mche85], and many had very legitimate doubts that a large-scale program would succeed this time. During the early 1980s the Soviet economy was experiencing steady decline and policy makers were less eager to commit to large, long-term investments in computing. Furthermore, Soviet industry was not capable of producing the advanced components, packaging, and cooling systems needed to compete in the race to develop the next generation of supercomputers [Koto91b, 30].

The founders of START wanted to create an efficient, controllable program that would build on what strengths the Soviet computing community had. In contrast, a large-

---

<sup>2</sup>Created in 1979, this body of the USSR Academy of Sciences was tasked with coordinating funding of large-scale academic projects and establishing and supervising links with industrial ministries [Petr79].

scale project would have to rely on funding from the military, a traditional sponsor of computer-science research. The military's involvement, however, would mean greatly increased bureaucratic overhead and reduced freedom in managing the project. The involvement of many institutions would also greatly increase the management difficulties. This was to be avoided, if possible.

Marchuk did not fully share this aversion to large-scale programs, however, and made some effort to expand the project. He invited the directors of other computing institutes such as V. S. Burtsev from ITMVT and V. V. Przhiyalkovskiy from the Scientific-Research Center for Electronic Computer Technology (NITsEVT) together with other officials, including some from the Military-Industrial Commission, to participate in a committee to develop a formal proposal. The final product consisted of an industrial portion and a fundamental research component. In an effort to return to the notion of a smaller, more flexible, less encumbered approach, Kotov, Narin'yani, Tyugu, and Bryabrin decided to continue pushing the smaller, fundamental research portion, leaving the others to promote the industrial component, if they desired. Apparently the industrial component never became viable, even though it officially existed.

The laboratories of the four founders—the Computer Center of the Siberian Department of the USSR Academy of Sciences in Novosibirsk, the Institute of Cybernetics in Tallinn (IK AN ESSR), and the Computer Center of the USSR Academy of Sciences Moscow (VTs AN SSSR)—formed the core of the project. A Novosibirsk laboratory of the Impul's Scientific Research Association in Severodonetsk, Ukraine was also incorporated as the sole formal industrial participant. The Novosibirsk laboratory was selected largely because of geographic proximity and existing working relationships. Although connections between Impul's and the other START organizations were rather weak, some industrial participation was helpful because of the hardware and construction expertise

that it could provide. The subsidiary of NPO Impul's and the prototype development and design bureaus associated directly with the Computer Center would not be able to provide all the industrial capability necessary to build MARS hardware prototypes, however, and relationships with industry factories outside of Novosibirsk would have a strong impact on the projects' progress.

As word about the new project spread through the Soviet computing community others sought to join but were turned down. The founders felt that the participation of many organizations would be too difficult to control. By keeping the proposal modest, the founders could exist solely on support obtainable by Marchuk through the GKNT and the Academy of Sciences, and not involve the military.

In August 1983, the USSR Council of Ministers passed a resolution "On Measures for Accelerating Scientific-Technical Progress" instructing various ministries to establish Temporary Scientific-Technical Collectives (VNTK) and defined in detail the characteristics of such organizations [Ntr85]. The original resolution was rather vague, but stressed that research and development teams, created for limited time periods and pooling the expertise of individuals in multiple branches of science and industry, should be able to conduct R&D in a more efficient manner than had been the case earlier. Such a resolution was made at precisely the time when START organizers were considering such questions, and they adopted the VNTK label and actually helped define what such organizations should be like. In later years they occasionally consulted for other groups trying to organize similar teams.

A number of measures were taken to stimulate productivity. First, the basic nature of an inter-branch, temporary collective served to bring together young, qualified researchers with complementary expertise into a single, high-intensity, high-visibility program. Second, although imposed by law, the three-year duration of the program forced re-

searchers to produce results within a compressed timeframe. Third, a bonus averaging 3,000 rubles was promised each participant if the project goals were achieved within the three years [Koto91b, 31; Manu88].

Although a major goal was the creation of an efficient, controllable, moderate-scale program, the effort to create a new type of structure actually added complexity in the initial phase. The creation of a research organization, particularly one with a new structure required the approval of several ministries at all levels. In spite of efforts to minimize the number of participants, the involvement of multiple ministries was unavoidable. The primary work was being carried out within the Academy of Sciences. Equipment was needed from the Ministry of Instrument-Building, Means of Automation and Control Systems (Minpribor). Financing was to come from the GKNT and the Academy of Sciences. Because START was a new organizational form, the Ministry of Justice, the Ministry of Labor and Wages, Trade Unions, and the Military-Industrial Commission all had to approve the proposal. In each ministry, one had to collect signatures from the lowest levels to the highest, even from ministers themselves. When objections were raised, they had to be addressed and approved by all the other organizations. The final document contained 121 signatures.

The turmoil in leadership of the country further delayed the process. The first draft proposals were written while Yuriy Andropov was General Secretary. Little progress was made while Chernenko was in office. Only in January, 1985 did the USSR State Committee on Labor and Wages and the All-Union Central Council of Trade Unions resolve the practical matters of wages for members of VNTK in the resolution “On Procedures for the Payment of Wages and Bonuses for Workers of Temporary Teams” [Ntr85]. START was finally organized on April 1, 1985.

As the START proposal was being pushed through the Soviet bureaucratic labyrinth, organizers found it very useful to draw strong analogies to the Japanese Fifth Generation project. START was not on the same scale as the Japanese effort in terms of duration or financing, but casting START as a response to the Japanese program both validated START goals in the eyes of policy-makers, and provided a national program which leaders could use to try to demonstrate to themselves and the world that the Soviet Union was a serious participant in leading computer science research at a time when a number of national and international programs such as Alvy and ESPRIT were being created. Phrases such as “The purpose of START is to perfect and test fifth generation computers...” were common in press reports about the new organization [Zakh85; Ntr85; Favo86; Afan87; Koto87]. Not surprisingly, once in existence, START continued to be promoted using *perestroika* terminology adopted by Gorbachev. The project was a means of promoting the “acceleration of scientific and technical progress and intensification of the national economy” [Koto87; Gorb87].

#### 6.4.2 Nature of the Research Plan

Although the founders desired to keep the scale of the project manageable, they felt it necessary to pursue advances on many fronts. As Kotov said, “The transition to the fifth generation is impossible without radical improvement of all the things that make up computer technology. Everything must be updated: the component base, means of communication, software, and primarily machine architecture” [Ntr85]. Not only are new components and subsystems required, it was argued, but progress in these areas had to be integrated to a much greater extent than was the case in the traditional ministerial, command-economic structure with its compartmentalization of development [Koto87]. This argument was used to justify the creation of a VNTK form of management in which researchers in different administrative entities could much more easily work with each

other. Furthermore, while critical of the results of the Soviet computer industry, Kotov consistently argued that the Soviet computing community needed to conduct at least *some* original research in all these fields to preserve its intellectual capital [Koto87]. This led to considerable breadth in START-related research, although individual projects did not depend strongly on each other.

The founders initially had wanted to tie all the threads of their existing research into a single, tightly-coupled project. Early descriptions of START speak of the development of the MARS computer, which would integrate a wide range of functionality, from AI to database management, to networks, to high-speed scientific computation. There were discussions about establishing a common representation for all the structures used in the AI components. When all was said and done, however, the degree of actual overlap between projects was slight. As explained below, the difficulty in bringing about this integration was greater than had been anticipated, and the final results of START would be better characterized as an aggregation of individual projects than true joint efforts.

The component research projects were, for the most part, extensions of research which had been in progress for several years and for which some preliminary results had been achieved. Tyugu had begun development on the PRIZ program synthesis system during the early 1970s [Tyug70]; Kotov and Narin'yani had published on asynchronous concurrency in 1966 [Koto66], and Kotov and Marchuk had first published MARS ideas in 1978; Narin'yani had conducted research in various areas of artificial intelligence, natural language processing in particular, during the 1970s [Jako85; Nari85]. It could be argued that this approach had to be taken to produce the necessary results within the three-year limit imposed by the VNTK legislation. As we shall see, the MARS-M could hardly have been built from scratch in three years.

## 6.5 The START Years (1985-1988)

### 6.5.1 MARS Research

START-related research was broad-based. At the Computer Center of the SO AN SSSR, four laboratories were involved. Under Yuriy L. Vishnevskiy, the parallel systems laboratory worked on creating the MARS-M computer. Vadim Ye. Kotov's laboratory worked on the development of two parallel languages, BARS and Pol'yar. Aleksandr G. Marchuk's laboratory focused primarily on developing the 32-bit Kronos microprocessor and the MARS-T parallel system incorporating the Kronos. Marchuk's laboratory also developed a computer-aided design system for designing VLSI chips, the Kronos in particular. Aleksandr S. Narin'yan was the head of a laboratory involved in artificial intelligence research. In Moscow at the Computing Center of the USSR Academy of Sciences, Yu. G. Yevtushenko and Viktor M. Bryabrin developed systems and applications software, in large part for personal computers. At the Institute of Cybernetics in Tallinn, researchers in the Systems Software Department under Enn Kh. Tyugu developed object-oriented software development systems, program synthesis systems, and an object-oriented workstation based on the Kronos processor. In this section we discuss the MARS hardware projects, MARS-M and MARS-T.

#### 6.5.1.1 MARS-M

The 1978 preprints by Kotov and Marchuk presented the high-level principles which defined the MARS architecture. In particular, they discussed dividing a computing process into four different kinds: control, memory access, execution, and communication. The MARS-M was the primary effort to implement these principles. The original name, "Mini-MARS," referred to the fact that this machine was conceived to be a small portion, the numerical processor, of a larger MARS configuration [Koto86b, 280-281]. Besides

being an effort to implement the principles outlined in the Conception, the MARS-M represents a practical effort to explore issues of parallelism at multiple levels within a single architecture. This involved the development of architectural support for parallelism at each level and language facilities for expressing parallelism at the corresponding levels.

The MARS-M incorporated the key principles outlined in section 6.3 (with the exception of hardware tags, which were deemed unnecessary as development progressed), but on a smaller scale than outlined by the Conception. In particular, while the Conception called for execution systems integrating a variety of special-purpose processors (general-purpose arithmetic, vector, associative, symbolic, etc.), a memory system incorporating addressable, associative, and orthogonal memory modules, etc., the MARS-M was limited to a single type of memory, and an execution subsystem incorporating control, execution, and addressing processors.

Kotov and Marchuk communicated these ideas to Yuri Vishnevskiy during the late 1970s and 1980 as they discussed the possibilities of creating a machine based on these principles. Kotov and Marchuk played only a minor role in deciding *how* to implement such an architecture, however.

Following a brief overview of the architecture and construction of the MARS-M prototype, we present a chronology of development, and analyze the factors influencing the development of architectural ideas, and their realization. A more extensive description of the MARS-M can be found in [Doro92].

**Architecture.** The MARS-M is a shared-memory heterogeneous multiprocessor having a control processor, a central processing unit, a peripheral subsystem, a memory management unit, and multiported main memory. The control processor consists of eight (virtual) systems processors, and the central processing unit consists of a control subsystem,

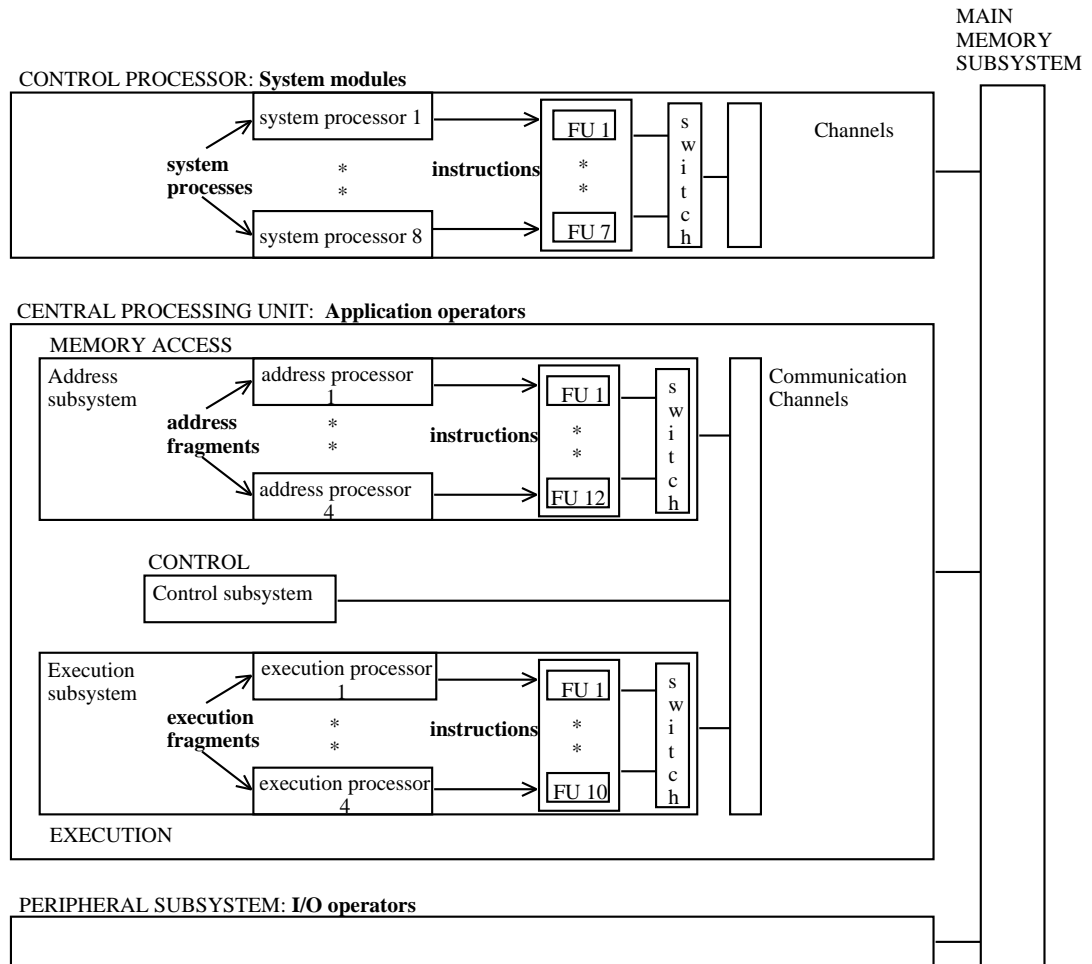


Figure 6-1 MARS-M Logical Structure

four address and four execution processors. These elements constitute the fixed “skeleton.” Figure 6-1 shows the logical structure of the MARS-M.

The MARS-M has the following principal characteristics:

- hierarchical organization of data processing;
- multiprocessing;
- multi-pipelined organization of each processor;

Level	Model of Computation	Language Objects	Execution of the Objects
System	Virtual heterogeneous multiprocessor	Modules and I/O operators	by the control processor and the peripheral subsystem
Application	Decoupled architecture	Application operators	by the central processing unit
Functional	VLIW architecture	Address and execution fragments	by functional units of the address and execution processors

Table 6-1 MARS-M Levels and Their Features

- asynchronous interaction of processors with the help of uni-directional buffer data and logical value transmission channels;
- powerful descriptor mechanism for working with compound data objects like vectors, arrays, etc.;
- adaptation of the MARS-M architecture for solving a specific class of problems by means of a selection of commands for the formulation of representations and processing of compound data objects.

The MARS-M has three architectural levels: systems, application, and functional. Each level has a corresponding computation model, language objects processed at that level, and hardware elements to execute those objects. Table 6-1 shows the levels, together with the computation models and associated language objects. The system and application levels were proposed in the MARS Conception. The third, functional level, represents a lower level needed to support the execution of application operators in the central processing unit.

*System level.* Computation within modules is defined by so-called control expressions. These contain instructions which call modules, operators, other control expres-

sions, etc. The execution of control expressions is called a system thread. A new system thread can be generated both explicitly by a parallel call operation invoking a control expression and implicitly with the help of a token control mechanism.

In general, this control mechanism is similar to macro (procedure-level) data flow mechanisms. Its specifics lie in the objects whose execution is to be dynamically scheduled: the MARS-M's modules, operators, control expressions, and system operations.

*Application level.* Like the central processing unit which supports it, the computation model for the application operators is based on a decoupled multiprocessor architecture. Decoupled architectures have two main elements [Smit86]. They have two separate instruction sets and concurrent processing of at least two instruction streams, one for accessing memory and one for performing function execution; and communication between the memory access and the execution processes takes place via architectural queues. Operators in the MARS-M are built from three types of operations: control, address, and execution. The latter two are called address and execution fragments. They are functions for performing complex computation and memory access operations that are typical for an application field. Different applications can have different sets of "elementary" address and execution operations. The execution of an address fragment by an address processor, an execution fragment by an execution processor, and control instructions by the control subsystem are called address, execution and control threads, respectively. Up to four address, four execution and one control threads can run simultaneously at this level. These executing threads communicate via queues of different types.

There are two asynchronous mechanisms to schedule parallel processing within an operator: data flow and control token. The data flow mechanism, hidden from the programmer, is built into hardware to synchronize communication between fragments through any queues. The control token mechanism is used by the programmer to impose

additional synchronization control on decoupled multi-thread processing. Here, thread exchange is enforced via special control messages sent through branch queues.

*Functional level.* At the functional level, the computational model is very-long-instruction-word execution of address and execution fragments. The control subsystem splits an instruction stream generated from application operators (from the *application level*) so some of the instructions are passed to free address and execution processors, while others are executed by the control subsystem.

Each instruction sent to the processors contains a fragment call operation. Fragments are independent program blocks which can consist of multiple basic blocks and a control part containing branches, loop-counters, literals, and other operations. A basic block is a sequence of code that can only be entered at the top and exited at the bottom. Conditional branches and conditional branch targets typically delineate basic blocks.

Four address and four execution processors execute up to eight fragments in parallel. Address fragments compute addresses of elements within structured objects such as arrays or vectors which either have to be fetched from memory into queues, or have to be stored into memory from queues. Execution fragments perform their operations on data being fetched from memory or computed and passed from address fragments via queues external to the subsystem and/or computed and passed by other fragments via the subsystem's internal queues. Instructions are issued strictly in program order but can begin and complete their execution out of order. The reason for this out-of-order execution is the asynchronous communication between processors and memory via channels. A data flow mechanism is used to initiate or suspend the execution of a fragment, depending on the availability of operands in its channel queues.

Both address and execution processors have a horizontal architecture with multiple pipelined functional units. The execution and address subsystems both consist of four

processors sharing a set of functional units. A single very-long-instruction-word contains multiple operations and in each cycle operations can be issued to all of the functional units simultaneously. A unique feature of the MARS-M is that it uses a combination of static and dynamic scheduling for forming and issuing very-long-instruction-words. A so-called fragment compiler schedules individual fragments statically, forming very-long-instruction-words to be executed on individual address or execution processors. These instruction words are combined at run-time by a fragment dispatcher to run concurrently on the functional units.

Since the MARS-M computer is oriented towards scientific computation, the effective processing of arrays and vectors was a primary goal. A number of address and execution fragments implementing typical array/vector operations were developed. In many cases, array-subscript expressions can be rather complex and contain conditional branches. To provide continuous address generation for such expressions, special array address generators were incorporated into the address subsystem, making it possible to generate an address in each clock cycle.

The MARS-M could be adapted for specific classes of application programs by creating specialized problem-oriented fragments. This library-oriented approach was similar in concept to that used in many attached array processors which provided a library of routines accessible by users. The library-oriented approach for fragments made it possible to load the code for all fragments into the distributed instruction memory before a program was initiated and fetch it locally and independently by multiple sequencers at run time. The compiler specified a timetable of all instruction streams to be issued by the control modules in executing any fragment.

**MARS-M Performance.** Investigations of MARS-M performance showed some weakness in the pure queue-based architecture of the address and execution subsystems

in scalar processing. This was due to a lack of general-purpose registers for storing local data of the operators being executed within these subsystems, resulting in the use of the complex dynamic resource allocation mechanism to perform simple scalar operations. To resolve this problem, the local data memory of each subsystem was divided into two parts: one for storing the constants of all the fragments being loaded into the subsystems, and the other for storing operators' local data. After redesigning some mechanisms, it became possible for fragments to take/pass their input/output parameters through the general-purpose part of the local data memories. This implementation significantly improved MARS-M scalar processing performance [Vish84].

The aggregate peak performance of the MARS-M prototype was 18.5 Mflops or 240.5 MIPS, which is the sum of the following components [Doro92]:

- execution subsystem: 18.5 Mflops or 92.5 MIPS
- address subsystem: 111.0 MIPS
- control subsystem: 9.25 MIPS
- control processor: 27.75 MIPS

**MARS-M Chronology.** Kotov, G. I. Marchuk, and Vishnevskiy made the decision to start implementing some of the MARS ideas in 1980. They felt that the basic ideas had been sufficiently worked out that they could begin designing a concrete architecture. They had gotten enough support from V. S. Burtsev that they felt that such a project was feasible. The GKNT agreed to help finance the project, but demanded a target performance rate of 50 Mflops. Kotov felt this was rather unrealistic and proposed 10 Mflops. As a compromise, a figure of 20 Mflops was agreed upon. The first MARS-M design was developed during 1980-1981. At this time certain technical parameters were established in conjunction with ITMVT. For example, the system would use ECL chips with 3

nsec/gate delays, water cooling, would have a maximum of 100 ICs/per board, would fit into standard El'brus cabinets, etc.

In 1982 a second version of the design was initiated. Here the number of functional units and the clock period were established. At the same time, the logic design of subsystems and the development of a fragment translator and simulator was begun. In 1983, further modifications were made to the instruction sets of the arithmetic and address subsystems, and the clock period had to be increased to 92 nsec. Design of the memory subsystem was completed in 1984. In 1985, the year START was formally created, design work on the MARS-M was completed, and prototype assembly began at the VEM Factory in Penza. In this year the clock period had to be increased to 100 nsec. In 1986, when prototype development at Penza was delayed, the Novosibirsk team developed the concept of a distributed operating system kernel, designed and built a simulation model of the control processor, and began design of the parallel structure machine language (KOKOS). In 1987, further software design was carried out, a debugging subsystem was created, and the clock period once again had to be increased, to 108 nsec. The prototype was finally completed in 1988 and installed at the Novosibirsk Subsidiary of ITMVT. For the next two years, until funding was terminated, it underwent testing [Doro92b, 2].

**Construction, and influences on the design.** The construction of the MARS-M was heavily influenced by the participation of ITMVT. Through ITMVT, the MARS-M developers gained access to components, subsystems, and CAD systems. ITMVT provided technical advice, logic design and a board layout system. The institute also produced production documentation for the factory. While the development of a MARS-M prototype would not have been possible without such assistance, these elements constrained development. Only by conforming to the technology available at ITMVT could the machine be built, however.

When the decision was made to use El'brus-related parts and systems to build the MARS-M, the intention was to integrate the MARS-M into an El'brus configuration as a special-purpose CPU [Golo86]. By not having to develop the entire I/O system, and being able to use many of the materials and facilities available through ITMVT, the MARS-M developers could save considerable time, money, and effort [Doro92b, 1]. Developers could use the El'brus racks, boards, memory, chips, power supplies, cooling system, and the entire El'brus I/O system. The system incorporated Soviet analogs of the Motorola's MECL 10K chips, the IS-100 series with a minimum delay of 2.5 nsec. These medium scale integration ECL chips were used in the El'brus-2 and Soviet high-end mainframes through the mid-1980s, when ECL gate-arrays were introduced [Anto81; Smol83; Loef83; Kuch85; Lomo87].

The MARS-M had to fit into the three racks constituting a single El'brus CPU. It was not possible to add a fourth rack because the racks were designed to fit in a three-spoke configuration, and the water cooling system and power supply were also designed for three racks, not four. The need to build the MARS-M small enough to fit into the racks constrained the implementation in a number of ways [Doro92b, 1]. First, the number of functional units and processors had to be limited. While the Conception called for multiple execution, address and special-purpose subsystems, the implementation was limited to one CPU with one address and one execution subsystem [Doro92]. Second, the designers were forced to reduce the number of functional units. There was room in the prototype for only one floating-point addition unit, and one floating-point multiply [Doro92]. Third, designers had to decrease the word-length to 48-bits, rather than use the standard 64. Fourth, the number of memory address channels and memory ports had to be reduced from eight each to six.

Main memory and the memory management unit occupied one of the three water-cooled racks. The 2M-word memory used 16-Kbit ECL DRAMs.

The designers also were forced to accommodate the El'brus system by choosing a clock period which could be easily synchronized with that of the El'brus-2. Their original choice, 88 nsec, was, in the early 1980s, exactly twice the planned clock period of the El'brus-2. Timing issues were never fully resolved, however. As they constructed the MARS-M, developers found that they actually needed a clock period of 108 nsec, more than twice the clock period of the El'brus-2, even though the latter ended up with a clock period of 47 nsec. Second, the host El'brus available to them in Novosibirsk was an El'brus-1, housed at the Novosibirsk Subsidiary of ITMVT located in an adjacent building. Not only did this machine have a different clock period, but the connection to the El'brus exceeded the six meters maximum called for in the design specifications. This too threw off the timing.

During the project developers realized that they would need to use conventional sequential languages. They planned to develop FORTRAN and C compilers at a later stage and move from library-oriented to conventional high-level language programming. These compilers would have to build a data flow dependency graph and find independent fragments of a program which could be executed in parallel by address and execution processors communicating with each other. FORTRAN and C compilers were not implemented before the project ended, however.

**Relationship to Western developments.** The MARS-M was a unique computer which combined an unusually large number of different architectural concepts. As we have seen, at the conceptual level, Kotov and Marchuk began with a survey of Western machines and selectively drew some basic principals from Western and Soviet research. At the implementation level, it is difficult to trace the origins of many MARS-M ideas.

The final design contained a number of features which are similar to some Western efforts, but in many cases the Soviet work pre-dated the Western or at least represented an instance of parallel development.

Perhaps the strongest Western influence on implementation were the horizontal architecture ideas which had been incorporated into the AP-120B attached array processor by Floating-Point Systems, Inc. This machine was first delivered in 1976 [Hock88, 206]. It used instructions which were 64-bits wide, and each instruction controlled the operations of all units in the machine [Hock88, 212]. This arrangement was called 'horizontal microcode.' One instruction could be issued per clock period, but since each instruction controlled multiple operations, the aggregate performance was much higher than for a machine with a comparable clock time, but only one functional unit. This approach inspired the use of very-long-instruction-words at the MARS-M functional level, even though actual implementation was unique.

A number of Western projects have characteristics which resemble portions of the MARS-M. The MARS-M library-oriented approach for fragments made it possible to load the code for all fragments into the distributed instruction memory before a program was initiated and fetch it locally and independently using multiple sequencers at run time. The compiler specified a timetable of all instruction streams to be issued by the control modules in executing any fragment. A similar extended VLIW approach, called XIMD, incorporating multiple sequencers and homogeneous functional units was proposed in [Wolf91].

Ideas similar to the MARS-M virtual multiprocessing scheme were implemented in the HEP-computer [Smit78]. Both systems use dynamic sharing of common hardware between multiple processes, performing processor switching when memory access operations are issued.

The ZS-1 and the PIPE computers are examples of a recently developed computers with decoupled architecture [Smit89; Farr91]. The MARS-M differs from these and earlier efforts in its use of multiprocessing to perform both memory access and execution tasks, and dynamic hardware allocation of both address/execution processors and communication queues.

The idea of virtual multiprocessing is not new. For example, the CDC 6600 used virtual multiprocessing for its peripheral processors in 1964. The approaches of the MARS-M and the CDC 6600 differ in key respects, however. Only two of the CDC 6600 virtual processors executed operating system functions; the others were used for communication with input/output devices. Also, the CDC 6600 peripheral processors were switched in equal time periods, in contrast to those of the MARS-M, which are switched dynamically.

Most implementation ideas originated with the Russians, however. Vishnevskiy liked to work in isolation of other work being done in the world, preferring to think up his own ideas. While working within the high-level framework established by Marchuk and Kotov, he made decisions about implementation not from the perspective of the MARS ideology, but from a more pragmatic viewpoint. He knew about the horizontal architecture of the FPS, and adopted this approach, but in general had limited access to foreign developments. He reportedly does not read English very well, so it was difficult to absorb Western ideas. In [Vish85, 79] he attributed some influence to ideas about parallel processing, pipeline processing, and distributed memory conceived by S. A. Lebedev, the Father of Soviet computing.

Dorozhevets, the principal designer of the control processor, was considerably better informed about Western developments. The MARS-M team did not appropriate Western ideas directly, but occasionally a Western development would generate a new idea, or reinforce some design decision already made. The ideas implemented were ultimately the

outcome of a complex interaction between foreign ideas, indigenous ideas, and untraceable trains of thought. Even in retrospect, it is difficult for designers themselves to trace the origins of ideas. Dorozhevets comments [Doro92c, 7]:

Maybe I had known about distributed operating systems, but I'm not sure now. It's difficult to remember why you proposed this.... I've read so many papers at that time that it was useful for me to see links, the similarities between ours [and foreign developments] and this coincidence sometimes resulted in new ideas how to take the next step. It was a complex of reasons. I knew about horizontal architectures. I knew about all the supercomputers in the USA at that time. But our information is sometimes very basic, not very deep. Sometimes it was not clear. So maybe, to compare our approach...it was not a direct compilation of ideas. That much is clear.

Although it never became a viable machine, the MARS-M was a useful research vehicle. Perhaps overly complex, it nevertheless demonstrated the possibility of integrating multiple architectural approaches such as data flow, decoupled heterogeneous processors, hierarchical systems, and VLIW scheduling. It also explored possibilities for combining static and dynamic scheduling in a VLIW implementation.

#### 6.5.1.2 MARS-T

The MARS-T was intended to be a testbed for experimentation with a variety of concurrent structures, communications methods, and architecture-algorithm relationships. It represented an integration of the MARS philosophy including earlier research on asynchronous processes by Kotov, Narin'yani, and others, and two bodies of Western research: transputer architectures and the associated Communicating Sequential Processes (CSP) computational model of C.A.R. Hoare, and Niklaus Wirth's Modula-2 research and the Lilith Modula-2 processor.

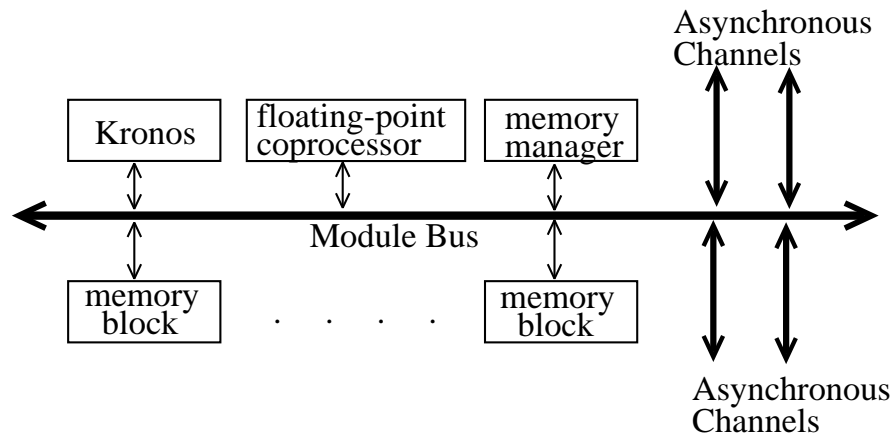


Figure 6-2 MARS-T Basic Module  
Source: [Koto91, 38]

The MARS-T contained one or more processing modules, shown in Figure 6-2, each consisting of at least a processor (called the Kronos), a memory block, a memory manager, a floating-point coprocessor, and asynchronous channels. In keeping with the philosophy of modularity and expandability, the MARS-T configuration could be increased either by adding additional processors, additional memory, a memory manager, or specialized processors such as an arithmetic co-processor to any module; or, by adding additional modules. Each module was considered a node, and nodes could be grouped in a variety of configurations such as a regular network or a hierarchy of nodal clusters. By varying the intra- and inter-nodal configurations, the system as a whole could be tailored to specific applications. One example, in this case the MARS-T prototype model, is shown in Figure 6-3.

MARS-T designers chose to base their system on C.A.R. Hoare's CSP model primarily because it is simple [Koto91, 37]. Hoare's contribution was to suggest that input and output are basic primitives of programming, that parallel composition of communicating

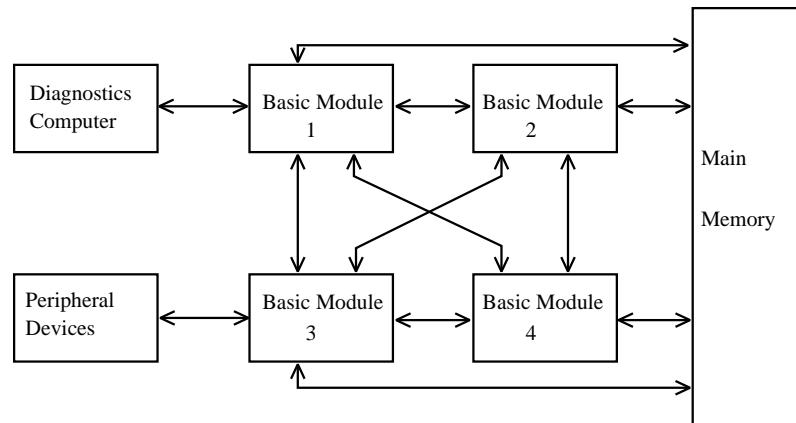


Figure 6-3 MARS-T Logical Structure

sequential processes is a fundamental structuring method, and that such communication can be expressed simply and usefully. CSP is based on three components: (1) Dijkstra's guarded commands which are used as a sequential control structure and are the sole means of introducing and controlling nondeterminism; (2) a parallel command to specify concurrent execution of constituent sequential processes; and (3) simple forms of input and output commands which are used for communication between concurrent processes [Hoar78]. CSP is the computational model underlying the Transputer and its native programming language, OCCAM.

The MARS-T computational model differs from CSP principally in two respects: the channels are asynchronous, being represented as first-in-first-out (FIFO) buffers, and processes are created dynamically with recursive creation allowed [Koto86, 11; Koto91, 37]. Processes can access a queue whenever it is ready to send or receive data. More details on the physical implementation of such queues are given in [Kuzn86, 16-17; Koto91, 38]. Modifications introduced in the implementation to improve efficiency in-

cluded: (1) a prioritization of processes accessing channels; (2) a time-limit on how long a (physical) processor must wait either to take data off of a channel or to put it on; and (3) a protection scheme to prevent unauthorized use of channels. The prioritization of processors accessing a channel does not change the FIFO prioritization of data already *in* a channel [Koto91, 37].

Hoare considered the alternative of making the basic communication process asynchronous. He deliberately rejected this alternative for two reasons: “(1) It is less realistic to implement in multiple disjoint processors, and (2) when buffering is required on a particular channel, it can readily be specified using the given primitives” [Hoar78]. The Novosibirsk designers had different reasoning. The asynchronous channels were more faithful to the overall MARS philosophy, and the data-to-process synchronization being implemented was viewed as simpler and more efficient than the process-to-process synchronization required by the CSP model [Kuzn86, 19].

Hoare also deliberately disallowed recursion. In explaining the rather static nature of the model, he explains that it was intended to be implementable both by a conventional computer with a single main store, and by a fixed network of processors connected by input/output channels [Hoar78]. Simplicity was a key consideration. Considerations of simplicity certainly played a role in the design of the MARS-T, as evidenced by the selection of CSP, but it was not part of the fundamental design philosophy.

The basic processor within a MARS-T is the Kronos. The Kronos originated as a student’s master’s project. Ye. V. Tarasov, D. N. Kuznetsov, and others developed a one-board processor for the DVK<sup>3</sup> computer, finishing at the end of 1983. Aleksandr G. Marchuk, head of the MARS-T project, was not involved with the project at this time.

---

<sup>3</sup>The DVK family of personal computers has an instruction set based on that of DEC’s PDP series.

In 1983, A. G. Marchuk began efforts to start a multiprocessor project and needed a basic computing element to serve as a processing module. He first considered using the MARS-M control processor, redesigning it on the basis of the Soviet analog of the Am2900 bit-sliced processor, the K1804. This would have required that Dorozhevets leave Vishnevskiy's laboratory to work for Marchuk. Vishnevskiy did not agree, so using the Kronos processor was one of the only options open to Marchuk. At this time there were no other 32-bit processors available in the Soviet Union. Marchuk adopted the Kronos and included the project into START. It grew to become the major START project, consuming more resources than any other.

In developing the Kronos, developers drew many ideas from the Lilith processor developed by Niklaus Wirth at the Swiss Federal Institute of Technology. Like the Lilith, the Kronos is a processor designed for fast and efficient implementation of Modula-2. Three fundamental elements of the Lilith design incorporated into the Kronos were: an instruction set carefully selected to map neatly to Modula-2 constructs; a stack-based architecture which is known to support the implementation of high-level procedural languages rather well [Bulm77]; and an addressing scheme relying, in general, on short addresses.

The Lilith used six instruction formats ranging in length from one byte to three. These were designed so that the most frequently used instructions had the shortest length. The stack-based architecture enabled the Lilith to use relative rather than absolute addressing, greatly conserving the space that would otherwise have been used for addresses [Ohra84]. The Kronos incorporated all these features, except that the Kronos instructions were one-, two-, or four-bytes in length [Kuzn89]. In both processors most of the commands are single-byte.

A number of other low-level design features were shared by the Kronos and the Lilith processors: the use of implicit, immediate, local, global, indirect, and external addressing; an immediate format using a four-bit opcode plus a four-bit constant, a feature previously unique to the Lilith; strict control over the over/underflow of the expression stack by the compiler [Ohra84; Kuzn86]; a limit of 256 local variables within a procedure, although this is a limitation imposed by the compiler rather than the hardware [Kuzn89, 2]; and indirect referencing of the values of structured variables—arrays and records—, so more than 256 variables can be accommodated if they are stored within such structures.

Both processors supported a process-switching command. The Lilith, designed to be a single-user machine, uses this command for co-routines. The Kronos, designed to be incorporated into a multi-process environment, uses the command for multiprocessing [Kuzn89, 3]. Like the Lilith, the Kronos (operating system) supports dynamic binding of separately compiled routines at run-time [Kuzn89, 3].

The Kronos processors were developed entirely using indigenous series production 1802 8n bit-sliced processors, manufactured using Schottkey TTL technology [Schl83; Goly83; Stap88; Kuzn89,3; Kron89]. The Lilith was built using the Am2901 bit-sliced processors, and the Kronoses were built using

The primary differences between the Lilith and the Kronos were that the Kronos was a 32-bit processor, compared with the 16-bit Lilith, and the Kronos incorporated a number of extensions to the instruction set to accommodate the communications necessary in the MARS-T multiprocessor [Koto86]. Besides increasing the size of the operands, the increase in word size permitted an expanded address space. The original Lilith permitted 16-bit addressing; the maximum length of an instruction was three bytes: one for the operand, and two for an address. It had 64 Kbytes of RAM although with a four-bit shift it could access 256K. Later models had double-word addressing, and was built with eight

Mbytes of RAM. The Kronos design permitted instructions of up to four bytes (one word) in one word, providing up to 24 bits of addressing. The amount of main memory included in the processors was, due to space considerations, limited to 512K-2 Mbytes, depending on the size of the RAM chips used [Kuzn89, 3].

The “signal” instruction activating the first of possibly multiple processes waiting for a channel was added to the instruction set to help support inter-process communication. Signals were also used to manage the queue of processes waiting to be executed by a given processor. The addressing scheme of the Kronos was extended to accommodate multiprocessor configurations. The local memories of each processor constituted a global address space accessible by each processor. The global physical memory consisted of a node number and a local memory address. The link controller of each node uses a mapping table with information about the configuration to route data from one link to another [Koto91, 39].

Apart from these primary differences, there were numerous minor differences. For example, in the Lilith the instruction fetch unit performed a prefetch of 64 bits of instructions (four words), while the Kronos permitted only 32 bits of instruction (one word) to be read at a time [Ohra84, 190; Kuzn86, 16]. Designed to handle large high-resolution bit-mapped displays, the Lilith incorporated a broad, 64-bit memory read bandwidth, a more conventional 16-bit write bandwidth, and a special shifting device to accomplish the shifting and masking needed for image movement on the screen [Ohra84, 188-190]. The Kronos did not have these features. The expression stack of the Kronos was seven words deep, while that of the Lilith was sixteen [Kuzn89, 2]. These differences reflect different goals for the machine (the Kronos workstation was not designed to be a graphics workstation) and the nature of the component base which limited the amount of functionality which could be placed on a single board.

Modula-2 was selected in part because it is quite amenable to the basic design philosophy of MARS. It is fundamentally modular, allowing programs to be partitioned into units with relatively well defined interfaces. These interfaces, enabling the specification of an object to be separated from its implementation, promoted information hiding (the control of the visibility of objects to other modules), and separate compilation of modules [Wirt84].

The extensibility of MARS-T software was also facilitated by the modularity of Modula-2. Information hiding allowed software building blocks to be constructed, compiled, and executed in a rather independent manner. The module specifications shielded the outside world from the internal details, providing a concise interface between modules. When modules were created and compiled, they could be linked dynamically at runtime. This obviated the need to maintain multiple copies of a module, reducing the size of the executable code, and speeding up the editing-compiling-execution cycle.

Multiple versions of the Kronos were implemented. The first processor was the Kronos 2.2 [Kuzn89, 3]. Designed for installation into the Soviet-made DVK microcomputers, this processor used Digital Equipment Corporation's Q-bus. It used 32-bit operands, but since the bus was 16-bit, two bus accesses were needed to fetch a word of data from memory. To enable the entire processor to fit on a single board, the ALU was only 20-bit. At least two clock cycles were therefore required for each data arithmetic operation. The processor included up to four Mbytes of directly-addressable memory and ran at a frequency of 4 MHz.

The Kronos 2.5 was the first full 32-bit processor, using a 32-bit ALU and interfacing to the Multibus-1 32-bit bus from Intel [Kuzn86, 15; Kuzn89, 3]. Completed around 1985, this two-board processor was installed in the Labtam computer, a 32-bit machine manufactured in Australia which had been sold to numerous Soviet research institutes, in-

cluding the Institute of Cybernetics in Tallinn, and the VTs SO AN SSSR. The Labtams were being used at the Institute of Cybernetics for software development, and the incorporation of the Kronos into a Labtam system in place of the native processor was one means by which the Tallinn group could contribute software to the Kronos.

The Kronos 2.5 was used in the MARS-T prototypes. Configurations with four processors were built, but plans called for eight processors [Kuzn86, 15; Koto91, 39]. In a further departure from the Transputer architecture, the MARS-T configuration included, besides 512K - 2 MBytes of main memory local to each processor, a common memory store using the four-port memory of the PS-3000 multiprocessor developed by NPO Impul's and described in section 7.5.1. The design of a full-function MARS-T system would also incorporate a diagnostics/console computer, a network interface, access to secondary storage, and possibly specialized processors for scientific computation such as the ES-2706 array processor (described in section 7.12.5) or perhaps even the MARS-M. Full configurations were never developed, however.

Early design specifications proposed a clock period of 200 ns (5 MHz) which, given an average 0.75 period instruction access time for a total instruction execution time of 350 nsec, would give an expected performance of 2.86 MIPS [Kuzn86, 16]. Unlike the Kronos 2.2 CPU, the Kronos 2.5 CPU consisted of two boards, however. To accommodate the inter-board propagation times, the clock period had to be increased to 330 nsec (3 MHz), reducing the expected performance to 1.5 MIPS [Kuzn89, 3].

The Kronos 2.6 was a version designed as the engine for a single-processor workstation. The architecture and chips used to implement it were the same as for the Kronos 2.5. The principal differences were that the Kronos 2.6 used a native bus rather than Multibus-1, and was constructed using boards conforming to the European E2 standard

[Kuzn89, 3]. Like the Kronos 2.5, the Kronos 2.6 had a clock frequency of 3 MHz and a peak performance of 1.5 MIPS [Kuzn89, 3; Kron89].

The Kronos 2.6WS was a workstation based on the Kronos 2.6. Another workstation built on the basis of the Kronos was the PIRS object-oriented workstation developed at the Institute of Cybernetics in Tallinn [Kya86; Koto91, 39]. The PIRS workstation was designed to be an object-oriented system, designed to support the NUT (New UTopist) software development environment also developed at the IK AN ESSR. It was distinguished from other workstations, both Kronos-based and otherwise, by the design of a specialized object server which was linked to the Kronos processor and memory via a systems bus.

The decision to build the PIRS dual-processor system was based on at least two considerations. First, it represented one of the few examples of an implementation of the MARS ideology of incorporating special-purpose processors into a single, multi-purpose system. Second, the developers wished to design a system which would, on the one hand, be able to take advantage of the rapid progress being made in conventional, sequential processor technology, and on the other, provide high performance in a specialized problem domain, in this case the management of objects. The desired flexibility was made possible by using a modular and open architecture [Tamm88, 459-461].

To implement PIRS, the Tallinn group obtained basic Kronos design documentation from Novosibirsk, built the processor, and added a board to implement the object server functions. A prototype unit was completed in March, 1988, shortly before the termination of START.

Kronos design documentation was sent to a number of other organizations which manufactured some units of their own. At least four industrial organizations, the KamAZ Automotive Factory, the Institute of Atomic Energy, and the ELAS research institute in

Zelenograd (under Minelektronprom), and the Scientific Research Institute of the Aviation Industry decided to use the Kronos design to manufacture processors for themselves.

Each organization built its own Kronos, and made minor modifications to the design to suite their own purposes. For the PIRS workstation, the Institute of Cybernetics added an additional board to handle graphics functions [Tamm88]. The ELAS institute changed the instruction set considerably. In short, there was no “standard” Kronos. Marchuk’s group continually improved the processor and among other things adapted it to run with three different bus architectures, the Q-bus, Multibus, and Multibus-II.

Two operating systems were developed to run on the Kronos. Excelsior, supporting a Unix-like interface in a multiprocessor environment, was written in Modula-2 for more efficient execution [Kuzn86, 15; Kuzn89, 3]. Although providing a similar interface to the user, the Unix operating system was also implemented. The primary reason for this was that Tyugu’s team, developing the NUT environment in C rather than in Modula-2, and the developers in Moscow preferred an operating system based on C. Preferences in both camps were strong, and rather than reconciling the differences and settling on one operating system, both were supported.

### 6.5.2 START Operation

Whatever the earliest intentions may have been for close cooperation, START was realized as a loosely-coupled set of complementary research projects under a unified funding umbrella. Contact between the groups consisted primarily of one conference per year, held at an Academy of Sciences *dacha* outside of Moscow. Each January, representatives from each laboratory met with representatives from the Academy of Sciences, the GKNT, industry, and others for approximately ten days to discuss organizational, financial, and technical issues. Discussions here were intense, but individual groups operated in a much more isolated fashion during the remainder of the year.

Tyugu and Narin'yani had worked together in the past on artificial intelligence problems, but other than the conferences, the main point of overlap between the participating laboratories was to have been the Kronos. Developed in Novosibirsk and scheduled for production in Kiev, the Kronos was slated to serve as the hardware platform both for Tyugu's NUT object-oriented software development environment and Bryabrin's applications software. But even this level of overlap was not fully realized, largely due to delays in manufacturing the processor and the lack of the systems software—specifically C compilers—needed by the Tallinn and Moscow groups. Porting the existing applications to the Kronos would not have been difficult had standard C compilers existed. Although projects to develop such a compiler were initiated in Marchuk's laboratory and in Tallinn, they were never fully completed.

In spite of administrative efforts, the technical characteristics of a project sometimes reinforced the technical isolation between groups. For example, while most technical decisions in the MARS-M project were made within Vishnevskiy's laboratory, decisions about the structure of the system kernel for the control processor involved Kotov and Marchuk. Kotov, Marchuk, and Vishnevskiy felt that a C or Modula-2 compiler should be built for the control processor and that this work could be done outside of Vishnevskiy's laboratory. In this manner, the MARS-M project could be more tightly integrated with other START projects at the Computing Center. Mikhail Dorozhevets, chief designer of the control processor, knew that the hardware and software issues for the machine were too tightly coupled to be developed by different groups. The synchronous scheduling of instructions and other hardware issues were very close to software issues and would have been very difficult to develop separately. Dorozhevets was eventually given a year to develop the system kernel which he did as part of his Ph. D. (*kandidat*) dissertation.

Within the Computing Center of the SO AN SSSR, two over-lapping organizational structures were in effect: the traditional laboratory-based structure and the newer “working-brigade” structure. The traditional structure of Academy research institutes was based on a rigid structure of laboratories. Researchers were assigned to laboratories permanently. Once established, laboratories were seldom changed; once assigned to a laboratory, researchers rarely moved to another.

The working-brigade arrangement, one of the first like it in the Soviet Union, arose out of the need to carry out so many different projects within a limited number of laboratories. There were some projects which required the participation of people in different laboratories, and there were too many projects to be accommodated within the four laboratories involved in START. Some means were needed to improve communication and management, and to accommodate all the necessary projects. The urgency of the START program made it possible to overcome the resistance of the traditional laboratory structure. In effect, all laboratories were made temporary and new brigades were created. Two brigades were involved in MARS-M development, and three were involved in the Kronos. These brigades contained researchers from more than one laboratory. There were no brigades which included individuals from more than one institute, however. When START ended, researchers returned to their original laboratories.

Although the VNTK increased organizational flexibility, it was not completely free of bureaucratic entanglements. In particular, the mechanism provided few incentives for industrial organizations to participate, and many procedures still had to be approved by a number of state agencies. Shortly after START had been formally initiated, Kotov enumerated some of the difficulties remaining [Ntr85]:

Naturally, everything is not going as smoothly as of now. Experience shows that directives alone are not sufficient for the practical participation of a base organization in the work of the VMNTK.<sup>4</sup> It is evident that certain changes in the economic mechanism are needed to ensure the interest of organizations in the progress and results of the work of the temporary team. Moreover, the system of managing the VMNTK still is not flexible enough: it is difficult to maneuver the staff of specialists, the wage fund, and subcontracting and specialized work. For example, in the course of research, as some problems are solved and others arise, there may be needs for new specialists and for changing the professional structure of the team. But it is difficult to transfer headquarters from one organization to another, because this requires agreement by many parties.

Funding for START was in some respects unconventional as well. Usually, proposals for projects were submitted to a single organization—the GKNT, the Academy of Sciences, a Ministry, etc. In the case of START, several sources of money were combined into a single project, primarily from the GKNT and the Academy of Sciences. Although this added to the complexity of the proposal, the fact that G. I. Marchuk at this time both was chairman of the GKNT and had close ties to the Academy of Sciences counterbalanced the additional complexity. During the three years of its existence, START received on the order of five million rubles per year.

## 6.6 The Post-START Years (1988-1991)

As planned, prototypes of three types of computers and many software systems were completed by the time START formally ended on April 1, 1988. In accordance with the laws on VNTK, START was disbanded. An interadministrative commission consisting of representatives from the GKNT, the Academy of Sciences, and the State Committee on

---

<sup>4</sup>VMNTK, elsewhere referred to as VNTK, stands for “temporary *inter-branch* scientific-technical collective.”

Computer Technology and Informatics (GKVTI) signed an act for the acceptance of the hardware and software systems, which included the MARS-T parallel processor, the MARS-M scientific processor, and a family of Kronos processors [Manu88; Sots880515]. Public statements by participants and policy makers and press reports had commented favorably about START throughout its existence, and with the stated goals met, praise continued [Manu88; Agam91]. Besides praising the specific research achievements, reports placed considerable significance on the fact that the work had been accomplished in a shorter timeframe than would otherwise have been possible. START's deputy director Yevgeniy P. Kuznetsov commented that "Over 200 scientists, engineers, and programmers from Novosibirsk, Moscow, Tallinn, Kiev, and Severodonetsk accomplished in three years a research program which under traditional approaches would have required not less than five years" [Sots880515].

Several factors contributed to START's success. First, the participants were, for the most part, young researchers eager to spend long hours working on interesting projects. The fact that START was unique in its organization and had a high profile and high-level support increased motivation. The promise of a 3,000 ruble bonus (roughly equivalent to ten months' wages) upon completion of the work added a concrete financial stimulus. The nature of START financing further differed slightly from past projects in that START leadership was given more freedom in how to spend the money it received. Another contributing factor was the fact that work had been organized in brigades. Although not necessarily a primary reason for START's success, this organizational form did facilitate communication and management. Under the conventional management structure which includes laboratories, sectors, and divisions, much time is spent acquiring approvals at the different levels for the next stage of the work [Manu88].

While work proceeded faster than it might have in projects with a more conventional organization, one should keep in mind that the START achievements were the product of more than three years of work. The MARS-M had been under active development for five years before START was created. Similarly, Tyugu's PRIZ system and Narin'yani's AI systems had been first prototyped several years before START.

In spite of the stated success of START, the enthusiasm of researchers and policy makers, and the interest in furthering the work expressed by a number of industrial organizations, the transformation of START into a new form was not smooth. The formation of START's successor from 1988 onward was strongly influenced by the changing and uncertain environment, the beliefs and strategies of the participants, and the nature of the technologies. The evolution of START's successors is characterized by a measure of casting about—given the opportunities and constraints of *perestroika* and the Soviet economy—for an appropriate structure, domain of activity, and stream of resources to keep the organization and the research alive. In particular, it reflects a tension between commercial and academic activities, between applied and fundamental research.

#### 6.6.1 Organizational Transformation

The post-START developments were characterized by a fragmentation of efforts. A unified successor to START, incorporating industrial enterprises together with the basic START laboratories was never created. Instead, the START constituents each pursued further development and commercialization of their work independently, experimenting with a number of new organizational forms over time. When START was terminated in April, 1988, the Impul's Scientific Production Association also ended its involvement.

Many factors played a role in these developments. With the end of the START program, new sources of funding had to be found. The GKNT and the Academy of Sciences were not as willing to fund the research past the prototype development stage. Large-

scale funding from a few sources became difficult to obtain. It could be argued that multiple, smaller projects pursuing their own funding would have better chances of success. The organization of a unified program incorporating multiple ministries and administrations would be, for reasons given above, difficult to carry out, especially during times when laws on organizations were changing regularly, causing an overall slow-down in the bureaucratic approval processes. At the same time, laws enabling research institutes to enter into joint ventures with foreign firms, establish cooperatives, and engage in more direct contract work with industrial organizations opened new possibilities for attracting funds. In the balance, the forces pushing towards a smaller, more decentralized arrangement proved stronger than the desires to maintain a unified, centralized program.

Kuznetsov placed considerable emphasis on transferring the results of research into commercial production. While this has always been a stated goal in the Soviet Union, and START had been oriented towards applied research from the beginning, the intensified demands of *khozraschet* made the production of a marketable product a greater necessity [Manu88, 21].

START's termination was followed by the creation of a host of organizations and organizations within organizations which carried out a spectrum of activities, from purely commercial trading, to contract work, to applied and fundamental research financed through state funding: academic institutes, joint ventures, and cooperatives.

#### 6.6.1.1 Industrial START

As START neared its termination, preliminary plans for a continuation of the work were being made by Kotov, Kuznetsov, and others. A primary goal of the plans and strategies was to keep the basic goals of the research, but also to commercialize the results of the START program. Preliminary plans called for using the "implementation belt" of industrial enterprises created around a core of academic and branch science re-

search institutes [Manu88; Sots880515]. Industrial organizations had been involved in some aspects of the research, but only for the purpose of facilitating the construction of prototypes. A natural extension of START was to establish a new, more permanent structure with a much greater industrial component which could take the research results and put them into series production.

Industrial organizations would provide more than facilities. A factor contributing to the success of START was the high motivation of the young researchers who were able to work on interesting and important projects. The conversion of the results of a research project into a commercial product involves a great deal of “uninteresting” work with low scientific content, such as writing device drivers, compilers, editors, etc. A new organizational arrangement was needed which would enlist technicians to carry out the work necessary to prepare a commercial product, and enable researchers to continue to work on “interesting problems.” As Aleksander Marchuk stated, “We don’t feel it is our job to develop [basic systems software]; our job is the scientific work.”

In 1988, it was not unreasonable to expect that such a structure could be organized. START enjoyed considerable good-will among policy-makers. A number of Soviet and foreign industrial organizations had expressed an interest in the work [Manu88, 21; Sots880515]. Nevertheless, changes in funding levels, relationships with industry, and the overall legal and economic environments favored alternative arrangements, discussed in the following sections.

#### 6.6.1.2 Joint Ventures

After the law on joint ventures was passed in January, 1987, Kotov’s group worked on establishing the POLSIB Polish-Soviet joint venture between the Siberian Department of the AN SSSR and several Polish electronics industry enterprises [Tryb88; Alek88, 3]. One of the Polish organizations, Metronex, had been selling computer systems in Siberia

since the mid-1970s, so existing business contacts made the Novosibirsk-Poland link a natural one.

The joint venture was formally established on June 21, 1988 [Tryb88b]. The Polish enterprises were to develop monitors, printers, and other peripheral equipment, and the SO AN SSSR would develop software products [Alek88; Tryb88]. Together, the partners were to expand distribution networks within the Soviet Union and Poland, providing not only hardware, but also custom-designed hardware/software systems, installation, training, and support services [Groc88; Tryb88b]. The initial plans called for the commercialization of three projects: the MRAMOR publishing workstation, the Kronos processor, and the Alisa local area network. The MRAMOR workstation facilitates the typesetting of newspaper text by supporting text input, editing, formatting, and font and pitch selection [Yers86; Valk87]. The Alisa is a token-based network supporting SM and Elektronika computers [Psas87; Kata88; Kuch90]. At the time of this writing, only the Alisa project has survived, although development and marketing operations for it have been moved to Moscow to bring them closer to the market. Several tens of MRAMOR workstations were manufactured, including some for the newspaper Pravda, but they were unable to compete with the desktop publishing systems imported from the West. The Kronos processors never entered production because the Ministry of the Electronics Industry (Minelektronprom) refused to sell the necessary chips to Poland.

From 1988 onward, Kotov, Tyugu, and Narin'yani made on-going efforts to find partners for joint ventures to refine and market their projects. Kotov visited the United States during August, 1988 with others from Soviet policy making, academic, and industrial bodies involved in computing. They visited numerous companies along Boston's Route 128 and in Silicon Valley [Netm88; Meye88]. On a trip to Silicon Valley in 1988, Gorbachev's chief economic advisor, Abel Aganbegyan, searched for American software

companies which would be willing to establish joint ventures with the Academy of Sciences. The MASTER integrated spreadsheet, wordprocessing, and database package, developed under the START program in Moscow, was among the examples of Soviet expertise which he demonstrated [Sese88]. Neither of these trips produced concrete results, however.

In 1989, A. G. Marchuk and two other Kronos engineers visited individuals in Utah who were also involved in the design of Modula-2 hardware and software systems. Although there was considerable professional overlap between the two groups, collaboration was impossible because of government restrictions. Because the VLSI CAD system used by the Americans had been developed using Department of Defense funding, the Russians were not allowed to see it.

#### 6.6.1.3 Institute of Informatics Systems (ISI)

The organizations formed to carry on START work reflect the desire and need to maintain both fundamental and applied research, further the research already done, and to draw on as many sources of funding as possible. Kotov's efforts were primarily oriented towards the creation of a new institute, ultimately named the Institute of Informatics Systems. Called informally "The Siberian Program" or "START-II" before it was officially created, the institute was to employ 200-300 individuals, including both START participants and some personnel from an artificial intelligence center in Novosibirsk. Officially a research institute of the USSR (Russian) Academy of Sciences, ISI is involved in both government-sponsored fundamental research and applied research supported through contracts with industry.

ISI was the culmination of a drawn-out, stop-and-go effort which had begun in the late 1970s to create a new institute. The details of this effort are not clear, but a major element was the desire to achieve a greater measure of financial and directional inde-

pendence from the Computer Center of the SO AN SSSR. Serving the entire *Academgorodok*, the Computer Center served a large and varied constituency. The allocation of resources to meet the needs of all was fraught with inequities and political agendas. The laboratories saw a considerable fraction of the funds allocated to them drawn off into the overhead of the Computer Center. Such practices were, apparently, not uncommon nationwide. After Marchuk left to become chairman of the GKNT, a decision was made not to form a new institute, but to establish something like a special design bureau (SKB) which would operate on *khozraschet*. The drawback of such an arrangement, however, was that the SKB would not be able to carry out fundamental research—very important to Kotov and the other researchers—so no action was taken, and discussions continued. START offered a measure of independence, but it was not permanent.

The design of ISI evolved in response to changing conditions. Although the structure was to be more traditional than the original START, requiring a more typical approval process, early plans already reflected the new economic and organizational opportunities. For example, in August, 1988, Kotov anticipated that ISI would consist of a number of small cooperatives which would help provide maintenance and software development services, as well as more traditional laboratories. The exact form would depend on the interaction between local participants and higher-level organizations. The latter would have to approve the overall structure and participating organizations, while the former would have the freedom to specify which cooperatives would be formed and what their missions would be. A small cooperative, Prolog, was in fact formed, but it proved not to be profitable [Kron89]. It is not clear how many cooperatives were formed by individuals based at ISI or the Computing Center. There were few formal relationships since such linkages were apparently viewed with suspicion by the Presidium of the Siberian Department of the Academy of Sciences.

The bureaucratic process of forming the new organization took a full two years, for a number of reasons. The institute had a more traditional form than START and the approval process could therefore follow a well established path through the bureaucracy. However, because 1987 and 1988 were years of considerable change in the legal framework for organizations, approving bodies such as the GKNT were postponing decisions in anticipation of a change in the regulations and laws, slowing the progress of traditional requests. Once the proposal for a START successor entered the pipeline, it took longer to gain approval than would have been the case under other circumstances. Nevertheless, the approval process both for START and for START's successor was simplified considerably by the fact that no new facilities needed to be built or acquired. In Novosibirsk, Tallinn, and Moscow the participating laboratories remained in their same buildings. When ISI was formed, it remained physically within the Computer Center of the SO AN SSSR on a rental basis.

A proposal to create ISI was initially submitted to the Presidium of the Siberian Department of the AN SSSR, then to the Presidium of the AN SSSR, then to the GKNT which had to approve the funding. Finally, resolutions needed to be issued and signed by the Council of Ministers of the RSFSR and the USSR Council of Ministers, signed by the USSR Prime Minister, Nikolay Ryzhkov. The proposal was submitted to the Presidium of the SO AN SSSR in 1988, and throughout most of 1989 and part of 1990 was slowly pushed through the bureaucracy. ISI was finally created on April 1, 1990. [Izv900606]. At that time, ISI had a budget of 3.65 million rubles, one million of which came from the GKNT for the development of the Kronos processor, plus a few hundred thousand rubles in contracts with industry . While START was funded entirely by the GKNT and the Academy of Sciences, ISI relied much more on industrial contracts. Through 1991, 100%

of ISI's wages from came from Academy allocations, but other budget items were financed through contracts with industry.

The proposal was not submitted until several months after START was terminated. Kotov and others felt that an interim period was necessary during which, they hoped, the changing legal and political environment would stabilize and they would have a better understanding of the best way to proceed. As an interim solution, the laboratories involved extensively in the START project were grouped together into a formal division within the Computing Center. In contrast to their pre-START status, the division kept the financial independence gained under START. The new division was included as a separate line item ('*otdel'naya stroka*') in the Computer Center's budget.

ISI gained not only the ability to determine more easily the direction of research, and control its finances, but also the ability to engage more easily in commercial activities and the freedom to pay employees higher wages than would be possible under state budget funding alone.

ISI was structured partially along traditional lines. Although non-traditional in some of its activities, the structure, and the people filling key posts, still had to be approved by the Presidium of the AN SSSR.

MARS research continued while deliberations to form a new institute were in progress. A sharper division of labor between those working on commercial products and those conducting research arose. The group working on the Kronos processor divided into two portions, one to continue to advance the processor, and one to focus solely on developing a prototype of a commercial Kronos-based workstation. Researchers from the local Special Design Bureau of Computer Technology were enlisted to participate in the hardware development, and some contacts were made with the Parma Scientific Production Association to develop the Unix-based systems software [Kron89]. Thus a loose co-

operation between factories, cooperatives, research institutes and design bureaus was established, even though no formal organization uniting them had been created.

When START was terminated, researchers returned to their original laboratories and the traditional laboratory again became the dominant structure. Some proposals were made again to base the work on a more flexible, task-oriented model, but these met with resistance. Dorozhevets proposed creating new groups to continue the MARS-M development, but this idea was at first not supported by the scientific council because it wasn't clear what the main duties of the heads of the laboratories would be if such groups had the freedom to set their own research agendas. Other unresolved issues were how finances were to be handled, and what kind of contact such groups would have with other organizations.

Attitudes changed when the budget tightened in 1991. Restricted resources forced each laboratory to think harder about how it was going to survive. Dorozhevets had ideas for finding work for about half of the members of the laboratory and was given permission to organize a group as a self-financing entity within the institute. This reduced the pressure on the laboratory's budget considerably. Other groups were formed and flexible, temporary groups soon became the dominant model of the institute. The institute leadership allowed groups to find their own contracts on the condition that they pay the institute a portion of their earnings to cover their use of institute facilities and resources.

The rigid laboratory structure continued to exist as the umbrella under which basic research was carried out. Research institutes within the Academy of Sciences could not be 100% involved in commercial or applied research. The laboratories were responsible for carrying out scientific research. They received money from the Academy of Sciences for carrying out basic research, and were required to give annual reports to the institute and the Academy Presidium about what had been accomplished over the year. The labo-

ratories continued to serve as the organizational construct through which the work was financed, carried out, and accounted for.

At the same time, the science budget was no longer sufficient to support such laboratories. By mutual agreement between institute leaders and laboratories, laboratories at ISI all operated on the principle of *khozraschet*. The goal, of course, was to make each laboratory financially self-sufficient and accountable. This reduced the administrative burden for the institute's leadership, and gave individual laboratories more control over their own finances and a greater share of the money from contracts which they found. The temporary working groups enabled laboratories to supplement their incomes. Out of the money they earned from contracts, they augmented researchers' wages, returned a percentage of their earnings to the laboratories, and gave another percentage to the institute. In the case of the MARS-M groups, researchers were spending 90% of their time on applied research, and 10% on basic research.

In practice this system worked, although a negative side was that the laboratories were more protective of their ideas and work than earlier. Formerly, laboratory members freely shared ideas, experiences and advice with other laboratories. Later, when the correlation between an idea and revenue became much tighter, informal sharing decreased. It also became more difficult to shift people from one laboratory to another because of the questions of who would bear the cost.

Although in principle it was possible for the working groups to include members of different laboratories, as of 1991 there was no cross-over between laboratories.

#### 6.6.1.4 Commercial Start

Besides the creation of ISI, other avenues were pursued with at least partial success. Kotov and others created a strictly commercial company—also called Start—in an effort to promote the Kronos commercially and provide a source of income which could be used

to further work at ISI and pay the workers adequate wages. This organization, independent of ISI although Kotov and others served in the administration of both, was organized as a stock company. It was formed after the START project ended but before ISI was officially formed.

In practice, no actual development work was carried out in commercial Start. Organizationally, it consisted only of Kuznetsov, who became the director, and his book-keeper. Controlled by Kuznetsov, Kotov, and Marchuk, commercial Start became an umbrella under which a variety of commercial activities were carried out. The commercial Start obtained money through bank loans, contracts with the Computing Center (i.e. from the Computing Center's budget), and through other commercial activities.

Individuals from Computing Center laboratories were hired to continue Kronos development. Commercial Start became a mechanism through which the real salaries of researchers could be increased, since for the same work—the Kronos—they were receiving both their wages as employees of an Academy of Sciences research laboratory, and payment for commercial activities through Start. Commercial Start was also to serve as the clearinghouse for Kronos processors, when they finally entered series production. The Kristall factory, part of the Mikroprotessor Scientific Production Association in Kiev which worked on constructing a Kronos chip set, would sell the processors to commercial Start, which would oversee the manufacture of Kronos workstations and retain the profit from their sale.

#### 6.6.1.5 Other Developments

Narin'yani and Tyugu also created new organizations. Narin'yani created an organization called Intelligent Technologies to carry out the AI and software development research. Initially, Intelligent Technologies was loosely coupled with ISI. Although administratively distinct, Intelligent Technologies did have one department which was actually

a part of ISI. Later, however, this laboratory was moved to Moscow, the location of the rest of Intelligent Technologies, to be closer to the markets.

Additional ideas for keeping not only the START work but also the *Akademgorodok* viable were proposed. Discussions between *Akademgorodok* and Berkeley, California centered on the establishment of sister-city ties. Conversations were also held with representatives from UNESCO. The president of the SO AN SSSR was very interested in developing foreign contacts and Kotov, desiring to be a pioneer in these efforts, proposed a network of “villages” around *Akademgorodok*, linked with electronic networks, which would provide housing and laboratory space for groups of specialists from around the world. The focus was on creating small, flexible groups of people who could work on focused projects for a limited period of time, then disband and be replaced by others. These ideas never passed the discussion stage.

#### 6.6.2 MARS Research

Following the termination of START, MARS-related research continued existing development lines. Although accepted by state commissions in 1988, further testing, refinement, and implementation of the projects had to be carried out.

##### 6.6.2.1 MARS-M

The MARS-M underwent testing and debugging from 1988 to 1990. In 1990, all funding for the MARS-M ended and the approximately 15 individuals who were still working on it had to find other projects to work on. Fundamentally, the reason was a lack of state funds to continue the program. A contributing factor, however, was that the El’brus-2 program, into which the MARS-M had been incorporated, had also come to an end. The MARS-M could not be incorporated into the El’brus-3 program because a pol-

icy decision had been made at ITMVT not to incorporate any specialized processors into the new machine.

The Ministry of the Radio Industry (Minradioprom) expressed an interest in 1990 in the development of a specialized processor incorporating some (but not all) of the MARS-M ideas with a peak performance of 50-60 Mflops. The process of coming to agreement on the project involved multiple iterations of design proposals and corrections. A letter of intent, indicating funding on the order of 10-20 million rubles over five years, had been signed. This project was never initiated, however, because of a lack of funding.

#### 6.6.2.2 MARS-T/Kronos

Considerable time and expense were spent during the post-START years developing the commercial Kronos workstation. The Kristall factory worked on manufacturing industrial quality Kronos processors, and systems and software development work for the machine continued in Novosibirsk.

Kotov had discussions with representatives of factories besides Kristall, including some from the United States, but apart from those with the ELAS Plant in Zelenograd, none produced any results. A Swedish firm trying to set up a joint venture with Minaviaprom (for which ELAS produced chips, in part for the space program) to manufacture embedded control systems was considering using the Kronos as the basic component. They were interested in processors that conformed to international standards, so Kotov was considering augmenting the Kronos to run under multiple modes, one Modula-2 based, and the other Unix based. This would not be difficult since some movement in this direction had already been made to accommodate those with a Unix/C preference within START.

The Swedes withdrew from this venture, and after this, ISI ceased working on Kronos hardware. The primary reason, according to Kotov, was that given the technological level

of Soviet industry, it was not possible to compete in hardware. Work on the conceptual and software components of the Kronos continued, however. The software platforms were ported to Intel-based machines.

#### 6.6.2.3 Technological Base

Besides adequate financing and support from industry, a viable development program must have appropriate tools. To carry out world-class software and hardware development, software engineering environments and computer aided-design (CAD) systems running on workstations with powerful processors and much memory are highly desirable. With the exception of the BESTA workstations assembled at ZIL automobile plant in Moscow based on imported Motorola 68030 microprocessors, such machines were not manufactured in volume in the Soviet Union. The Kronos was too experimental to be used for true development work. CoCom restrictions and high costs made the acquisition of Western workstations prohibitive. The best tools obtainable at ISI were Western personal computers with 386 microprocessors.

From 1990 onward, the Computer Center and ISI lost a number of key people to cooperatives and joint ventures. ISI was able to pay its employees among the highest wages in the Academy of Sciences, but cooperatives and joint ventures were able to pay still more. By the end of 1991, a number of key researchers found positions in the West. Some in Novosibirsk have left ISI for other opportunities within Russia. By mid-1991, ISI had lost roughly 25% of its employees.

#### 6.6.3 Relationships with Industry

The support of industry was critical to the success of applied hardware development. To be effective, support had to consist of both a willingness to participate in the program, and the technological ability to do so. Each hardware project—the Kronos processor and

the MARS-M-established relationships with factories, but the nature of these relationships varied. In both cases the relationships were an on-going source of frustration. In the case of the Kronos processor, the Kristall factory exhibited a willingness to participate (at least at the level of the scientific-production association and the factory), but lacked, or was not able to obtain, the necessary technological capability. In the case of the MARS-M, the VEM factory in Penza had the necessary technological capability, but lacked a strong willingness to participate.

#### 6.6.3.1 Mikroprotessor Scientific Production Association

While START was in progress, Kronos developers had reasonably good relationships with the Mikroprotessor Scientific Production Association in Kiev which had been involved in the manufacture of functional duplicates of Intel microprocessors as well as some control computers in the Elektronika family [Bork91; Soko85]. START had been able to contribute considerable funds, on the order of 1-1.5 million rubles per year to Mikroprotessor in exchange for their participation in the development of the Kronos. The main task at the first stage of the project was to design and manufacture a chip consisting of 30,000 transistors, using 3 micron technology, which was to contain the arithmetic-logic unit and register file for the Kronos. Researchers from Novosibirsk were to supply the logic design, but the implementation was to be worked out solely in Kiev. In principle, the division of labor was clear, but in practice, there was regular interaction between representatives of the Design Bureau at Kristall and Marchuk's laboratory to clarify inconsistencies and work out how the logic was to be separated out into the various processor chips. Relationships were best with the research and prototype development unit of Mikroprotessor, but the Kristall factory was also interested in the Kronos.

Ultimately, Kristall was unsuccessful in developing the processors. After approximately four years of work, it did produce a version of the chip, but one which reportedly

contained many errors and was not usable. The reason for this was ultimately technical, although higher-level policy decisions were contributing factors.

Kristall had been one of the organizations which, during the early 1980s, had been trying to clone the Intel 8086 microprocessor. For reasons that are not entirely clear, it was not successful. Perhaps the participating engineers were not sufficiently skilled; perhaps there were problems with their development technology. The end result was that Kristall had gained a reputation for being a less than a first-tier chip manufacturing facility. It is likely that for this reason the factory was not as over-loaded with orders for ICs as leading fabrication plants and was eager for additional work, such as developing the Kronos. Kristall may have been given this task because it had development and production capacity available and the Kronos project would not crowd out projects deemed more important.

Bringing in over one million rubles a year, the Kronos project was rather profitable for Kristall. Some speculate that one reason Kristall never produced a fault-free processor was that this would have meant the end of the development project and the associated rubles. START was terminated in 1988 and ISI, its formal successor, was officially formed in 1990. In the intervening years, however, money was still allocated to the computing center to continue START-related work. The annual amount totaled approximately 3.5 million rubles, part of which was money from the GKNT to continue the Kronos work. A new contract was signed between the computing center and Kristall and roughly a million rubles a year went to Kristall in 1988 and 1989. Only in 1991 did investment in Kristall end, with 355,000 rubles contributed in that year. Until that point, Kristall had been eager to continue development work, if not production.

In principle, the *perestroika* reforms, with emphasis on direct relationships between organizations and self-management and cost-accounting could have facilitated the intro-

duction of the Kronos into series production. In practice, these efforts were hindered both by the ministerial structure in Minelektronprom which reformed very slowly as well as the new incentive structures. In spite of the changes in the laws, production decisions in practice still required multiple levels of approval within the ministry. Here the Kronos encountered officials reluctant to alter existing production lines to manufacture a processor for which demand was not clearly large. In general, although the ministries could produce items in small quantities, there was little to be gained from this. They received little political credit for manufacturing small numbers of items designed by others, yet when production problems arose, they would have to bear the responsibility. At best, the scale of Kronos activities at Mikroprotssessor were modest; only about 10-15 individuals there were working on the project. Although approval was eventually obtained, retooling for the Kronos would not take place until 1990 at the earliest, when the new state orders were scheduled to begin.

Following START's termination, NPO Mikroprotssessor became less interested in the Kronos work for two reasons. First, less money was available from the Novosibirsk group. State funding for START had ended and although Kotov's laboratory was still obtaining resources through contract work and from the Academy of Sciences for research, overall funding was more fragmented and constrained. A more important reason, however, involved the changing nature of doing business at Mikroprotssessor. As institutes and enterprises were given more freedom and responsibility to become financially self-supporting, factories, with real production capacities, became the key economic units in associations such as Mikroprotssessor. While the Kristall factory had been supportive earlier, it increasingly found it more beneficial to manufacture simple components which could be exported for hard-currency than retool for a complex unit like the Kronos.

Speaking about his funding for Mikroprotsessor Kotov remarked, ‘‘It’s just money. It’s not enough.’’

#### 6.6.3.2 Penza Electronic Computing Machines Factory

The MARS-M prototype was build in conjunction with the Penza Electronic Computing Machines Factory (VEM). This factory had participated in the production of a number of Minradioprom computers such as the Ural series, the ES-1052, the ES-1066, parts of the El’brus-2 computers, and others [Glus79; Vino80; Bbc86; Info88]. It was through their ties with ITMVT that the MARS-M developers established contacts with the VEM Factory. It is difficult to determine why, precisely, this factory was chosen and not another, but some suggest that MARS-M development here would be less disruptive to main ITMVT activities than at the principal El’brus factories closer to Moscow.

As in the case of Kristall, the Novosibirsk team developed the logical design, and the design bureau associated with the factory worked out the physical construction. After the logical design of boards was completed, however, the Novosibirsk team used a CAD system running at ITMVT to do routing. Sometimes the logical designs needed to be reworked to make them easier to route. When the routing was finished, the designs were sent to Penza.

Unlike Kristall the Penza VEM Plant usually had more orders than it could fulfill, given its production capacity. The five-year-plans were taut, but over the course of the five-year-plan the plant would receive additional, ‘‘priority’’ orders. The director made decisions about which orders to fulfill first. In the case of the MARS-M, constant pressure from researchers and policy-makers was needed to get the MARS-M prototype built on schedule. Representatives from START in Novosibirsk traveled to Penza monthly for nearly three years.

The visits from Novosibirsk also were to make sure that the system got built in a proper manner. For example, certain resistors known to be unreliable were used in the El'brus-2. MARS-M developers knew that if these ICs were used in the MARS-M, the system would have enormous problems. They themselves established contacts with other suppliers to ship higher quality chips to Penza. Without regular supervision by Novosibirsk researchers, such chips would have been used to relieve bottlenecks in other production schedules, not in the MARS-M.

It was not always clear what type of outside pressure would be most successful in pushing the project through. The original order to begin constructing a MARS-M was signed by the deputy-minister of Minradioprom, N. V. Gorshkov, in 1982, with scheduling revisions signed in 1983. However, little or now action was taken on this order through 1985. In the first half of 1985, G. I. Marchuk, frustrated with the lack of action now that the three-year clock for START was ticking, called local Communist Party leaders who gave the director of the Penza VEM Plant a stern talking-to. Offended, the director dropped the priority of the MARS-M and let the project go idle all together. Work proceeded only at the end of 1985, after Marchuk had talked the Minister of Minradioprom, P. S. Pleshakov, and persuaded him to issue an order.

Through 1989, constant pressure needed to be applied to the Penza VEM Plant to continue work on the MARS-M. In contrast to Kristall, the Penza VEM Plant had the technical capability to carry out the work, but not the incentive. This changed in 1988, after START was terminated, however. Thanks to the introduction of *khozraschet* and decreasing state orders, the Penza VEM Plant did develop excess capacity and it became legally possible for the plant to profit from the production of computers. It became very interested in producing the MARS-M, since the prototype had been completed and the production documentation had been finished. It asked the Novosibirsk developers to help

organize mass production. It was, however, still necessary to find money to invest in re-tooling the production line, and arrange for the needed components to be manufactured and shipped to Penza. Before such arrangements could be made, funding for further development of the MARS-M dried up, in 1990. It is further interesting to note that while the Penza VEM Plant was eager to manufacture the machines, it had little interest in doing the work necessary to find and acquire the necessary chips. This task was left to the developers in Novosibirsk.

#### 6.6.4 Levels of Support

In spite of efforts to establish a viable organization, during 1990 and 1991 in particular, ISI's ability to function eroded. These years witnessed a decrease in research funding, waning support from industry, uncertain markets, the attrition of skilled researchers, and the a growing inability to acquire the tools necessary to carry out state-of-the-art research.

##### 6.6.4.1 State Support

Because the Computer Center and ISI were officially part of the Academy of Sciences, they were recipients of Academy financing, and participated in competitions for Academy and GKNT grants for fundamental research which began in late 1988 [Nemo88; Veli89]. In 1989, the Computer Centers in Novosibirsk and Moscow received many millions of rubles. Funding for research in this year was higher than it had been in previous years, but in the years that followed budgets were cut, and the purchasing power of allocations decreased sharply in the face of high inflation rates and worsening shortages of all kinds of supplies throughout the Soviet Union. In 1990, by the time wages of all personnel had been paid, ISI had only on the order of one million rubles to spend on research. Most of this was paid to the Kristall factory for work on the Kronos. Funding throughout the Academy of Sciences became desparate in 1992. After not supporting Yel'tsin during

the abortive coup in August, 1991, the Academy lost much sympathy among the Russian leaders.

To compensate, ISI sought other resources. The commercial banks could provide a source of temporary funding, although at high interest rates. In 1990, ISI was able to borrow 5 million rubles from a variety of sources to help get the Kronos workstations into production. Technically, ISI assumed 3.5 million rubles of debt; and the commercial Start, 1.5 million rubles.

At the time of this writing, ISI had little hope of remaining viable conducting applied research. By the end of 1991, commercial Start had paid off its portion of the bank loan, 1.5 million rubles. ISI, on the other hand, continued to experience a deep financial crisis, since the interest payments alone on the remaining 3.5 million ruble principle was approaching one million rubles a year. ISI's annual budget decreased significantly in 1991 as well. In 1990 the money allocated by the GKNT for the Kronos had been added to the Academy of Sciences' allocation for scientific research, resulting in more money per capita in ISI than in any other institute in the Siberian Department of the Academy of Sciences. Leaders of the Siberian Department frowned on this practice and forced ISI to give 650,000 rubles to the Computing Center, reducing the budget to three million.

#### 6.6.4.2 Non-State Support

Although ISI had originally been established to continue MARS-related work, other projects which could bring in resources had to be pursued. Maintaining sort viability became the highest priority. As Kotov said, "The most important thing is to survive."

In spite of Academy objection, commercial Start became heavily involved in the computer resale business. The basic budget of ISI in 1991 was three million rubles, but through commercial Start Kotov increased revenues by an additional several million ru-

bles. Through commercial Start Kotov obtained bank loans and dealt in trading computers, stocks, food, etc.

The weak support from NPO Mikroprotessor was in part a result of a weak market for the Kronos workstations. By the end of 1990, tentative orders for 30-50 units at 100,000 rubles each had been placed. Under the volatile economic conditions of these years, however, the market did not remain stable. Great uncertainty remained about how many organizations would be willing to pay 100,000 rubles for the machine. But a greater concern, according to Kotov, was that development of the workstation would be retarded by unexpected delays. He felt that the current customers were not just buying the workstations for its own sake, but in anticipation of future versions. If new machines were not delivered within a reasonable timeframe, even these customers would discontinue their support. This is what happened. Less than ten units were actually sold, and the remaining units remained in a warehouse in Novosibirsk. In the final analysis, the barriers to successful development of the Kronos hardware were too high for continued work to be realistic.

A major barrier for further production of the MARS-M was the lack of customers. Although the MARS-M was cheaper than an El'brus, it still relied on the El'brus I/O capabilities. The potential market therefore was limited to El'brus users. The actual number of customers was much smaller. The Institute of Catalysis in Novosibirsk expressed an interest in the machine for running simulations of complex chemical reactions, but required that a FORTRAN compiler be developed so they could use existing applications code. Unfortunately, early design decisions about what a fragment was made it very difficult to use programs written in conventional languages without rewriting them to make calls of address and execution fragments explicit. When it became clear that this was too difficult to do in the time requested, the institute withdrew its support. Another institute

from the Ministry of the Atomic Energy Industry expressed an interest in the MARS-M for running simulations of atomic reactors. When this customer found out that the MARS-M was still under development and not ready for such applications, it too lost interest.

## 6.7 Discussion

MARS research was initiated at a time when emphasis on applied science was increasing with the Academy of Sciences. With the help of powerful patrons, the work obtained high-level attention and support as part of the Soviet answer to the Japanese Fifth Generation Program. It enjoyed many advantages over other Academy high-performance computing projects and was a better than average example of what could be achieved in applied computer development within the Academy.

Although applied projects, the MARS computers (the MARS-M in particular) had a very strong basic research orientation. The MARS-M had a unique architecture incorporating a large number of novel elements. The MARS-T showed a stronger influence of Western work, but was designed as a platform for research in parallel processing. The machines had long development cycles, and neither reached series production.

During the reform period, MARS-related research experienced enormous changes in levels and sources of support, relationships with industry, and opportunities for organizational transformation. The projects and the organizations associated with them provide a good lense through which to view the changes, their impact on research within the Academy of Sciences more generally, and responses within R&D facilities to those changes.

A variety of organizational forms—intra- and inter-organizational temporary working collectives, cooperatives, joint ventures, and commercial enterprises—were experimented with in an effort to find a combination of research orientation and organizational form

which would achieve the principal goal: preservation of the research collectives and the principal research directions. In spite of these efforts, many projects, the hardware development components of the MARS projects in particular, did not survive.

In this section we examine the principal factors influencing the development of the MARS technologies and the structure of the organizations within which they were developed. We conclude with some comments on likely prospects for R&D in the field of high performance computing at the Institute of Informatics Systems.

#### 6.7.1 The Technology

The major factors influencing the development of the MARS-M and MARS-T are presented in table 6-2.

The MARS computers grew out of a desire to explore qualitatively new approaches to building computers. In contrast to other HPC efforts, such as the PS-2x00 machines which were geared toward industrial applications, the principal considerations of the MARS projects were research results—the generation, implementation, and verification of new ideas about machine architecture. In the case of the MARS-M in particular, this research philosophy resulted in a machine which had many interesting, novel, and even exotic features, but which was also very complex, incompatible with existing software, and not well suited for industrial use. The machine made contributions to the field of computer science by demonstrating the possibilities of integrating multiple architectural approaches in a single system, and combining static and dynamic scheduling in a VLIW implementation. It did not have the operational or price/performance qualities necessary to be a successful industrial machine, however.

The MARS Conception, embodying the principles of parallelism, modularity, asynchronicity, extendability, hierarchical organization, architectural integration of hardware

<p><u>Environment</u></p> <p>Growing allowance for applied research within the Academy of Sciences Existence of powerful patron (Marchuk)</p> <p><b>Loss of powerful patrons</b></p> <p>Laws on VNTK Three-year limit on VNTK Laws and conventions on structure of Academy research institutes</p> <p><b>Laws introducing principles of <i>khozraschet</i> in Academy research institutes</b></p> <p>Lack of specific industrial sponsors Non-existent market for MARS systems Japanese Fifth Generation Program Weak links with industry</p> <p><b>Growing willingness of factories to consider production of (profitable) Academy innovations</b></p> <p><b>Trends away from funding large-scale hardware development projects</b></p>	<p><u>Technology</u></p> <p>Nature of El'brus construction technologies</p> <hr/> <p><u>Technological availability</u></p> <p>Examples of Western research (including Lilith, transputer, horizontal architectures) ITMVT facilities and El'brus-2 construction technologies (established technology in mid-1980s)</p> <p>Difficulty acquiring necessary components Lack of prior experience in hardware development Inadequate facilities at Kristall factory Availability of 32-bit processors (early work on Kronos)</p>
<p><u>Organizational structure</u></p> <p>Traditional division/laboratory structure of Academy research institutes Flexible, temporary scientific-technical collectives starting with START</p> <p><b>Coexisting structures including traditional laboratories, temporary collectives, cooperatives, joint ventures</b></p>	<p><u>Beliefs (design principles)</u></p> <p>Mission is do make research contributions Explore qualitatively new approaches in a variety of research domains Demonstrate architectural principles through the construction of prototypes MARS Conception: parallelism, modularity, asynchronicity, extendibility, hierarchical organization, integration of hardware and software, etc.</p>
<p><u>Organizational slack</u></p> <p>Generous funding under START, and through 1990</p> <p><b>Rapid decrease in state funding, especially for hardware development</b></p> <p><b>Inadequate income from commercial ventures, contract work</b></p>	<p><u>Strategy</u></p> <p>Establish links with industrial R&amp;D and production facilities Cast MARS in cloak of Japanese 5th Gen. Proj. Use existing technologies for prototype construction</p> <p><b>Explore commercial opportunities</b></p>

Table 6-2 Factors Influencing MARS Projects

and software, etc. formed the framework for MARS-M development. These principles, with the exception of tagged architecture ideas, remained largely intact throughout the

implementation phase, even though they were not always implemented to the extent originally envisioned.

The MARS Conception played a significant, but weaker role in the design of the MARS-T. The four pillars of MARS—parallelism, modularity, asynchronicity, and extendability—remained in place, but a very different implementation path was chosen. The Kronos was initially developed outside of MARS research. When it was selected as a base component of the MARS-T, however, features such as the inter-processor communications channels were adapted to support the kinds of systems envisioned in the Conception. The goal of the MARS-T project was to develop a research vehicle, not an industrial machine. The MARS-T was designed to be a test-bed with which to explore various concurrent structures, communication methods and mechanisms, and the relationship between architecture and algorithms [Koto91, 37]. A guiding principle was that the MARS-T should provide the basic constructs for a wide range of potential structures. The same system should be able to be configured for coarse-grained parallelism as well as fine-grained; for shared-memory, and distributed memory, etc.

Although the MARS machines were primarily research vehicles, the systems did have to be built, both to satisfy research goals and to attract the needed political, financial and material support. A number of strategic decisions were made to facilitate development. One decision, more political than technical, was to wrap a number of projects into a politically attractive package—as an answer to the highly visible Japanese Fifth Generation Project. This strategy was successful and did help acquire much needed support. It did not, however, introduce any new technical requirements, other than those derived from a desire to improve the overlap between some of the START projects.

Another key strategic decision was to use existing technologies—components, tools, systems—to the degree possible. The MARS-M was built as an add-on to the El'brus sys-

tems and was built using El'brus components. The MARS-T incorporated memory from the PS-3000, standard peripherals, and series production chips. It can be argued that designers had little choice. Although the START program enjoyed considerable high-level support, the commissioning of new components and subsystems would have added new dimensions of complexity and delays that the project could ill afford. There were problems enough getting parts already in production.

Three of the most significant environmental factors were the support of influential patrons, the relationship with industrial factories and suppliers, and the lack of industrial sponsors. G. I. Marchuk was able to use his influence as Chairman of the SO AN SSSR and of the GKNT to promote MARS proposals within the various planning and funding agencies. The MARS projects reveal, however, the limits of influence even of high ranking officials, and sheds light on the nature of the gap between the Academy of Sciences and production facilities within the industrial ministries.

The selection of a factory for the MARS-M was based on existing working relationships between ITMVT and Minradioprom factories. ITMVT agreed to provide development tools and facilities, forcing developers to use the component and technical base of the El'brus. If the factory had not already been using technology, components, and materials geared towards the El'brus line, implementation of the MARS-M would have been *very* problematic. By using ITMVT connections, the task of finding a factory to build the MARS-M was simplified.

The Computing Center of the SO AN SSSR, the Academy of Sciences, and even the GKNT had no direct authority and little influence over operations at the Penza VEM factory and NPO Mikroprotessor. Marchuk had no ability to select the factories directly; the Ministries offered factories which suited their own purposes. Even after formal agreements had been made and resolutions signed, G. I. Marchuk could only influence a fac-

tory through persuasion. Much depended on the good will of Minradioprom officials, from the minister to the factory director. Steady pressure was required to get factories to fulfill their obligations.

Another characteristic of the environment was the lack of industrial sponsors who would set specific requirements for the machines. The work was financed through the GKNT and the Academy of Sciences with little, if any, industrial co-sponsorship. The basic requirements for the MARS-M were that it run, and have a peak performance rate of 20 Mflops. Within these sparse guidelines, operational characteristics were considerably less important; developers were free to define system characteristics themselves. Although (potential) user requirements came to play a growing role for the MARS-T (or, more precisely, the Kronos processor), this machine also was predominantly a research vehicle. In short, there was little user influence to restrain experimentation with “interesting,” but somewhat impractical architectural ideas. This is not surprising, however, given the emphasis within the Academy of Sciences on producing original research results.

A fundamentally important factor was financial and material support, an important component of organizational slack. The financing provided by the GKNT and the Academy of Sciences was a necessary, although not sufficient, condition for successful execution of the MARS projects. The MARS-M project ended in 1990 because insufficient funds could be found to continue its development. The GKNT and Academy of Sciences were not willing to fund the project past the prototype development stage, and there was no commercial market for the machine. The machine lacked such basic features as compilers for traditional languages (FORTRAN, C) and was, in 1990, not suitable for industrial use. Later developments such as the the willingness of the Penza VEM factory to manufacture the machines could not overcome the fundamental problems of a lack of funding and the lack of a market.

The possibility of developing a commercially viable, profit generating product spurred the development of the Kronos processor which, of all the MARS hardware projects, had the greatest emphasis on industrial use. Originally just the component processor for the MARS-T, the Kronos was spun off as a major development project because Kotov, A. G. Marchuk and others realized the many potential uses of the processor, both for research and for generating funds to support other research. Thanks to generous GKNT financing and large bank loans, Kronos development through 1990 had adequate funding. Its principal barriers to development and manufacture were technological. The inability of Kristall to manufacture a chip in a timely manner meant that by the time some Kronos workstations were manufactured, they were not competitive with comparably priced imported personal computers. Consequently, the termination of state funding for Kronos hardware development, the inability to earn money through sales of Kronos systems, and a lack of prospects for doing so made it all but impossible to continue Kronos hardware development. The commercial efforts coexisted with the more research-oriented work, but competed for resources and conflicted at times with individual researchers' desires to do "real science." This was especially true as the fortunes of the Kronos systems declined.

Technological availability had a profound impact on both the MARS-M and the MARS-T. Three important components of technological opportunity are ideas, know-how, and the technology necessary to construct the prototypes. The MARS ideology was formulated on the basis of a review of existing computer architectures and a few key implementation ideas—horizontal architectures in particular—were absorbed from Western work. In spite of parallels with recent Western work, most of the MARS-M implementation ideas originated in Novosibirsk. The role of Western ideas was considerably greater

in the MARS-T and Kronos projects which drew strongly and directly from work on the Lilith processor and transputers.

The MARS-M and MARS-T were the first high-performance hardware projects undertaken by their developers. Significant time was spent learning how to build computers and this lack of experience and know-how caused delays in the development cycle.

The availability of tools, manufacturing technology, components, and other supplies significantly impacted the construction and design of the prototypes, the length of the development cycle, and, ultimately, the lack of success in the marketplace. Had not the MARS-M developers been able to use technologies and facilities provided through ITMVT, the MARS-M would not have been built. However, the need to accommodate ITMVT practices forced a scaling back of design plans: limiting the number of processors, the word length, the number of functional units, etc. The linking of the MARS-M to an El'brus complex obviated the need to develop I/O capabilities independently. This feature would have seriously limited the market for the MARS-M even if the unit itself had been more amenable to production and use. By the end of the 1980s probably not more than fifty El'brus installations existed.

There were problems with the use of industry production facilities and series production components, however. The difficulty in acquiring the necessary components and the reluctance of the Penza factory to make the system a high priority stretched out development times. The real production time required for the machine would have been only a year if it had been made a top priority. Even if the Penza factory had made MARS-M construction a priority, it is debatable whether it could have been completed in fewer years, given the time amount of time spent hunting for components. Researchers from Novosibirsk visited numerous factories throughout the Soviet Union, spoke with deputy

ministers, and tried to work through military organizations. For want of chips, work on the MARS-M was halted for months at a time.

The nature of the MARS-T implementation was significantly impacted by the Kronos processor which, while not built specifically as part of a parallel system was locally available to developers. Had the MARS-M control processor been used as the basic computational element as A. G. Marchuk originally proposed, the basic software environment, physical size, and interconnect mechanisms would have been very different.

It is not possible to speak of a technological trajectory for the MARS computers alone, since only one generation of MARS-M and MARS-T were built. It is therefore difficult to determine fully which design principles and features were part of some overarching technological paradigm, and which were simply incidental to these machines. It is clear that certain guiding principles were at work. These have been discussed at length above.

Had hardware development continued, which elements of the guiding principles would have remained? We may never know, but there is reason to believe that significant elements would have been rejected, or at least altered. The MARS-M is viewed by some of its engineers as overly complex, incorporating too many widely different architectural ideas. Furthermore, a basic element of the MARS Conception, that the future evolution of computers will be towards systems which can be statically reconfigured to adapt to individual applications, does not appear to be endorsed by world practice. Rather, an emerging trend in HPC appears to be towards multiple, heterogeneous computers linked together via networks in an integrated environment where individual applications run on the most appropriate computer(s). No reconfiguration of individual machines is required.

### 6.7.2 The Organization

The organizations involved in the research and development of MARS machines underwent regular transformation and reorganization during the period covered in this study. These changes reflected on-going efforts to find organizational forms which were most viable from a financial and technological perspective, the opportunities presented by the legal environment, and the nature of the technology itself. Some of these factors are shown in table 6-3.

One of the most consistent factors running through the transformations is the desire to achieve greater levels of financial independence and viability, to maximize the resources available to the organization. For years prior to the creation of START, Kotov and others had been searching for ways to separate themselves financially from the Computer Center and gain greater control over their own finances. The START program, besides bringing in welcome research funds, provided an opportunity to achieve a measure of financial independence. The formation of ISI in 1990 was largely driven by similar concerns.

The experimentation with joint ventures and commercial ventures reflected an effort to draw in resources from sources which were non-traditional, at least for Academy institutes.

The internal structure of organizations was shaped in part by another aspect of organizational slack: the financial pressures of shrinking budgets and rising inflation of the later reform years. These pressures supported trends towards decentralized control and responsibility of finances which gave both greater control and greater responsibility for finding sources of income to the individual laboratories or research teams. At the inter-organizational level the shrinking research allocations and trend away from large-scale projects also forced a fragmentation of effort. It became easier to find multiple smaller contracts than a few large-scale ones.

<u>Environment</u> Legislation establishing traditional organizational structure of Academy research institutes Administrative process for approving creation of institutes Legislation establishing VNTK <b>Legislation on cooperatives, joint ventures, other commercially-oriented structures</b> <b>Legislation on khozraschet in research institutes</b>	<u>Technology</u> Number of tasks needed to be carried out under START Nature of the individual tasks  <u>Organizational structure</u> Existing laboratory structure Organizational structures under START
<u>Technological availability</u> <b>Examples of use of cooperatives, joint ventures, etc.</b>	<u>Beliefs</u> Organization should be structured to maximize organizational slack, efficiency Research is core activity, and organization should be structured to facilitate it
<u>Organizational slack</u> Generous, unified funding under START, Continued funding for MARS-M, -T through 1990 <b>Rapid decrease in state funding, especially for hardware development</b> <b>Inadequate income from commercial ventures, contract work</b>	<u>Strategy</u> Experiment with new organizational forms <b>Give laboratories greater responsibility for own viability</b>

Table 6-3 Factors Influencing Organizational Structure Around MARS Projects

The internal organizational structure was also shaped by the nature of the technological developments, however. One of the chief motivations for adopting the working brigade organization in START was a mismatch between the number of existing laboratories, the number of projects, and the positions of the individuals who were to work on the projects. Although the formal ISI structure remained based on the traditional laboratories, the actual structure depended on the structure of the flexible teams.

Each organizational change (with the exception of the formation of ISI which was along more traditional lines) was enabled by changes in legislation which created opportunities for alternative organizational forms. In particular, the 1983 decree on temporary scientific-technical collectives, and the 1987 decrees on joint ventures and cooperatives,

opened the door to organizational experimentation. The 1987 decree on the transition of scientific research institutes to *khozraschet* made decentralization of financial responsibilities possible.

It is difficult to identify a premeditated organizational transformation strategy. The most consistent strategy to emerge over the years was one of paying close attention to the organizational opportunities available, and applying them to the greatest advantage.

### 6.7.3 Prospects

The short-term prospects for applied research in hardware develop are very bleak. Both the MARS-M and the MARS-T/Kronos have ceased to exist as hardware projects. Without the appropriate tools, component base, devices, talent, and funding, the institute cannot carry out world-class applied research. There are two alternatives: do commercial work with low research content and scientific interest, or concentrate on theoretical developments which do not rely as heavily on the availability of tools, components, and other material resources. Kotov's heart lies in research, so his desire is to scale back ISI operations to a core of talented individuals who can focus on theoretical development.

Doing research remains a strong desire of those at ISI, but survival remains a dominant goal. Small-scale contractual work, with or without significant scientific content, will remain a major activity until levels of funding for basic research rise again.

What are the implications of recent developments at ISI on its ability to conduct applied HPC research in the future? The prospects for improved idea generation are mixed. On the one hand, researchers have, in principle, greater access to ideas from the West. Thanks to electronic mail connections, researchers can interact with Western colleagues

in a fast, reliable manner.<sup>5</sup> Researchers have been able to travel and work in the West where they have gained considerable exposure to Western developments and practices. Thanks to the emphasis on scientific institutes' "paying their own way," researchers have increased interaction with industrial customers. The degree to which this contact advances science is variable, however.

On the other hand, several developments serve to reduce idea generation. The transition to *khozraschet* at the laboratory level has made laboratories more protective of their ideas, especially those that could have some commercial value. The present financial crisis and the accompanying lack of hard currency make it difficult to subscribe to Western publications and to travel to Western conferences where much exchange of ideas occurs. In the balance, the short-term prospects for the idea generation are quite poor, although the long-term prospects, assuming the basic state of the Russian economy improves, are considerably brighter.

Recent developments have eroded the political base of support for research at ISI. Marchuk gave up his post of GKNT chairman in 1986 to become the President of the Soviet Academy of Sciences. In the Academy elections in December, 1991, held in the aftermath of the attempted coup and breakup of the Soviet Union, he stepped down and was replaced by Academician Yuriy S. Osipov. The Academy of Sciences as a whole has lost much influence in Moscow because of its non-support of Yel'tsin during the August, 1991 coup attempt.

Recent developments have led to a fragmented pattern of resource allocation. Basic funding for fundamental research continues to be supplied by the Academy of Sciences, although at a level which is hardly sufficient to support ISI scientists. Additional income

---

<sup>5</sup>For example, [Doro92] was written nearly exclusively via e-mail between Novosibirsk and Tucson, Arizona.

is earned through contract work, and this has been augmented with some income through commercial dealings coordinated by the commercial Start. In principle, these mechanisms—particularly the centralized state funding—could supply the financing necessary for high-performance computing projects. In practice, the short-term prospects are very poor, given the catastrophic state of government finances and the decline in allocations for basic and applied research.

Recent developments have had both positive and negative consequences for the ability to prototype systems. On the positive side, the ability to form flexible research groups which are oriented around a specific project and which can draw in the scientists best suited to the task help shorten development cycles and focus research energies in key areas. Also, it has become possible to acquire Western technology which previously was accessible only through a long and uncertain acquisition process through the Academy of Science, foreign trade organizations, KGB, etc. CoCom restrictions have been raised considerably and much technology is freely available in Russia, for a price. On the negative side, financially strapped institutes despair of being able to acquire the basic technological inputs for research: CAD systems, high-quality components, subsystems, and instrumentation, etc. Good research only partially compensates for poor equipment and inputs. Under very difficult economic conditions, scientists are forced to spend extraordinary amounts of time earning money on the side doing tasks with little scientific content, searching for basic material necessities, caring for their families, etc. Development life cycles are drawn out considerably as a result.

There have also been positive and negative changes in the ability of the Academy institutes to move innovations into production. As demonstrated by the changing relationships between ISI and the factories involved with MARS development, the factories are increasingly willing to consider taking on the production which will use their idle capac-

ity. On the other hand, they are unwilling to tool for the manufacture of products which enjoy only a small market. An additional negative factor, observed in the case of the MARS-M, has been the unwillingness of the factory to invest the resources necessary to bring a prototype machine into series production.