CERN 90-08
12 October 1990

EUROPHYSICS CONFERENCE
ON
CONTROL SYSTEMS
FOR
EXPERIMENTAL PHYSICS

Villars-sur-Ollon, Switzerland

28 September-2 October 1987

B. Kuiper
Editor

GENEVA
1990

ACCELERATOR CONTROL SYSTEMS AT THE INP (STATUS AND PROSPECTS)

A.N.Aleshaev, A.M.Batrakov, S.D.Belov, V.V.Kargaltsev, V.R.Kozak, E.A.Kuper, V.A.Kupchik, B.V.Levichev, A.V.Ledenev, V.I.Nifontov, Yu.I.Oshchepkov, G.S.Piskunov, I.Ya.Protopopov, S.V.Tararyshkin, Yu.I.Eidelman

Institute of Nuclear Physics, 630090 Novosibirsk, USSR

Abstract

The historic review and current state of the accelerator control systems in the INP is presented. Questions concerning organization of distributed multiprocessor systems as well as specific features of measurement and control devices are discussed. The structure of the control system for the VEPP-3 collider, based on the use of CAMAC-oriented 24-bit microcomputers is described.

History and Principles

First experiments on the application of computers for accelerator control began at the INP in 1970 at the storage ring VEPP-3 when the necessity arose to increase the luminosity of the storage ring by changing the magnetic system of the experimental straight section in the presence of a circulating beam, the latter requiring high precision and matching of a change in the characteristics of the magnetic structure components. For this purpose, a second generation computer "Minsk-22" was employed, whose control program was written in machine codes. However, it has soon been found more expedient to use a dedicated computer. Since 1972 a deal of effort has been undertaken in the automation of both the existing accelerator facilities and those being under construction: VEPP-2M, NAP-M (in the 1970s), VEPP-4 (1978), SIBERIA-1, BEP (in the 1980s) and some others. Quite different computers were used: "Minsk-22", M-6000, Odra-1300, computers of the "Elektronika" series.

In the first systems only the most vital parts were automated, i.e the power supply for the magnets. Later the computer control covered a greater and greater number of the subsystems with the ever-decreasing share of manual control. So, the VEPP-4 facility was designed from the beginning to be completely computer-controlled. The use of duplicate manual control subsystems at the facilities of such a scale proves to be very expensive. This lack of the possibility of manual intervention in machine operation gives rise to a specific requirement for control equipment: the last commanded settings must retain in it irrespective of the computer availability. This absence of duplicate manual subsystems is characteristic of the up-date INP control systems.

When designing the very early versions of the VEPP-3 control system the decision has been made not to engage professional programmers. In fact, since they are not experts in the questions concerning the operation of particular subsystems of the facility, they are unable to write programs with due regard for the specific nature of a task. At the same time, the complexity and experimental mode of the operation lead to hardly identifiable, diffuse specifications in the definition of a problem. In these conditions, it seems more effective to engage in the development of applied software the persons who possess a practical knowledge of the facility operation.

At present the control systems of the main acceleration facilities of the INP comprise microcomputers compatible with the Odra-1300 series. The choice of Odra-1304 for the VEPP-3 control system in 1972 was largely a matter of chance. However, the use of this machine has had a pronounced effect on further development of control systems at the INP.

Computers of the Odra-1300 series are functionally analogue to the known family ICL-1900, designed in the late 1960s. They are classified as universal computers and had much better parameters compared to minicomputers which have been in common use in the 70s. The application of a sufficiently powerful computer to the control system has enabled one to reach a high level, qualitatively different from that provided by the minicomputers available. A considerable, at that time, memory size as well as high throughput offered the possibility of creating the neccessary software for control systems, exclusively in high level languages and, consequently, has made it possible to involve, in the software development the machine staff.

The open nature of the most dynamic component of control systems, the software and the possibility of its extension and modification by the machine staff are typical of most of the INP control systems. Consequently, the second principle which we tried to follow was the the following: applied software should be designed by the persons who are expected to use it. To do this, it is very desirable to ensure the possibility to use a high level language within the frames of an accessible and simple-to-handle system of program preparation.

In the course of creation and maintenance of a control system one has to get used to an active contact with a computer. As is often the case, the computer is responsible for a good deal of work, and automation and computerization begin hindering the work. In view of this, it is extremely important to treat the possibilities which can be in principle offered by a computer with due care, critically. The major purpose of automation is to save an operator from routine and exhausting work, the arrangement of convenient and effective procedures for measurements, a simple and clear presentation of information in terms of physical and technological quantities. One can start with introducing the closed loops through a computer only after the outlined above has been done. The advances and failures of many INP control systems have been largely determined by an extent to which the designers were successful in the consistence of the general and multi-purpose character of the approach with the real requirements.

Thus, the third principle which was taken a guide may be stated as follows: the universal and global nature of the requirements for a control system often excludes even the possibility of its creation and, therefore, it is very often unjustified to bring in the system, at the stage of design, the services which will be needed in the remote future only. On the other hand, the system should possess a potential for its extension and modernization, with the structure and ideology being kept.

Structure of the hardware and the choice of a computer.

In solving the problem of efficient communication between computer and control hardware we came through several stages. At first stage, i.e. when putting the VEPP-4 control system into operation, the most of the control hardware was made using mechanical standard for nuclear electronics called "Wishnya" as the so-called autonomous functional units (AFU). The AFU were designed for the solution of particular tasks on the basis of standartized PC boards: memory, state machine, interface with data transfer network, multi-channel ADC or DAC.

Simultaneously with the accelerator cycle, the AFU performed the measurements and stored measured values in the internal memory. The interaction with a computer was limited to writting or reading the files of structured data from that buffer memory. The OS overhead to initiate the transfer of a large block of information through the DMA channel of the computer had little effect on the total throughput. All the devices were connected to the computer through a serial data transmission network having a three-level tree-like topology.

The VEPP-4 control system was designed on the basis of such a scheme. It included 6 computers "Odra-1325": one machine for each of the subsystems such as an injector, an intermediate storage ring VEPP-3, main ring VEPP-4, and a high-frequency generator "Gyrocon". Later one more computer for supervision of the technological parameters of the entire facility (vacuum, temperature, radiation, etc.) was added. This system operated in 1978-84.

The life time of an accelerator lasts usually several decades. It is modified all the time because new ideas and new demands appear. The control system must evolve in time together with the accelerator, this process occurring through rapid advances in computer technology. Continuous modernization of the control devices on a long-range perspective has necessitated the change-over to a more universal, computer independent standard. At this time the only suitable was CAMAC. The entire set of equipment designed for VEPP-4, was intended to be connected to the "data transfer system of VEPP-4" and, consequently, it was difficult to apply it for other systems. Thus, a transition to the CAMAC standard was also stimulated by the need in the unified hardware in the frames of the whole Institute. Since 1978, almost all the electronic equipment intendend to be used in the control systems was designed in CAMAC.

System integration in the CAMAC standard has some bottlenecks, due to a high intensity of interaction with the computer, which is inherent to the standard. In essence, CAMAC is intended for the use of the so-called program I/O. In our case, Odra-1325 computers did not even posses appropriate instructions on the user level to perform the I/O - all had to go through the multitasking OS, which led to a very high overhead.

The first step in the solution of this problem was a development of a dedicated crate controller connected to the serial data transfer system and performing the simplest interactions with the modules in the crate. In the controller there was a memory containing the previously written control information: N, A, F, waiting of Q, LAM, a branch to the next instruction, etc. Such a controller makes it possible to generate internally the required low level control infomation and transmit to a computer only the "pure" structured data. Although this controller has enabled to achieve satisfactory time characteristics; however, in the multiprogram environment inherent to the control systems, the problem has arisen of an arrangement of the controller-to-programs interaction.

The next step was the use of an intelligent crate controller being a 16-bit computer with a simplified instruction set and a small memory. Its application encountered the known problems typical for many systems at the late 70s: a minicomputer, at the top level, and an intelligent controller based on a low-power microprocessor, at the lower one. The major disadvantage of these systems is the complexity of controller programming to be done by experienced programmers, and this violates the principle stated above: the software is designed by the physicists. For this reason, such controllers have found its application only in the fixed-task systems, where no modification of the programms was needed.

A radical decision was made in the late 1981 - to design an intelligent crate controller with the instruction set of Odra-1300 computer. This allowed us to modernize hardware and to retain most of the software. In 1983, such a controller, "Odrette", was ready for operation. It was first used as a slave processor in the Odra-based systems. In the summer of 1985 we abandoned the further use of the Odra computers and began to employ only the Odrettes.

Odrette occupies doublewidth module in the controller position of CAMAC crate. In fact, it is a 24-bit computer with Odra-1300 (ICL-1900) instruction set. Its basic features are:

- address space 4M words;
- on-board memory 64K 24-bit words;
- 0.4 MIPS;
- hardware implementation of floating point operations;
- User / System modes;
- multitasking support / memory management unit;
- CAMAC oriented I/O instructions;
- vector and vector / scalar operations;
- bit-slice implementation based on the AMD 2900.

The bit-slice implementation has offered the possibility to change the original architecture in the desired direction. Besides the introduction into the architecture tools to interact with the CAMAC bus, some important modifications, useful in real-time systems, have been done. Among such innovations was the mechanism of process switching. The point is that the availability of the multiprogramming is of principal value for a computer used in control systems. The systems designed to control the accelerator, in which the accelerator is not only a tool but also an object of research, are characterized by frequent changes in the regimes and configuration, by presence of a great number of research problems. The multitasking support, tradicionally attributed to large machines, appear to be useful also in an autonomous crate controller. Operating system regards all the tasks waiting for LAM signal or program "event" as being active. On occuring the expected event, the processor initiate, at the microprogram level, the needed program according to its priority. Thus, the most time critical part of the multitasking OS was implemented on the microprogram level. This led to more than an order of magnitude improvement in the time response.

CAMAC modules for the control systems.

One of the peculiarities of our laboratory is that almost all the electronic equipment intended for control systems is developed and manufactured in the Institute. The ever-increasing complexity of the accelerators and the experiments which they involved in often specify the requirements for the parameters and the structure of the modules being designed. The metrological characteristics, throughput and noise-immunity must be adequate to the operating conditions of the system.

For example, when solving the problems of the transportation of reference voltages in the power stabilization systems, digital-to-analogue converters based on a principle of pulse-width modulation (PWM) are in common use. An 8-channel unit, placed in the crate, generates width-modulated signals, transferred over galvanically isolated coaxial cable directly to the point, where the reference voltage is needed. There the PWM signal is converted into analogue form. Thus at minimal expenditure an accuracy equal to 0.01% is achieved. For precise control systems a PWM-DAC with an accuracy of the order 0.001% has been designed.

The control systems are also responsible for numerous multichannel measurements of DC and quasi-DC voltages. In this case, the sources of signals are often far from each other. To achieve the necessary interference suppression, integrating ADCs have been designed with variable integration time of the input signal. Depending on the required precision, the measuring time is program-ranged from 5 to 320 ms, the scale varying from 13 to 20 bits. The ADC accuracy is 0.01% in the 20-50 C temperature range. It has floating analogue part. The control is performed through optocouplers, while the power is applied via high-frequency transformers with minimum straight capacitance. The internal memory of the ADC contains 256 words. To arrange multichannel measuruments, the ADSs are equipped with 64-channel two-terminal analogue multiplexers. A multiplexer may be controlled by the ADC or from the CAMAC bus.

The necessity to analyse pulse signals has given rise to the new class of devices - transient recorders. These devices comprise a fast ADC, control circuits, internal memory and an interface allowing an on-line control of discretization frequency and a change of the input voltages range. The detection and analysis of the shape of pulsed signals in different systems by the ADC with further computer processing have enabled a number of important problems to solve:

- to monitor the behaviour (current and coordinates) of the beam during the first several thousands turns of injection;
- observation of small amplitude signals in presence of a heavy interference by means of a preliminary digitazation of the interfering signals in the absence of an useful one with subsequent substraction from the sum signal;
- detection and studying the evolution of the non-predictable events by writing the measured results in a ring buffer, with a stop when an event (say, a beam loss) appears;
- as a program controlled oscilloscope.

We present here the main parameters of the two most popular recorders. For the ADC101S, they are: 10 bit resolution, minimal sampling rate 1 μs, memory capacity 4 K words, 400 kHz bandwidth. For the ADC850S they are: 8 bits, 50 ns, 1 K words and 4 MHz, respectively.

Besides the devices mentioned above, a great variety of modules necessary to ensure the functional completeness of the set have been designed. The total number of different types of modules designed exeeds 100. There are the generators of time intervals, timers, meters of instantaneous values of the pulse signals, multichannel modules required to monitor such mass parameters as vacuum and temperature, position transducers, control registers, colour graphic displays, etc.

Software of the control systems

The software of the control system may be grouped as follows:

- control application programs which are developed, as has already been mentioned, by physicists and engineers;
- the software used as the tool in preparation and running the control programs, i.e. the system software which should be designed by specialists having the corresponding knowledge and skill in the field of computer science.

When designing control system it is necessary to choose a language for the applied software. The designers often consider the traditional high level programming languages like FORTRAN and Basic to be inappropriate candidates since they lack the specific features of a given field of application. The existing versions of these languages, intended for the work with the equipment, are more suitable for the automation of small set-ups. It is not surprising that when designing the early control systems, the majority of the designers have invented a language of their own, though in some cases the old good FORTRAN was used.

After choosing the language (and more often, simultaneously with the choice) the question raises concerning its realization; there are three ways of its solution:

-the language is provided by the computer vendor;

-designing of an interpreter to execute the programs written in the chosen language by a given computer;

-designing of a compiler to translate the programs from the language into the machine codes.

The possibility of applying the first way should be considered as rather a lucky chance, meanwhile more often the need arises for the one's realization of a language of its own. Since the interpreters are usually somewhat simpler, putting up with a certain loss in efficiency, one usually implements the interpreters which are responsible for some functions of the operating systems as well.

The software of the early INP control systems (VEPP-3, 1973-77; VEPP-2M, 1974; NAP-M, 1975) was based on the use of interpreters. The interpreter of the VEPP-3 control system on an Odra-type computer was a complex overlay program, written in FORTRAN, which implemented, besides the functions of executive system, the function of a text editor of the source programms-"processes", a librarian, a magnetic tape back-up server and many others. The programming language, specified by this interpreter, has offered the possibility of arranging, despite an awkward syntax, rather complicated algorithms in control processes and of using a dialogue and a graphic display. Many ideas on the arrangement of the control process have still remained useful and have found its implementation in modern control systems of the INP.

The control system for the NAP-M facility also used the interpreter at an Odra-type computer, but in this case it was written in assembler and, therefore, it was considerably faster and the overlay proved to be needless. The programs written in the language of this system, could contain the code insertions, to provide a maximum efficiency.

It seems rather simple to implement the programming language of the control system on the basis of an interpreter. This approach inevitably limits the possibilities of evolving the system: the access to the control system gets limited by the frames of a single exotic language, and the latter, because of its specialized nature, may prove to be inadequate to new problems. In addition, interpretation considerably slows down the execution of the programs.

With this in view, when designing the VEPP-4 control system, an interpreter has been dropped and the designers have aimed on developing a multitasking operating system. In this case, the system is not referred to any language and both the interpreting and compiling programming systems may be used. Of course, the software of the control system contains interpreters but they are largely used as the calculators to perform simple computations. All the programs, working in the control system, are written in a FORTRAN-like language TRAN whose compiler is an integral part of the program preparation system.

The program preparation system comprises a text editor, a compilation system, a library maintenance program and some auxiliary programs.

The compilation system consists of various translators, consolidator and loader. All the used translators generate ICL-standard intermediate codes and then a consolidator unites them in a single program. At this stage the fragments written in different languages may be consolidated in one program. The major language in the system is the TRAN mentioned above, which is an extended subset of FORTRAN. The entire compilation system is rather fast and it takes about 30 seconds to translate a program of average size (1.5-2 thousands of lines).

The system of program preparation relies on a rich library of subroutines, which offers to a programmer a wide choice of mathematical functions and also a set of procedures intended for interaction with devices of the control system, terminals, graphic displays and files, as well as interaction with the OS and other programs.

The OS of the VEPP-3 and VEPP-4 control system offers a standard enough set of functions of modern real-time operating system: memory management, processor sharing, arrangement of the access, through the system, to the peripheral devices, file management, synchronization of the control programs operation via the system of external interrupts, interprogram message passing, etc.

The content of the applied software for the VEPP-4 control system may be illustrated by a set of programs working in the main computer of the facility in the stationary regime:

- BANK program working synchronously with a peripheral timer and by means of which all the remaining programs get access to the equipment of the control system;
- a program-process providing the particle accumulation, a raise of the beam energy up to the energy of the experiment with the needed parameters of the beams;
- a program of "manual" work, which permits one to change the operational mode of any component of the storage ring;
- a program intended to measure the beam luminosity and polarization;
- a program holding the position of the collision point with respect to the angle and coordinates.

Besides the above programs, a program is resumed in each 5 minutes which checks the agreement between the state of the storage-ring components with the values held in the machine's data base. In addition, the operator can initiate the programs which measure the parameters we are interested at the moment, and are not subject to regular supervision.

The control systems mentioned above are distinguished by the use of a single control computer. With the appearance in the system of an "intellect" allocated in several lower-level microcomputers, some difficulties in the arrangement of their effective use have been revealed. In similar multi-processor structures, it seems difficult to divide the functions of the control programs into the pithy fragments suited for separation in some of the processors because of the strong information links between the subsystems of the facility. For this reason, Odrettes were initially used as auxiliary low-level processors responsible for the solution of isolated enough and information-exclusive problems.

As the experience in the use of Odrette has been accumulated, it has become evident that such a structure of the control system interfers with its further evolution and that the switch-over to the distributed structures is inevitable. At the same time, the creation of a distributed system specifies new requirements for the system software and lead to a certain redistribution of the responsibility for the support of the system functions. So, for example, the programs in different processors should be linked in a unified way, at the level of operating systems, which also arrange a joint access to the common data bases and provides the sinchronization of the processes throughout the network.

The distributed systems may be arranged in a certain hierarchical order. However, the functions of a central computer reduce, in a definite way, to the functions of a file machine which offers the peripheral computers the resources such as disc space, terminals, printers, as well as possibility of access to the archives on magnetic tapes. In this case, the choice of a topology for the network has a lesser influence on the architecture of the system as a whole. So, we have chosen a star-like configuration since it is the simplest in implementation, though the designs with ring and bus structure are possible as well.

The new function, not implemented in the early operating systems is the provision of an interprogram interaction for the programs working in different lower-level processors. In this case, it is desirable to ensure a maximum unification of the program interfaces of a peripheral operating systems for the exchanges of various types: interprogram in a single processor, interprogram for different processors, program exchanges with both the files placed on a disc of a given computer (local files) and the files placed on the discs of a file server.

The current status and the aims

At the present time, the control systems of two big accelerator-storage facilities of the INP, VEPP-2M and VEPP-4, are being updated and, in particular, the switch-over is made to a distributed control system on the basis of a microcomputer Odrette.

The shutdown of the VEPP-4 facility for modernization has enabled us to avoid a gradual transition to the distributed control system and to make it at once, without the stage of coexistence of the old and new systems. At present, the new control system covers injection part of the facility, that comprises a linear accelerator with Gyrocon-based RF system, a synchrotron B-4, beam transfer lines, and an intermediate storage ring VEPP-3. For control, this part of the facility is divided into two, weakly linked subsystems: control of the injector and control of the storage ring.

Now the functions of the computers, forming a star structure, are assigned as follows:

- The central top level computer with 56 Mb winchester disc serves as a file machine and network router server for peripheral processors. There is also a link to Odra-1305 computer (George-3 operating system) providing file archive servises with automatic tape back-up. In total there are 16 Odrettes in the system connected the central one via a serial 3 Mbit/s links. Apart from providing the file servises, the central computer is also responsible for message passing between tasks in different processors of the network, virtual terminals support, mailboxes, etc. The message passing and file exchange mechanisms are implemented in an unified way. There are two classes of exchanges synchronous and asynchronous. The synchronous transaction is equivalent to the mechanism of rendevous.

Eeach peripheral subsystem based around one Odrette may reside in a single or in several CAMAC crates. Additional crates are connected to the one containing processor via a serial 10 Mbit/s driver and a simple non-intelligent controller. The time requied to perform a 24 bit data transaction on the CAMAC bus in local crate (containing the CPU) is about 25 μs; most of this time is spend on parameter passing. In the block transfer mode the rate is 300,000 24-bit transactions/s.

The operating system of the peripheral Odrette provides support of up to 4 local terminals (connected to the local crate) and additionally up to 4 virtual terminals. There may be up to 10 core-resident independent tasks, scheduled on a simple priority basis or on the round-robin basis for the tasks with equal priority.

The Linac is controlled by three microcomputers with the functions assigned as follows: control, monitoring and supervision, beam parameters measurements.

The control equipment is placed in 12 CAMAC crates. At present, there are 260 measurement channels and 230 control ones, and the control channels will be further increased in number.

The applied software of the injector subsystem comprises 5 continuously running programs and some set of auxiliary ones whose execution is initiated, if necessary, by operator or by one of the running programs. An interaction between the control and monitoring equipment is performed by means of two BANK programs working, correspondly, in the control and monitoring microcomputers. The rest of the programs recieve the needed information from them over the inter-program links.

The information, necessary for a facility operator, is output to a terminal or graphic displays. An operator-machine dialogue is carried out mostly via a terminal.

The injector data base consists of two parts: the first includes a description of the elements (address, access rights, scaling factors, etc) and the second contains the table of regimes.

The storage ring VEPP-3 is also controlled by three microcomputers with the following assigned functions:

- control and monitoring of the magnetic system of the storage ring;
- control and monitoring of the RF system;
- measurement of the beam parameters (currens, closed orbit, frequency of betatron oscillations, etc.).

The equipment is placed in 10 CAMAC crates; the number of the control points is 200 and that of monitoring is 500.

Now the applied software includes 8 main programs. The principle of interaction of the programs with the control and monitoring devices is the same as in the injector subsystem. To set the regimes, there exists a program enabling one to change "manually" settings of the elements within a given regime. The program also accomplishes a comparison of the given regime with the current state of the storage ring components.

The general volume of the applied software for the injector subsystem is about 20000 TRAN operators. And its volume grows rapidly as the system evolves.

Since we are not quite experienced in the use of the distributed control systems, we are planning to perform, in the nearest future, a comprehensive analysis of the possibilities and operating characteristics of the systems described above to reveal possible bottlenecks and to outline the ways of their further improvement and development.