The universal Subscriber computing system is a system of technical and mathematical means for the computer to be used in the on-line mode to solve a variety of problems. The fundamental purpose of the system is the solution of problems of large volume in a real time scale with an interrogation response time of a few seconds. Problems of this kind occur most frequently in the simulation of complex processes, the design and construction of control systems, and other such problems. A typical characteristic of these problems is the large volume of calculations in a limited period of time.

The fundamental computing power in the Subscriber system is provided by the BESM-6, together with its software, which includes an operating system, machine oriented language (autocode) translators, and translators for problem oriented languages (FORTRAN, ALGOL, LISP) [2, 3, 6-9, 12, 13].

A subscriber is connected to the BESM-6 through a terminal (teletype) and an input-output buffer. Any computer of average power with an interrupt system can be used to provide the input-output buffer. A control program in this processor controls the two-way flow of data to and from the BESM-6.

The system provides the following facilities for the subscriber:

1. Use of the BESM-6 software: translators for the above languages and a general library of standard programs.
2. Monitoring of the translated program.
3. The ability to organize archives of user programs in programming languages in the backing store of the BESM-6.
4. Computation in the high-speed memory and correction of the text of a program in the user's archive.
5. Organization of a personal library of translated error-free programs in the backing store.
6. The input of data during monitoring and the output of intermediate and final results from the monitoring, diagnostic information, and the text of the program on the printer of the front end processor or of the BESM-6.
7. The facility to terminate monitoring of one program and initiate the solution of another problem.

The following fundamental principles are at the basis of the development of the software for the Subscriber system:

1) maximum use of software already available for the central processor (BESM-6);
2) the provision of facilities for a remote user to use the data processing and computing power of the central processor at the level of available algorithmic languages;
3) modular principles for the construction of additional software to supplement and improve the existing software.

The software for the Subscriber system has a hierarchical structure. The hierarchy of the structure of the software for the Subscriber system is expressed throughout the various program levels.
At the top level is the control program D-68. Based on D-68 is additional software which ensures that the BESM-6 and its peripherals function as a system.

The control system operates through directives specified from a terminal and control commands transmitted along the communication link with the user program. The set of directives and control commands forms the language of the operating system. The operating system language is simple in structure, it is constructed on modular principles, its syntax is close to the syntax of language in which the user communicates with the Project system [3] which was developed at the Institute of Cybernetics, Academy of Sciences of the Ukrainian SSR. The syntactical definitions of the directives and the control commands are the same. We give the syntactical definition of the concept "directive."

\[(\text{directive}) ::= (\text{directive name}) (\text{parameter list})\]
\[(\text{directive name}) ::= (\text{identifier})\]
\[(\text{parameter list}) ::= (\text{empty}) | (\text{parameter}) | (\text{parameter list})\]
\[(\text{parameter}) ::= (\text{number}) | (\text{identifier}) | (\text{text})\]
\[(\text{text}) ::= (\text{any sequence of symbols not containing the symbol*})\]
\[(\text{number}) ::= (\text{decimal number}) | (\text{octal number})\]
\[(\text{identifier}) ::= (\text{any sequence of letters and digits beginning with a letter and possibly containing space symbols})\]

We can use the control directives to carry out the following actions:

1) introduce the subscriber to the system (with an indication of the number of the subscriber) (*LOGIN SUBSCRIBER N);
2) specify the programming system for the job (*FORTRAN, ALGOL, *ITM);
3) indicate data input (*START (computer type));
4) reject the problem being solved (*OUT);
5) indicate termination of the job (*LOGOUT);
6) connect additional program modules with the system (*USE IN SYSTEM (name));
7) reject modules indicated in the directive (*EXCLUDE FROM SYSTEM (name));
8) use certain statistical characteristics of various modules (*EXPLOIT (name)). After completing a task the system outputs information that:

1) the task is complete and the system is ready to accept the next task (*READY);

Fig. 1. Overall functional diagram of the software for the Subscriber system.