EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH
CERN - SPS DIVISION

CERN SPS/84-18 (ACC)

THE MESSAGE ARCHITECTURE OF THE LEP CONTROL SYSTEM

J. Altaber, V. Frammery, C. Gareyte, R. Rausch and P. van der Stok

Paper presented at the 1984 Nuclear Science Symposium,
October 31-November 2, 1984, Orlando.

Geneva, December, 1984.

THE MESSAGE ARCHITECTURE OF THE LEP CONTROL SYSTEM

J. Altaber, V. Frammery, C. Gareyte, R. Rausch and P. van der Stok,
European Organization for Nuclear Research (CERN),
CH - 1211 Geneva 23, Switzerland

Abstract

The LEP control system will be constructed as a global communication system where microprocessors will be used everywhere, from the management of the communication mechanisms, the execution of complex control procedures, and the supervision of the equipment. To achieve this, the global control problem has been cut into sizeable functions which will be encapsulated into microprocessor modules containing enough hardware for the function to be mostly self-contained. This leads to a function architecture where messages are exchanged between the functions on miscellaneous media. It will be shown how these message exchanges can be organized into a uniform flow of data all through the system.

Introduction

The LEP machine is an e^+e^- collider which CERN is constructing on the French/Swiss border. The machine itself is located in a 30 km ring which lies 100 metres below the surface of the ground. Control equipment is housed in galleries which are parallel to the ring and in surface buildings which are grouped on the top of the eight access points.

In 1983 the control system design and implementation was assigned to the controls group which constructed the control system of the SPS, a currently operating proton-antiproton collider and fixed-target machine. This group had to face two conflicting requirements:

- To design in 1983 a control system which will start operation in 1988.
- The design must be compatible with the SPS control system which was conceived in the early seventies.

Fortunately these two requirements can be satisfied due to the architecture of the SPS control system: a distributed network of minicomputers [1]. The global SPS control problem has been cut into functional units of a practical size which can be solved separately. Each of the functional units has been embedded into an autonomous minicomputer. This method leads to the concept of the control system as a communication system, the communicating elements being programs, the information exchanged being messages.

This concept will be further extended for the LEP control system, making use of microcomputers which allow the construction of autonomous entities much smaller than the minicomputers of the 70's. This therefore suggests the construction of a control system where microprocessor-driven equipment are supervised by process control assemblies (PCA) a cluster of microprocessor-based modules playing the role of a former minicomputer, the PCA being interconnected by a network. Methods and protocols similar to those which were used for connecting minicomputers of the SPS control system will be implemented between the PCA's, between the microprocessor module of a PCA as well as between a PCA and the microprocessor-driven equipment. The LEP control

system will use a message architecture between all components. Certainly message architecture is very attractive for the system designer, its modularity and flexibility making his task much easier. Nevertheless, such an architecture suffers from a main defect: communication takes time and we have to construct a real-time control system.

This major difficulty can be alleviated by the combination of three principles.

Firstly, one can make extensive use of the parallel processing capability offered by the architecture:

- between PCA's,
- between the microprocessor module of a PCA,
- between the PCA and the attached microprocessor-driven equipment.

This of course entails defining a programming language which provides the facilities for expressing the parallelism within the control procedures.

Secondly, one can split the control into suitable groups such that much of the communication occurs within a group and a restricted amount between the groups. The communication can be further reduced by making use of local pre-processing and post-processing whenever possible.

Thirdly, time critical synchronization between distributed groups can make use of a dedicated broadcast communication system outside of the message communication system.

The communication between the PCA's

The PCA assignment

Each of the PCA's of the LEP control system will be assigned to a well-defined functional task. As the LEP equipment is spread all along the ring, the PCA assignments will follow the geographical/functional layout.

The control requirements which correspond to an octant of the machine are concentrated in eight sites. In a site, each sub-system such as services, beam monitoring, power conversion, will be given a dedicated PCA. For the central facilities the same rule will be applied, a number of PCA will perform specific duties such as general purpose consoles, alarm analysis,...

Languages

The programming languages for the control procedures must include the facilities necessary for the exploitation of the architecture according to the principles explained above.

Remote-procedure-call is an essential feature to satisfy the requirements, together with the ability of the invoked remote procedure to remit results of its execution to the calling procedure as well as the possibility for the latter to synchronize itself on

the arrival of the results. The remote procedure call is well adapted to express the distributed nature of a control procedure and to exploit the parallelism. Nevertheless, it does not fulfill the ever-increasing requirements of flexibility as in such a feature the remote procedure pre-exists in a fixed manner in the PCA.

Indeed an essential aspect of a machine like LEP is that it will be a day-to-day production plant as well as an experimental device. In effect the behaviour of a machine of this kind is never well-known and it is under continuous development, new equipment being added, new control strategies being experienced. Similar requirements for the SPS accelerator have been satisfied by using an interpretive language called NODAL [2]. The NODAL interpreter has been enhanced by statements allowing a piece of source code, together with the essential parameters, to be exported from the control procedure body to any remote minicomputers, which includes the interpreter.

In addition to its simplicity, this mechanism has two outstanding advantages for the saving of communication resources. Firstly, as the program is sent towards the data, packing and data reduction can be performed in the remote computer thus reducing the amount of data to be remitted. Secondly, as this remote tasking mechanism exists everywhere, it allows a non-hierarchical arrangement where every minicomputer can be a master for a given control procedure while being a slave for another one.

An identical interpreter scheme will be used for the LEP control system. Investigations are currently going on to extend other languages such as Fortran with similar features.

Communication

LEP is a very large machine with its 30 km of tunnel and distance between control sites in the order of 3.5 km. The networking mechanism between the PCA's should make use of a shared medium and should allow long distance.

Fig. 1

The protocol described in the IEEE recommendation 802.5 and usually known as token ring, accommodates itself very well to this requirements. In addition, it provides a deterministic response time as well as good behaviour on heavy loads which are two major advantages for real-time operations. Moreover it offers enough recovery features to be considered as a safe way to transport security information. The LEP network between PCA's will be constructed from two rings: a machine control ring and a service ring which will be fed by a secure mains and which may be duplicated (Fig. 1).

Nowadays, any protocol is only worth the VLSI glue chips which implement it; thus a collaboration has been set-up between CERN and the IBM Corporation which is actively developing a token-ring chips set. Since more than a year, an evaluation has been going on with the joint efforts of the Roeschlikon and Raleigh Laboratories, the results being very positive.

Communication with the equipment

Equipment structure

The LEP machine is composed of many thousands of pieces of equipment. Few of the devices under control are unique; there are, for example, many vacuum pumps, many power supplies, many beam position instruments. The equipment has already been organized into geographical/functional groups under the supervision of a dedicated PCA. This leads to a further level of organization where equipment is made into families and where equipment within a family is identical from the PCA point of view.

The equipment interfaces are grouped into a crate and bus system which operates in the classical address/data mode. In each of these crates there is a single master microprocessor which drives the I/O interfaces. In a crate like this one can have a unique equipment, several pieces of equipment of the same family, or several families of equipment. The crates are connected to the appropriate PCA as mentioned above.

To make the maximum use of the autonomy of the equipment, the local microprocessor will execute regular surveillance of the devices attached, perform systematic acquisition of time consuming ADC parameters, or of table of parameters. It will also survey the fluctuation of an analogic value and perform close loop regulation.

Language

This organization of the equipment naturally leads to a programming method in which variable control procedures are executed in the PCA, calling for the execution of elaborated fixed functions in the equipment crate. Most of the initiative comes from the PCA towards the equipment, in the turn the equipment is able, if necessary, to trigger the execution of a procedure in the PCA (alarm conditions for example).

The programming languages for the procedures in the PCA will have a uniform calling sequence for the equipment. The calling sequence will convey the generic name of the family to which the equipment belongs, its logical/serial number within the family, the desired action and the necessary parameters. An extra level of parallelism will be obtained by adding to the individual equipment call described above, the ability to invoke several equipment within a family in a unique call, the parameters/results being grouped into one array.

Communication

The decisions described above have two points of impact on the equipment's communication design. Firstly, the communication occurs between PCA and the equipment and not directly between equipment; secondly this communication can be structured as messages.

The MIL-1553 protocol [3] satisfies the requirement for a communication of this nature. It offers a multidrop connection which reduces the cabling between PCA and attached crates/equipment and provides a block transfer mode in both directions. The mastership of the communication is in the PCA which can poll the equipment in a simple command/response mode, providing the possibility to request attention from the PCA, together with an implicit surveillance of the medium.

As for as the flexibility to insert/remove equipment which has been explained previously, it will be obtained by adding a gateway in front of equipment communication systems. The control procedure executed in the PCA will call equipment by name rather than by absolute hardware address. The gateway, which contains the directory of the equipment attached to the PCA, will do the necessary routing.

Communication within a PCA

PCA structure

For the definition of the structure of the PCA it is necessary to identify functions which will be given enough hardware to operate mostly in an autonomous manner. The basic hardware construction brick is a microcomputer with its private memory and an interface for communicating with other GPU's. For a function which implies the communication outside a PCA, the GPU can be associated with the necessary hardware interface.

Three main activities have already been identified for a PCA :

- token ring network,
- access to equipment,
- execution of control procedure.

The token ring activity is easy to encapsulate as a function. A data link unit (DLU) made from the association of a GPU with the token ring VLSI will contain all the necessary software for the exchange of messages between PCA's.

A PCA has several multidrop highways under its supervision; each of them will be operated from a highway controller unit (HCU) which consists in a GPU associated with the MIL 1553 VLSI. The HCU will contain the software for a secure transmission of messages in both directions with the equipment. In order to fulfill the requirement of calling equipment by name, an equipment directory unit (EDU) constructed on a GPU will map symbolic name with hardware address and route accordingly the equipment call.

For the execution of control procedures we intend to emulate a multi-programming real-time operating system by the combination of several GPU's .

One component of the operating system is a real-time kernel whose duties are :

- the launching of tasks on request from another task, at scheduled events, on occurrence of an external event,
- retrieval of control on completion of a task,

- abortion of tasks which break the rules of the system.

All this is a well defined function which can be housed into a GPU named the process supervisory unit (PSU).

In a multi-programming system the number of co-resident tasks waiting to use the central processor at any one time may be very large; priority algorithms are used to determine who uses the processor next. The experience with the SPS control system has shown that, for controlling an accelerator, it is possible to contain this problem by organizing the tasks into a limited number of classes, and permitting the concurrent execution of only one task from each class. Typical classes are tasks submitted from the network, scheduled tasks, event driven tasks, local/remote operator intervention.

An arrangement of this kind is ideally suited to our goal: a GPU named a process unit (PRU) will be dedicated to each access point, thus allowing the construction of a multi-processing system. Within this organization a basic PCA will look as shown in Fig. 2.

Fig. 2

Communication

From the above it is clear that the communication between the process activity and the network activity as well as between the process activity and the equipment activity, can be made in a message form. Between the GPU's, which make the process level, the relationships, which are made via monitor calls in a classical multiprogramming system, can equally well take the form of messages. All the communications between the components of a PCA can be organized into messages.

A convenient arrangement for the construction of a PCA is a crate and a multi-master backplane bus [4]: The bus that interconnects the functional GPU's will be used almost exclusively for message exchange between GPU's rather than for program execution as in many classical microcomputer systems. Indeed this obeys the principles quoted earlier as it reduces considerably the flow of data on the common resource, i.e. the bus. There are basically two possible protocols for exchanging messages in a crate and bus system of this kind.

The pass-by-reference protocol makes use of a global memory shared between modules and which contains the message. Message communication between module is simply done by exchanging pointers, the data are not moved in the transaction. This method which is very fast suffers from two major defects. Firstly when more than two modules are involved the software for managing the shared data structure becomes complex without the help of special memory management hardware. Secondly this protocol is very sensitive to failure or misbehaviour of any of the modules.

The pass-by-value protocol copies the message from source module to destination module. This is of course a time consuming process but it is very well adapted to a many modules configuration and it provides a much higher degree of security compared to the pass-by-reference protocol. The communication within the PCA will make use of the pass-by-value protocol, the time penalty being compensated by the reduction of the flow of data.

The uniform message scheme

In the LEP control system messages will be exchanged via three types of protocol:

- token ring protocol between PCA's,
- pass-by-value protocol inside a PCA,
- MIL1553 between PCA and the equipment.

These miscellaneous protocols are used to carry information which are of an identical nature. It is therefore highly desirable to define a uniform way to handle the communication problem all through the system.

The basic reference model for open system interconnection (OSI) provides an ideal framework for implementing this uniform scheme [5]. The OSI model defines seven layers as follows:

7 Application	3 Network
6 Presentation	2 Data link
5 Session	1 Physical
4 Transport	

The communication between two PRU's in two different PCA's can be sketched as follows:

PCA			PCA		
PRU	DLU		DLU	PRU	
(7)				(7)	
(6)				(6)	
(5)	(5)	[5]	(5)	[5]	(5)
(4)	(4)	[4]	[4]	(4)	(4)
(3)	(3)	[3]	[3]	(3)	(3)
(2)	(2)	[2]	[2]	(2)	(2)
(1)	(1)	[1]	[1]	(1)	(1)
PCA bus			Token ring	PCA bus	

{n} is the PCA bus layer n

[n] is the token bus layer n

The PRU contains all the seven layers as it handles the control procedures. The DLU will go up to the transport layer which manipulates messages independently of the fragmentation done at lower layers; it bridges the two media at the session layer where different kinds of services are provided.

Similarly the communication between a PRU and an equipment can be sketched as follows:

Equipment	PCA		
	HCU	EDU	PRU
{7}			(7)
{6}			(6)
{5}			(5)
{4}			(4)
{3}	{3}	(3)	(3)
{2}	{2}	(2)	(2)
{1}	{1}	(1)	(1)
MIL 1553		PCA bus	

{n} is the MIL 1553 highway layer n

The EDU contains up to the network layer where the mapping between symbolic name of the equipment and its hardware address is performed. The HCU will bridge the two communication mechanisms at the data link layer. The equipment may contain the seven layers but will probably end at the transport layer, the network layer being empty.

Conclusion

A method has been described allowing a control problem of a very large machine to be cut into functions of a limited size. These functions are interconnected by the appropriate mechanism and communicate by messages. The application of the OSI model on these messages has provided a uniform scheme for mastering the data flow in the entire system. In addition to its extreme flexibility this architecture exploits in the most efficient way the parallelism offered by such a system which will compensate largely for time overhead which is inherent to any communication scheme.

References

- [1] M.C. Crowley-Milling, The design of the control system for the SPS, CERN 75-20.
- [2] M.C. Crowley-Milling, G. Shering, The NODAL system for the SPS, CERN 78-08.
- [3] MIL STD-1553B-9/21/78, Aircraft internal time division command/response multiplex Data bus.
- [4] J. Altaber, R. Rausch, A multiprocessor bus architecture for the LEP control system, IEEE Trans. on Nucl. Sc., NS-30 (1983).
- [5] ISO/DIS 7498, Information processing system, Open Systems Interconnection, Basic Reference Manual.
- [6] J. Altaber, P. Beck, M.C. Crowley-Milling, R. Rausch, Suggested Principles for the control of future large accelerators, IEEE Trans. on Nucl. Sc., NS-26 (1979).