

## THE MEA COMPUTER CONTROL NETWORK

W. de Vries, L.J. Oostrijk \*).  
National Institute for Nuclear and High Energy Physics  
Nuclear physics section (NIKHEF-K).  
Postbox 4395, 1009 AJ, Amsterdam, The Netherlands  
and

M. de Lange, ACE b.v., Amerbos 13, 1025 XX Amsterdam, The Netherlands

### Summary

A high speed general purpose distributed computer network controls both the MEA linac and its experiments. Fast realtime operation as well as timesharing are supported. A general addressing scheme allows networkwide communication. A distributed database concept is implemented. Control actions operate on a centralised copy of the total accelerator status. Network layout and software approach are outlined, and control operation is described.

Keywords: network distributed database  
message switching synchronisation realtime

### Introduction

The IKO general purpose message switching (mini-) computer network was primarily designed for the control of IKO's new 500 MeV 10% duty factor linear accelerator (MEA). The network offers fast realtime response for the exchange of information between the central accelerator control computers and local machines connected to accelerator equipment. For simple actions like changing a parameter value, and watching its effect, a response time within the range of 0.1 to 0.5 seconds can be obtained, fulfilling ergonomic requirements for human interaction. Providing a high throughput data dump facility for experiments is another objective of the network. There may be several experiment computers running simultaneously, dumping acquired data after compression through the network, to centrally available data storage devices such as disks and tapes. Presently data throughput of over 50 Kwords/sec can be obtained.

The logical network architecture shows a layered structure. From the bare hardware upwards, several layers can be distinguished which may differ from one machine to another, but converge towards a standard network layer called DATANET. This layer features a logical addressing structure for entities (tasks as well as data) throughout the network, independent of physical position. It allows transparent communication between tasks which are distributed over the network, and access to data storage devices interfaced to different machines in the network. At the operating system 'user' level of all network machines the logical structure for addressing entities in the network, and associated task to task communication are available through standardised primitives.

\* ) Currently at CERN, Data Division, Geneva, Suisse.

For easy routing of messages through the network, and for invulnerability to local machine breakdown, a star structured network configuration has been chosen. However, hybrid structures are supported by the addressing mechanisms used. For failsafe purposes, central machines which are critical for the general network operation and for its main applications are available in duplicate.

### Historical Background

In an early stage of the design of the MEA accelerator, it was decided to build a centralised computer control network for the accelerator, without manual control functions for backup. This decision was partly based on tremendous cost of cables and installation along the 250m linac when a more conventional approach would have been followed. More decisive however is the fast response requirement for some local control equipment, which is due to the pulsed nature of the accelerator, and its high repetition rate (2500 Hz). The decision for distributed computer control combined with a centralised control interface, laid the basis for a general purpose realtime network, on which intelligent computer control applications can be built.

Simultaneous construction of accelerator and computer control system seems advantageous as compared to automation at a later stage. However, in the early days of accelerator construction no solid specifications existed as to the type and amount of variables to be controlled, control algorithms, and aspects of man-machine interfacing. Furthermore, absence of manual control options demanded for intermediate delivery of standalone test and control systems, to be changed and expanded continuously.

### Systems Software Development

Lack of specifications enhanced great flexibility in design. First, a fast realtime multi-task executive for the local accelerator control machines was developed from scratch. For portability and ease of programming, a higher level system implementation language was developed, to be used for all systems software. The executive offered a.o. several levels of task synchronisation primitives, and a general task to task communication facility. A similar realtime system was developed for central control and experiment machines. In addition this system is capable of running a powerful timesharing system as one of its tasks. Great emphasis was put on the construction of various tools for program development and testing. In a later stage the network communication system was

implemented, extending the task to task communication, and synchronisation, across machine boundaries. A general network-entity addressing scheme was devised, allowing transparent communication on the user level between a set of tasks running on distributed processors in the network. A central network manager function was added to supervise creation, protection, and removal of sets of communicating tasks. Using the general network facilities a network wide distributed database concept was implemented, providing standardised data representation and access mechanisms. These aspects combined constitute a powerful general purpose distributed processing facility, perfectly suited for its two main applications, and also capable of performing general user oriented time sharing tasks.

### Configuration of the Network

A number of local nodes are situated along the accelerator for local Accelerator Control (ACM). There are several sites where the beam can be guided towards different experiments. Control of magnets (a.o.) in the beam switch yards is done by Beam Control Machines (BCM). At the experiment sites, computers for Experiment Control and data collection (ECM) are installed. One (duplicated) central node performs the store and forward Message Switching function (MSM). As a rule, geographically distributed (local-) nodes have no mass storage devices. The interface to the equipment they control is through the nuclear physics

interfacing standard CAMAC. A number of general purpose machines along with their large peripherals is concentrated in a central computer room. Finally some remote nodes are connected to the network for general user service functions. The local accelerator- and beam switch yard control machines are ALPHA LSI/2 minicomputers (naked mini), while most other machines presently connected to the network are minicomputers of the PDP-11 series: 11/45 for the General Service Machines (GSM), 11/34's for the MSM's and the Experiment Control Machines, and LSI-11/2's for operator-Console Control (CCM) purposes. (See the figure). Communication between all machines is done through Camac based fast serial datalinks, 5Mbaud, full duplex operation.

### Principles of Accelerator Control

Control of the MEA accelerator is based on a centralised database concept. Each local accelerator station controls a number of apparatus of various kinds, represented together in a number of equipment dependent databases. Some database variables represent values that change autonomously and need only to be reported when exceeding a certain limit. Other variables need to be given a value only, possibly in a time dependent way. Each local database has a counterpart at the central site, maintained by a Central Database Administration program (CDA), running in a general service machine (GSMA). Local and central databases inform each other

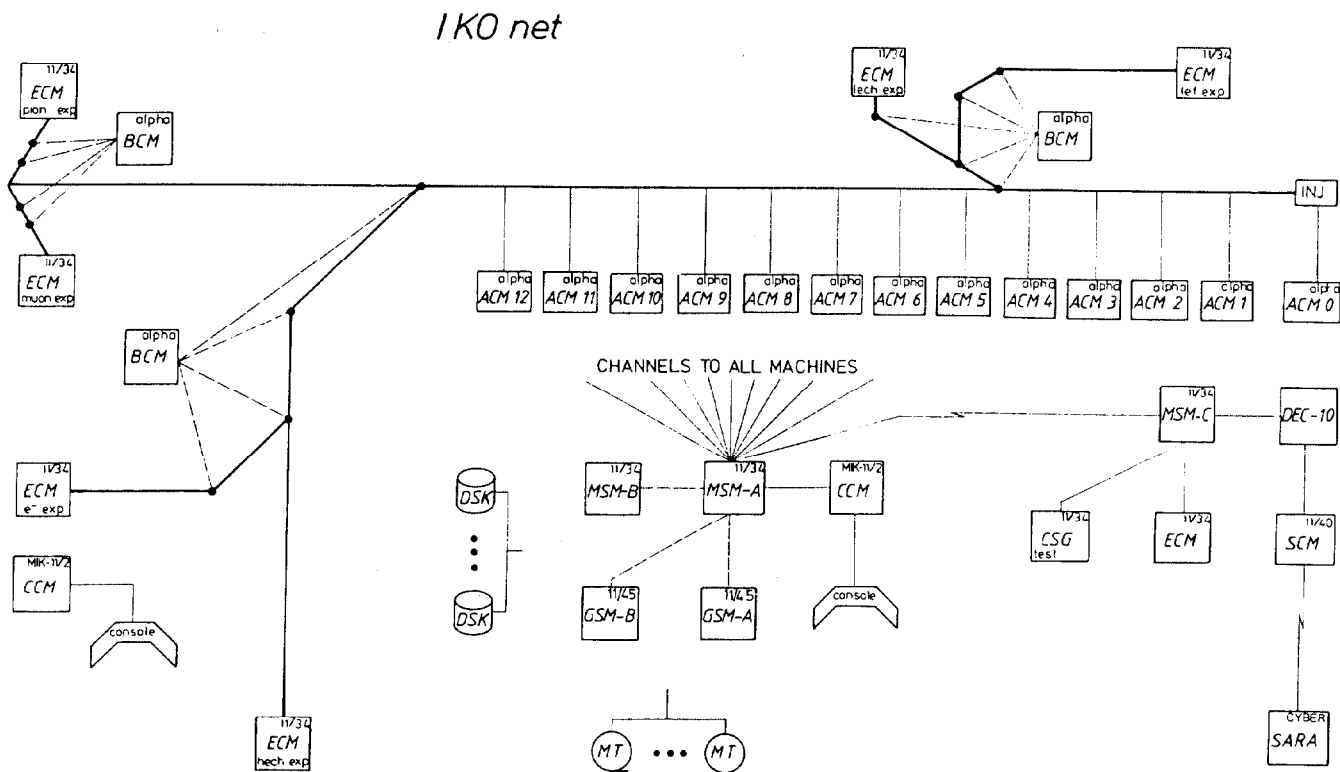


Figure 1  
The IKO accelerator and experiment computer network layout

of changes in their database, so while running, databases are kept in conformance to each other. All local database equivalents combined form a large central database, representing the total state of the accelerator hardware.

Control activities from the operators console affect the central database only. The database administration program reports updates to the local station(s) concerned. Furthermore, it allows variables of the central database to be monitored. Any change in the variable is then reported to the monitoring task, which may be a display process, or an intelligent optimising process.

The central database approach offers the great advantage of having all accelerator parameters (up to 10000) together available in one large database. If a particular beam specification has been obtained, the whole database can be dumped at once, to be restored another time when the particular beam characteristics are to be reproduced. Furthermore, it allows closed loop control at the central site, since all accelerator parameters are easily accessible from a central control program.

### Operator Interaction

Interaction between the central database mechanism presented above and the operator's console functions is for the greater part performed by a user friendly interactive interpreter. The interpreter has especially been developed for this purpose. LISP and macro processor type facilities are offered for programming, and a range of useful features such as logging, gathering operation statistics, save, restore and restart facilities for accelerator settings are provided.

By its structure a command interpreter as well, it provides a powerful tool for empirical try-out of control algorithms. In a simple, straightforward way it can relate database variables to each other, or to controlling devices, such as turn wheels, touchpanel, video display, etc, available at the operator's console. Values and operations, bearing user defined expressive names, can be mixed in an interactive control program.

For example, linking a turn wheel to a variable, and watching the effects of changes on other variables displayed on the video screen, is a straightforward action. Activation of a local accelerator station presently consists of a few simple commands only, yet comprises many closely related, conditional actions. Any level of complexity can be obtained in combining commands to higher order functions.

In a later stage, when more knowledge has been gained about accelerator behavior, control and optimisation programs are foreseen which perform complete accelerator tuning according to a few user supplied specifications such as energy, frequency, experiment target, etc. In that still remote future, the only human actions required are in the field of maintenance, repairing malfunctioning parts, c.q. locating causes of unexplained misbehaviour.

### Future Developments

Interpreters are powerful tools for interactive program development, but also inherently slow. Therefore a compile option should be provided for interpreter programs to speed up performance of

thoroughly tested and established try-out algorithms. A heuristic approach for optimising accelerator startup and beam tuning will be incorporated in the interpreter.

In the design of the pulsed accelerator simultaneous experiments service by means of a pulse to pulse switch of beam specifications has been taken into account. Local equipment must then support a switch of hardware setting within 400 usec. Partly this can be done by fast dedicated (micro-) processors which store a number of beam vectors, but for some hardware the time span certainly is too short (eg. magnets). An intelligent beam defining program must see to it that no impossible state transition can occur, accepting or rejecting a 'beam' requested by the operators.

The operator's interface must be able to distinguish between the several defined, and simultaneously active beams. The central database must be extended to a set of central databases.

### Conclusions

The network as it is presently operational fully satisfies the requirements for accelerator control. Its performance can be improved in several ways, allowing future demands for increasing loads to be satisfied.

The central database approach, combined with the interactive (command-) interpreter has proven to be an ideal combination for easy development of control algorithms. It does not pose any limits on the degree of automation of accelerator control.

Investments in terms of hardware and software development (35 manyears to date) can be considered low, compared to the figures for computer control systems in similar environments. We believe this is due to the extensive use of high level languages, sophisticated software tools, and rigid project management.

### Acknowledgements

The work described in this paper is part of the research program of the Nuclear physics section of the National Institute for Nuclear and High Energy Physics, realized by financial support from the Foundation for Fundamental Research of Matter (F.O.M.), and the Dutch Organisation for the Advancement of Pure Research (Z.W.O.).