Centralized Multiprocessor Control System for the Frascati Storage Rings DAΦNE

Giampiero Di Pirro, Catia Milardi, Mario Serio, Alessandro Stecchi, Luciano Trasatti.
INFN, Laboratori Nazionali di Frascati
P. O. Box 13, 00044 Frascati (Rm) ITALY, e.m. trasatti@irmlnf
and
Barbara Caccia, Vittorio Dante, Raffaellino Lomoro, Eleuterio Spiriti, Stefano Valentini.
Istituto Superiore di Sanita' and INFN, Sezione Sanita',
Via Regina Elena, 299, 00161 Roma, ITALY, e.m. caccia@sanita.infn.it

Abstract

We describe the status of the DANTE (DAΦne New Tools Environment) control system for the new DAΦNE Φ-factory under construction at the Frascati National Laboratories. The system is based on a centralized communication architecture for simplicity and reliability. A central processor unit coordinates all communications between the consoles and the lower level distributed processing power, and continuously updates a central memory that contains the whole machine status. We have developed a system of VME Fiber Optic interfaces allowing very fast point to point communication between distant processors. Macintosh II personal computers are used as consoles. The lower levels are all built using the VME standard.

I. DAΦNE

DAΦNE [1] is a two ring colliding beam Φ-Factory under construction at the Frascati National Laboratories (See Fig. 1).

Construction and commissioning is scheduled for the end of 1995.

The luminosity target is ~10^{33} cm^{-2} sec^{-1}.

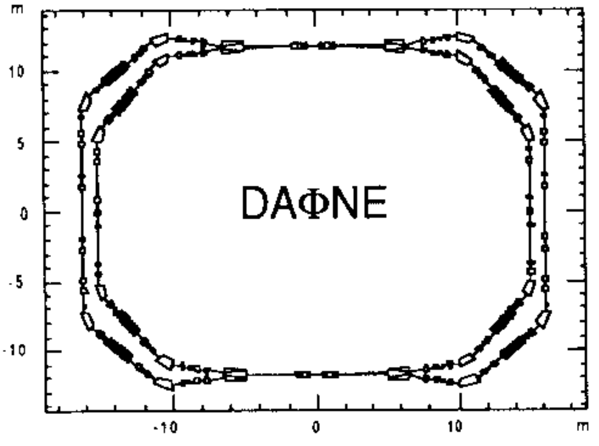

Fig. 1 : The DAΦNE Φ-Factory layout

II. System Structure

Fig. 2 shows the general architecture of the control system. Three levels are defined:

PARADISE (PARAllel DISplay Environment) is the top level, implementing the human interface. Several consoles, built on Macintosh personal computers, communicate with the rest of the system through high speed DMA busses and fiber optic links.

PURGATORY (Primary Unit for Readout and GATing Of Real time Yonder) is the second and central level of the system. It essentially contains only a CPU and a Memory in a VME crate. The CPU acts as a general concentrator and coordinator of messages throughout the system. The central Memory is continuously updated and represents the prototype of the machine database.

HELL (Hardware Environment at the Low Level) is the third level of the system and is constituted by many (about 60) VME crates distributed around the machines.

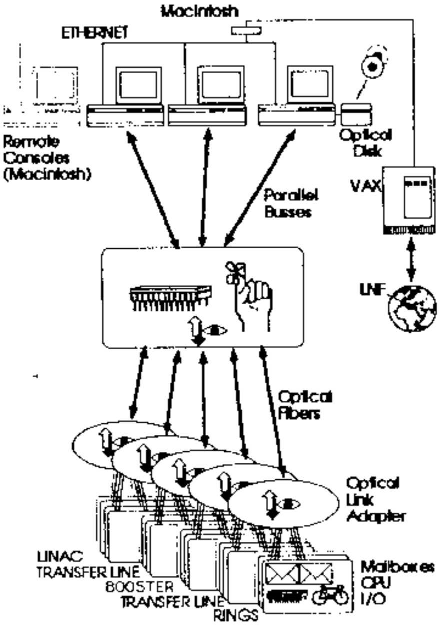

Fig. 2: Control System Schematic Diagram

A CPU in every crate performs control and information hiding from the upper levels.

VME is used throughout Purgatory and Hell

A first estimate of the system gives about 7000 channels to be controlled.

Centralized Communication Control

We have chosen an architecture based on a single central controller of communications instead of the usual network for reliability and performance.

A system with a central CPU controlling all the others through high speed point to point links and a polling mechanism is much easier to implement and to maintain than a network.

Data integrity is easily achieved through a write-readback mechanism and the failure of a peripheral unit can be diagnosed and isolated very efficiently.

Performances are much better than usual networks, due to link speed and protocol simplicity: in a previous paper[3] we measured:

1700 messages/s from Paradise to Hell

10 µs polling time for Purgatory to check out a Hell CPU

450 KByte/s data transfer from Hell to Paradise.

In this architecture the link between the consoles and the central coordinator (Purgatory) is easily implemented through high speed DMA busses. The same is not true for the links between Purgatory and Hell, since the peripheral CPUs are spread over a wide geographic area, well above the standard 70m allowed by DMA busses. The best solution for these links is to use fiber optic connections, with their high bandwidth and noise immunity.

III. OPLA' (OPTICAL LINK ADAPTER)

The OPLA' (OPtical Link Adapter) project for an interface between a standard third level VME crate, and an optical fiber has been developed. This project aims at realizing a multipurpose system for fast data transfer over long distance.

Use of the AMD Taxi chips allows data rates of up to 160 Mbit/s, which is more than a standard CPU can transmit on a VME bus.

A simple architecture will be implemented: a 16 bit word presented to the transmitter will be stored on the other side of the link in a 2048 word FIFO. FIFO overflow will be automatically prevented by back transmission of a FIFO-full status message.

A first prototype board has been developed and tested. The board can be divided into two sections:

i) Tx/Rx toward the optical fiber;

ii) VME interface.

Software for controlling and testing the board has been developed on a Macintosh IIfx using LabVIEW®[6]. LabVIEW® allows to create a front panel that specifies inputs and outputs providing the user interface for interactive operations. Behind the front panel there is a block diagram, which is the executable program.

A panel for the preliminary tests on the OPLA' prototype boards has been built This panel allows access to two VME boards connected through optical fibers.

To access to VMEbus a MICRON (Mac Vee Interface Card Resident On Nubus)[2], developed at CERN and a MacVee (Microcomputer Applied to the Control of VME Electronic Equipment) are used.

The next step of the project will be to implement four complete Tx/Rx sections on a single board. In fig.3 the schematic design of the board is reported. Gate arrays will be used to reduce component count and therefore design time and number of required boards. For each section a Xilinx programmable gate array will implement the control of data transmission and the receive logic. A single gate array will implements the interface toward VME.

The AMD Taxi chips are used to implement the interface toward the optical fiber, due to their ease of connection and high integration.

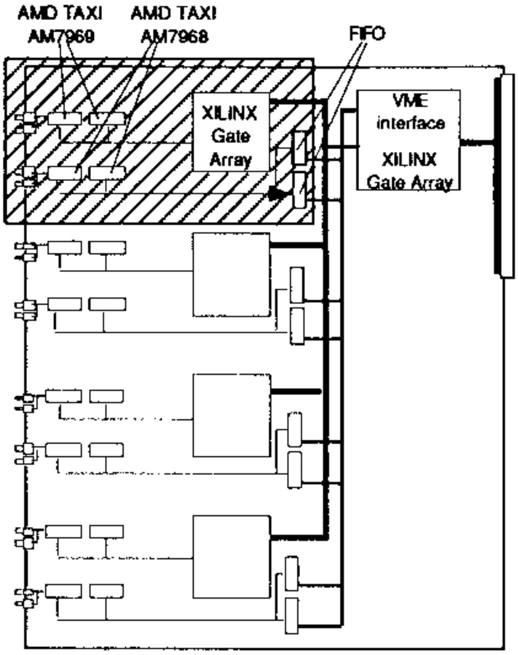

Fig. 3 : OPLA' schematic diagram

IV. CONSOLES

Macintosh personal computers have been chosen for the system consoles. In the last few years we have seen an impressive effort by personal computer firms and third parties to supply large quantities of very high quality software at very low prices.· The situation is now, as far as software is concerned, definitely in favour of the use of large diffusion machines as opposed to high cost, "high" power, low diffusion workstations. Hardware prices keep getting lower, while the cost of software development has reached about 80% of the total cost of an installation, with all the reliability risks of in-house software development. The Macintosh family of computers is at the moment the best candidate for a human interface development, since the effort expanded on software development on this machine has been the most striking on the market.

Previous experience with Hypercard [4,5], on the other hand, has shown that high level software packages can decrease software development times by strong factors. Faster and more powerful human interface packages are coming out every day.

We already mentioned LabVIEW®: it is the first large diffusion software package specifically designed for data acquisition and controls. Its main features are:

- Very easy creation of "virtual instruments", i.e. human interface panels containing controls and displays, acting on the appropriate hardware interface;
- Graphic development of programs through an "icon" language.
- Large scientific library containing frequently used tools (histograms, fourier transforms, etc.).

We are convinced that the use of this kind of large diffusion packages will allow us to save a very large amount of time and effort, not only in program development, but mainly in debugging and maintenance.

Remote Consoles

A good example of the above is the problem of remote consoles. Using an Ethernet or LocalTalk link and a commercial program, Timbuktu® [7] it is possible to gain complete control of a remote Macintosh, under the protection of a system of passwords. This was a specific request for the DAΦNE control system. How long would it have taken to build and debug such a facility?

V. VME OPERATING SYSTEM

In our previous experience with a similarly structured control system we used no operating system for the lower level CPUs. Simple FORTRAN or C programs took care of the relatively easy tasks of a small and dedicated CPU that only has to perform a few simple tasks. The general idea is still:"A CPU for each task". While this is a rather extreme statement, we think that the software environment for the lower level CPUs must be kept as simple as possible, at the expense of increasing their number. On the other hand, the advantages of using a standard environment are obvious as far as bookkeeping and standardization are concerned. At the moment we are evaluating several Real Time Kernels and we plan to reach a decision by the middle of next year.

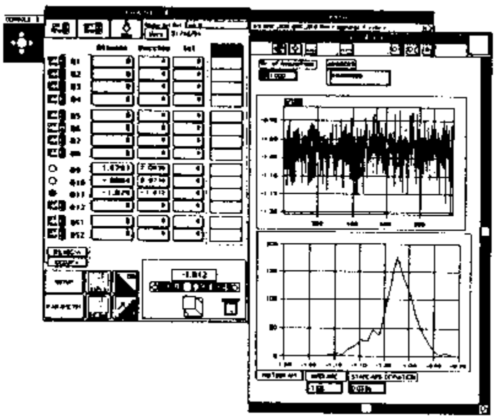

Fig. 4 : Field tests showing progressive migration from Hypercard to LabVIEW

VI. FIELD TESTS

We are testing out these ideas on small accelerators in Frascati. A first implementation on a set of steering coils at ADONE showed the feasibility of Hypercard as a human interface tool, at least for small systems.

Later, on a small machine, LISA, we have tried out the three level system and we have started migrating the human interface, originally written in Hypercard, to LabVIEW (see fig. 4). We have shown that a progressive migration is feasible, and these tests will allow us to measure the real performance of these software packages in the field. At the moment we believe that LabVIEW will prove adequate even for the large DAΦNE control system.

VII. SUMMARY

The control system we are building is based on highly distributed hardware and software capabilities, with a strong accent on openness to other environments. We believe that the human interface will be the most arduous problem to solve, and that the use of high diffusion software packages can be a big help in that direction.

A high speed fiber optic link adapted to the accelerator control environment is being developed.

VIII. ACKNOWLEDGEMENTS

We would like to thank the Accelerator Group of the LNF for continuing discussions and encouragement. The work with the LISA group helped us develop a set of techniques that will be very useful in the future.

VI. REFERENCES

[1] G. Vignola et al., talk presented at the San Francisco Particle Accelerator Conference, May 1991.

[2] B.G. Taylor, "The MacVEE Hardware User Manual"; Rev.4.1, CERN EP Division, 1987.

[3] L. Catani, G. Di Pirro, C. Milardi, L. Trasatti, F. Giacco, F. Serlini "The High Speed Bus System for LISA" Presented at the International Conference on "Accelerator and Large Experimental Physics Control System" Vancouver, Canada, Nov 3, (1989) *In Press to Nuclear Instruments and Methods.*

[4] L. Catani, G. Di Pirro, C. Milardi, A. Stecchi, L. Trasatti.:MARCO - Models of Accelerators and Rings to Commission and Operate, presented at the San Francisco Particle Accelerator Conference, May 1991.

[5] HyperCard v2.1, Apple Computer Inc, © 1987-90.

[6] LabVIEW, National Instruments Corp., © 1990.

[7] Timbuktu, Farallon Computing Inc, © 1987-91.