A NEW LOW LEVEL PROCESSOR FOR THE DAΦNE CONTROL SYSTEM

G. Di Pirro, G. Mazzitelli. I. Sfiligoi, A. Stecchi, INFN-LNF, Frascati (RM), Italy

Abstract

The DAΦNE Control System [1,2] is based on a central shared memory where many distributed processors continuously store information and where user applications fetch the data form.

The shared memory is obtained by merging all the distributed processors RAM into a common VME address space through point-to-point optical interfaces. This structure proved to be extremely flexible during the upgrade of the 2nd and 1st system levels. In such occasion, the multiplatform feature of our software environment allowed to reuse, with only little modifications, all the 2nd and 1st levels software in spite of the drastic hardware change.

Now we are facing a similar challenge: the 3rd level VME processors, based on low-cost Macintosh logic board, become obsolete and we are going to install Pentium Embedded processor with Linux instead. This paper describes the strategy that we have followed to fit different platforms into the existing 3rd level framework always reusing the existing software.

1 A CHOICE OF SIMPLICITY

The DAΦNE Control System is in charge of managing a large amount of devices (Tab. 1) distributed over a wide area.

Table 1: Device List

Device	Quantity	Interface Type
Magnet Power Supplies	425	Serial
Temperature monitor	220	Fieldpoint
Vacuum Pumps	157	Serial
Beam Position Monitors	123	MUX,DVM
Vacuometers	43	Serial
Fluorescent Flags	23	I/O
Ion Clearing Electrodes	39	ADC,DAC,I/O
Beam Loss Monitors	31	Scaler
Kickers	10	ADC,DAC,I/O,GPIB
Video Multiplexers	10	Serial
Scrapers	12	Stepper Motor
Programmable Delays	7	GPIB
RF Cavities	3	ADC,DAC,I/O
Beam Current Monitors	3	DVM

In order to optimize the system load we decide to control with each processor one or few classes of elements. This implies to have a quite large amount of distributed CPU all over the machine area. Having little manpower available, we decided to use VME as hardware platform for its wide diffusion and simplicity of use and LabVIEW as software environment for its friendly and fast development tools.

Also the system architecture has been kept as simple as possible. The front-end processors share data and messages through a common address space without protocols.

2 SYSTEM GENERAL DESCRIPTION

We distribute VME crates all around the machine in order to have the front-end hardware close to the devices to be controlled and to optimizs consist of VME crates containing one or more CPU's with the corresponding device interfaces. The processors have dual port memory for data and messages storage. The crates are linked to a central cluster of VMEs through point-to-point optical links to create a central common address space called 2nd level.

From a data point of view the 3rd level consists of several local memory pages where all the machine objects are represented with continuously updated descriptive records.

The final result is a virtual central memory where the machine RTDB (Real Time Database) resides.

The user level (or 1st level) consists of console applications able to access the 2nd level VME address space. The interaction between the user applications and the machine RTDB is still just a memory access.

3 SYSTEM IMPLEMENTATION

In the first system version we adopted Macintoshes everywhere. The consoles were based on Macintoshes 68040 and the front-end CPUs were low cost Mac 68030 with a custom VME interface.

The connection between the operators' consoles and the VME has beene the number of devices under each CPU. The 3rd level realized by means of dedicated interfaces. The link between the 2nd level and the distributed 3rd level VME crates was made with optical point-to-point connections.

The main peculiarities of the system (see Fig. 1) were:

- "true" memory mapping of the central virtual memory into internal address space of the consoles;
- uniform hardware and OS (Macintosh at any level).

The DAΦNE commissioning started in 1995 and continued until December 1999 with this setup.

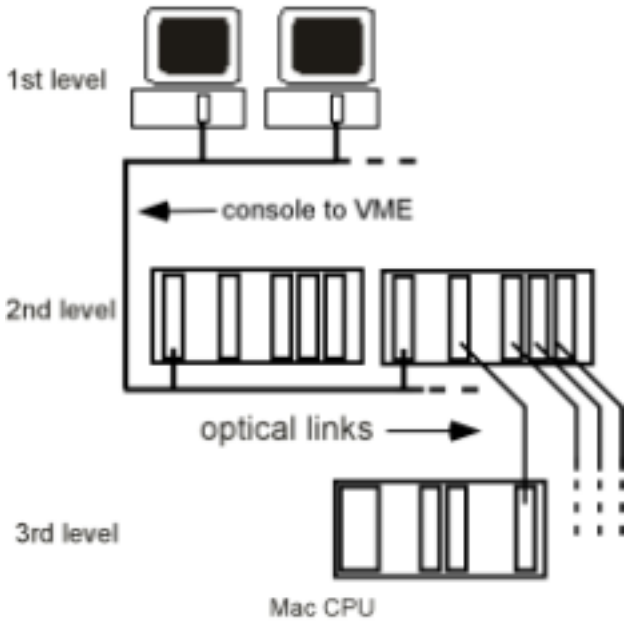

Figure 1: Original System Implementation

At the beginning of 2000 after 5 years of operation we made an upgrade [7] of the console level. In the new version VME Sparc CPU [4] with Sun Solaris [3] as operating system and SunRay thin clients replaced the previous Machintosh. With this new setup we were able to increase the number of consoles and implement new services distributing the information on web and to the experiments [8].

4 THIRD LEVEL UPGRADE.

After the console level upgrade we are now facing the substitution of the front-end processors.

The need to change the Macintosh processors is due to the following main reason:

- lack of spare CPUs;
- new LabVIEW versions no longer running on 68K processors;
- remote development and debugging hard to do on MacOS 7.

We set the following targets:

- to improve 3rd level performance and reliability;
- gain remote access on the front-end CPUs.

An obvious issue is to reuse as much as possible the software already developed and to leave the front-end acquisition bus unchanged. These two reasons led to adopt VME computers able to run LabVIEW.

We have chosen Pentium VME embedded computers by VMIC [6] with Linux Operating System.

In a first phase we will have simultaneously the new and old processors in the control system. To obtain a smooth insertion of the new processor in the control system we have rewritten the basic read/write VME and RTDB access routines in the new operating system keeping the same data structure. After this, the porting of all the 3rd level applications in LabVIEW for Linux will require only a small tune-up.

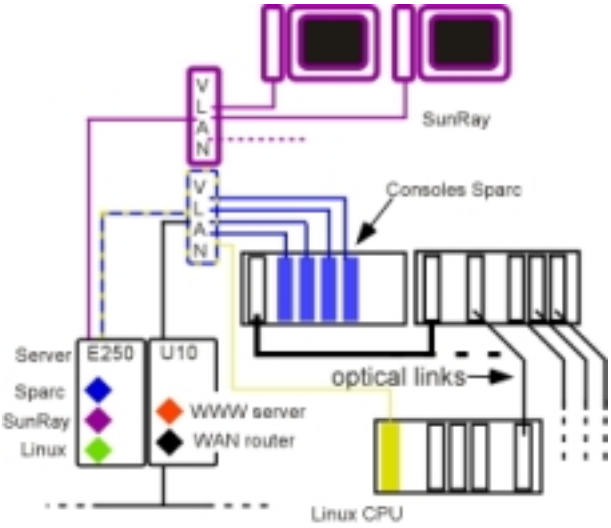

Figure 2: New Implementation of the Control System.

5 STATUS

We use the VMIC in a diskless configuration. This allows us to avoid many CPUs distributed around the machine and we will boot the 3rd level Linux processor from the same Sun server used by the Sparc consoles.

The VME access routines have been tested and a first Linux CPU is running “on the bench” 3rd level application coming from the old platform.

We plan to install the new processor for the control of the new devices starting from 2003.

Concluding, the choices of the shared memory connection and of multiplatforms software allow us to make changes in the control system hardware with little worries.

6 ACKNOWLEDGEMENTS

We want to thank G. Baldini and M. Masciarelli for their commitment and essential contribution in the system implementation.

We are also grateful to all the Accelerator Division staff for their continuous suggestions and encouragement for making a good and useful job.

REFERENCES

[1] G. Vignola and DAΦNE Project Team, DAΦNE: The First Φ-Factory, EPAC'96, Sitges, June 1996, p. 22.

[2] G. Di Pirro et al. "DANTE: Control System for DAΦNE based on Macintosh and LabView", Nuclear Instrument and Methods in Physics Research A 352 (1994) 455-475.

[3] LabVIEW, National Instruments Corporation, 11500 N Mopac Expwy, Austin, TX 78759-3504 USA (http://www.ni.com)

[4] Force Computers GmbH, Prof.-Messerschmitt-Str. 1, D-85579 Neubiberg/München (http://www.forcecomputers.com)

[5] Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, CA 94303 USA (http://www.sun.com)

[6] VMIC, 12090 S. Memorial Pkwy, Huntsville, AL 35803 USA (http://www.vmic.com)

[7] G. Di Pirro et al., The Evolution of the DAΦNE Control System: A History of Liberation from Hardware, ICALEPCS 2001, San Jose California, November 2001, WEAT003, (http://www.slac.stanford.edu/econf/C011127/proceedings.html).

[8] G. Mazzitelli et al., The KLOE/DAPHNE Status Logging, Analysis and Database System, ICALEPCS 2001, San Jose California, November 2001, WEDT002 (http://www.slac.stanford.edu/econf/C011127/proceedings.html).