TUAP003
hep-ex/0111076

PRESENT STATUS OF VEPP-5 CONTROL SYSTEM

D.Yu.Bolkhovityanov, R.G.Gromov, E.A.Gousev, K.V.Gubin, I.L.Pivovarov, O.Yu.Tokarev
BINP, Novosibirsk, Russia

Abstract

This report concerns the present status of VEPP-5 control system. The control system hardware consists of CAMAC blocks, a set of crate controllers based on INMOS transputers and ICL-1900 architecture processor Odrenok, and Pentium-based workstations. For small tasks simple serial CAMAC controllers are used. For slow controls of power supplies the CANBUS is begun being used. The workstations are running Linux and are connected via local net using TCP/IP. Odrenok crate controllers are joined into other local net and are used for control of equipment in high voltage pulse condition (klystron gallery). Transputer crate controllers are linked directly to the server computer and are used for high performance diagnostics (BPM). The three-level software complies the so-called "standard model".

1 INTRODUCTION

1.1 General design

The VEPP-5 forinjector is a large installation that includes: DC-gun, klystron gallery, power supply system, bunch compression system, thermo system, BPM and others [1]. Operation is pulsed with repetition rate from 1 to 50 Hz. The control system of the VEPP-5 injection complex has a standard three-level model[2] (fig. 1).

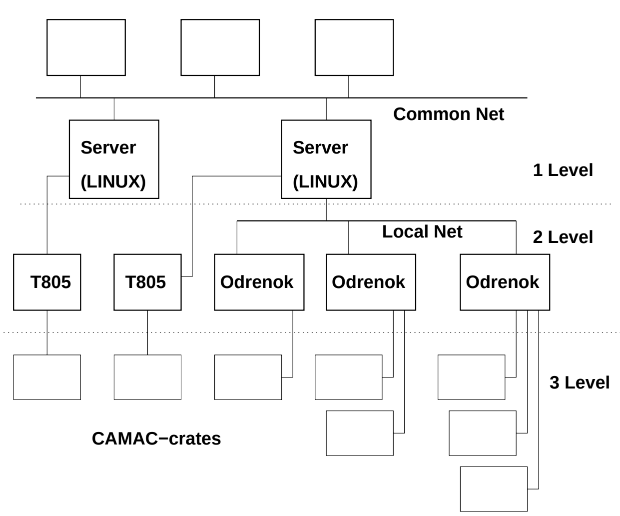

Figure 1: Control system network.

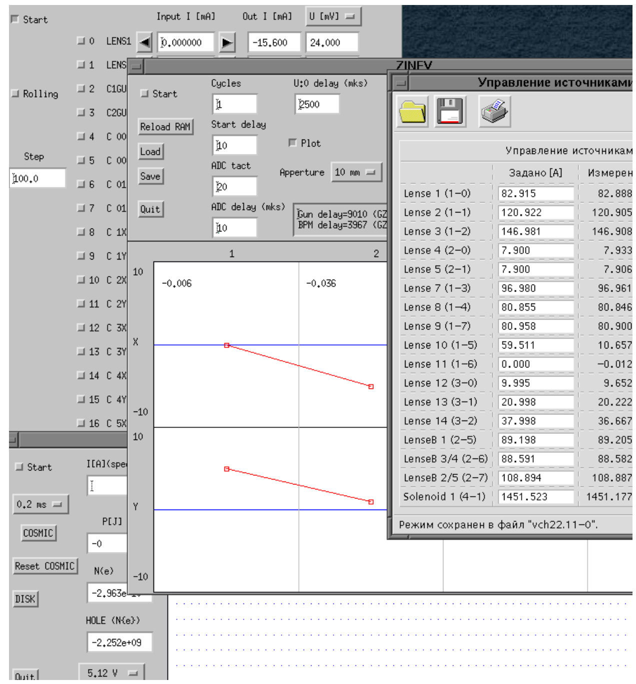

Figure 2: Several generations of high-level control software.

The lowest level is composed mainly from CAMAC electronics. The second level is a set of CAMAC crate controllers with supplementary low-level software. There are several types of intelligent crate controllers based on INMOS T805 transputer, ICL-1900 Odrenok and Motorola 5200.

The high level consists of a server which runs under Linux (RedHat-7.1). All client programs have access to the low level only via the server, which performs initial loading and initialization of the crate controllers and supplementary programs.

1.2 Software design

Historically there have been several generations of control system software on VEPP-5. The very first programs were simple – they implemented both client interface and hardware access. This approach is still used in some tasks[1].

However, we quickly switched to three-level architecture, because it is much more suitable for control tasks. First ver-

[1] A Russian proverb says: there's nothing more constant than temporary.

sions were bound to specific controllers – separate software for Odrenok and T805. They are still in use, but a general version was developed which is able to serve various types of controllers simultaneously (see Fig.3). The design of the general version was greatly influenced by design principles of X11.

The new version uses the same mechanisms (which are dictated by the nature of the controllers) as previous versions, described in sections 2 and 3.

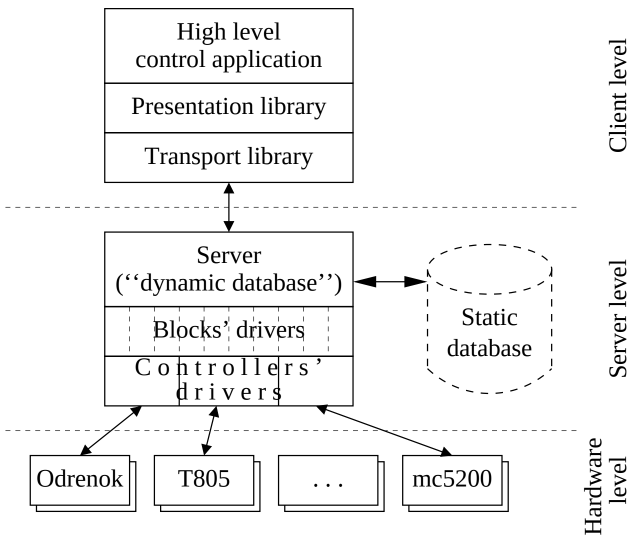

Figure 3: Structure of the current clients ↔ server ↔ hardware system.

High-level programs use Motif for the client interface. A special library was designed, which builds control windows "on the fly" from a database description, thus significantly reducing application creation time & costs. (An example of such a window can be found at the right of Fig.2.)

2 ODRENOK ↔ SERVER COMMUNICATION

Odrenok is the most popular crate controller in BINP. It has the ICL-1900 instruction set supplemented with commands for CAMAC bus access and vector operations. The new CAMAC adapter was developed to have a possibility to connect CAMAC with PC via Ethernet. The CAMAC-Ethernet adapter provides a speed of data exchange up to 400kB/sec [3]. It is sufficient for real-time network operation. The server workstation has two Ethernet cards: one is for communication to the institute network and another is for the local Odrenok net.

The ODOS (ODrenok Operation System) protocols use Ethernet packets of non-standard type, therefore the server requires I/O facilities for these packets. It is implemented by a kernel module which provides sending, receiving and waiting functions via a standard socket interface; the module is supplemented by client libraries and utilities (Fig. 4).

However, usage of home-made protocol have shown many inconveniences. So, this year UDP support was added

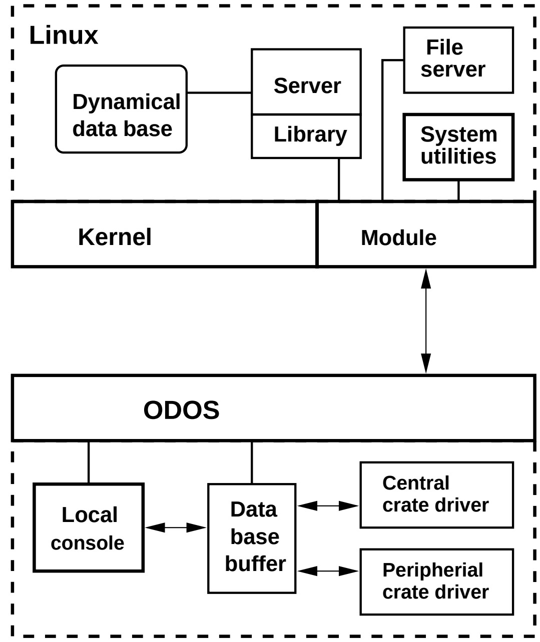

Figure 4: Linux ↔ ODOS Software structure.

to ODOS, which enables to use standard Unix communication interface and solves many other problems (routing between networks, kernel upgrades, stability).

3 TRANSPUTER ↔ SERVER COMMUNICATION

Crate controllers based on INMOS T805 Transputer [4] have been developed in BINP as high performance devices for data acquisition systems. This type of controllers is a suitable tool for wired BPM and pick-up electronic control. This is because T805 has a powerful floating point unit. In this case all calculations are performed on the 2nd level and the calculated data are sent into the 1st level software.

Transputer controllers have no OS (despite the fact that transputers are excellent for parallel tasks), so the following set programs had to be made (See Fig. 5). First, a dispatcher performs communication between the local transputer net and a set of block drivers. Second, an event handler ("PINT") which enables drivers to read data when it is ready. Finally, a set of drivers for individual CAMAC blocks (1 driver process per block).

4 MOTOROLA CRATE CONTROLLER

The Odrenok crate controllers are very old. Production of T805-based controllers had ceased. So, we have to find another crate controller.

This year we began using another one, based on Motorola 5200 processor. 100MB Ethernet is used for communication to host computers.

This controller runs uClinux [5] – a Linux clone designed for processors without a Memory Management Unit

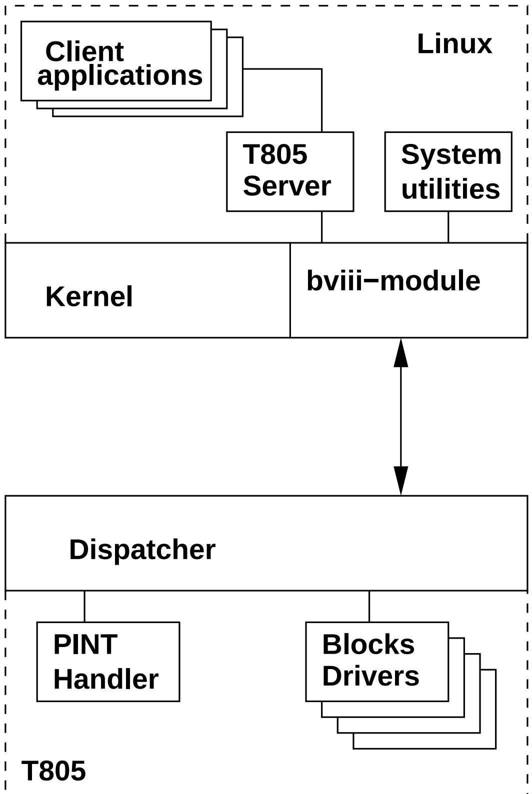

Figure 5: Linux ↔ T805 Software structure.

(MMU). Having Linux in both workstations and crate controllers significantly eases the life. On the other hand, lack of MMU makes multitasking and multithreading too tricky.

So, using ARM, PowerPC or an x86 clone in CAMAC controller could be much better choice, but 5200 was chosen mainly because of abilities of BINP electronic design department.

5 FUTURE DEVELOPMENT

Currently the information about hardware structure and knowledge of how it is mapped to "physics" information is spread along the various parts of software – in the server config files, hardcoded into application programs, etc. This is extremely inconvenient and error-prone, so we switched a significant part of manpower to design a database, which will contain all this information (a so-called "static database"). The database is based on PostgreSQL, but for more flexibility all pieces of the control software will access it through the server.

In the last several years VEPP-5 began to use other hardware in addition to CAMAC. In this process we tried to employ the most standard interface as possible. The ultimate goal is to replace all custom-made PC↔hardware communication boards with standard ones, such as Ethernet.

For slow controls of power supplies CANBUS devices are used. However, this decision is still half-CAMAC-based – the CANBUS controller is itself a CAMAC block. This was done because it was impossible to find PCI CANBUS controllers with open specifications. The CANBUS hardware fits nicely into our software architecture, so we'll widen its use.

Some devices (like TV cameras) require very high bandwidth, which can't be obtained from CAMAC. So, our lab designs TV camera with 100MB Ethernet interface. Having negative experience with non-standard communication protocols, we decided to implement transport protocol over UDP.

Since Ethernet chipsets are very cheap, even not-so-demanding hardware is being implemented with Ethernet as communication media. Thus Ethernet extends its presence as one of control system's low-level buses. But various types of hardware (crate controllers, TV cameras, slow devices) will be physically connected to separate buses.

Currently the VEPP-5 control system is being designed and tested on the forinjector, and in the future it will also be used on damping ring.

6 REFERENCES

[1] A.V.Aleksandrov et al., "Test of electron Linac for VEPP-5 Preinjector", Proc. EPAC'2000, Vienna, 2000, http://accelconf.web.cern.ch/accelconf/e00/PAPER/MOP4A05.pdf

[2] Götz et al., "Experience with a standard model'91 based control system at the ESRF", Proc. ICALEPS'93 (Berlin), NIM 352A (1994) p. 22-27

[3] A.N.Aleshaev et al., "VEPP-5 clystron gallery control system", Proc. PAC'99, New York, 1999, http://bnlinfo2.bnl.gov/cgi-bin/pac99/displaypaper.cgi?MOP41

[4] INMOS Limited, "Transputer Instruction Set", Prentice Hall, 1988, ISBN 0-13-929100-8.

[5] Lineo, Inc., "Embedded Linux/Microcontroller Project", http://www.uclinux.org/