# AN EPICS IOC ON PC104

M.Giacchini[1], G.Bassato[1], F.Gramegna[1], R.Lange[2]
[1]INFN, Legnaro, Italy, [2]BESSY, Berlin,Germany

## ABSTRACT

The control system of the NUCL-EX[1] experiment at LNL has been upgraded by adding a 600 channel analog matrix to route signals from Constant Fraction Discriminators to the data acquisition room, allowing a complete check of signals during the experiment's initial set up as well as monitoring the signals during the experiment itself. The matrix is implemented on a Camac crate equipped with home-made multiplexer boards. The Camac controller is a Highland Technology model M250 with a Ethernet 10BaseT interface. The control of this device has been developed using Epics (Experimental Physics and Industrial Control System)[2] tools; to minimize the cost of this realization the EPICS IOC has been ported to a PC104 target running under Linux (Slackware rel. 8.1). The PC104 board is a TMZ104 model by Tri-M System Engineering; it is based on a Crusoe TM5500 processor running at 500 MHz. The board is specifically designed for embedded control applications: it does not require cooling fans nor external hard disks to operate. A solid state 48 MB DOC (Disk on Chip) is large enough to store the Linux OS, together with the EPICS IOC core and the application code. A new EPICS driver for the M250 Camac Controller has been developed using the Asyn driver model: this approach makes it highly portable through many types of physical interfaces. Using the same model, an EPICS driver was written also for an Allen Bradley Micrologix 1500 PLC that is used to monitor the average temperature of detectors electronics.

## INTRODUCTION

When an experiment involves a high number of detectors, the capability of remotely testing the signals coming from the analog front-end becomes crucial to assure a correct experiment setup. For the NUCL-EX experiment we have realized an analog matrix based on Camac multiplexer boards: the crate controller is a Highland Technology model M250 with an Ethernet 10BaseT interface. To develop the control software it was decided to use EPICS, to benefit from its open architecture and from the wide availability of device drivers and debugging tools. Last, but no less important, EPICS licence is free to research institutions. Cost containment was one of the constraints of this project, so dedicating a VME system with VxWorks to run the IOC was not viable, even though we could have taken advantage of previous experiences with this architecture. As an alternative, it looked promising and exciting to exploit the features of the latest release of EPICS core (R3.14.7) and use the new Asynchronous driver model to develop the support for the M250 crate controller under Linux. Since the M250 interface is an Ethernet port, the IOC can reside on any node of LNL network: so, to further minimize the cost, an inexpensive PC104 board was selected to run the IOC on. Linux Slackware was chosen as OS for the PC104[3] target, while Linux Fedora 2 was used, on a standard PC, to run the graphical interface. The paper describes how the asynchronous driver operates and the steps we followed to develop the device support; moreover, it describes how the operating system has been tailored to fit into the disk-on-chip of the target board. Finally, some results are presented to illustrate the performance of the adopted solution.

## IOC HARDWARE

If the number of I/O channels is limited, an embedded PC104 can be used to run an EPICS IOC efficiently at a cost that is more than one order of magnitude lower than a system based on a VME crate. When choosing a card and a processor, a number of characteristics are important: first of all, the board should support a Disk on Chip device to boot the OS and the application code: avoiding to use a mechanical drive reduces the system size and improves the reliability. Then, the processor should be selected by applying the criteria of a right compromise between computational performance and dissipated power. If the dissipated power is low enough (limited to two or three watts), the board can operate without cooling fans which, again, contributes to improve the overall reliability. It should be noted that the ratio performance/diss.power doesn't depend only on the CPU clock, but also on its internal architecture. For example, the processors produced by Transmeta Inc. take advantage of an innovative architecture (VLIW, Very Long Instruction Word) and execute the binary X86 code using much less power with respect their competitors. The choice of peripherals is also important: many boards include the Ethernet controller as well as the VGA controller. While the Ethernet chip is essential in a network distributed systems, the VGA is often useless and should not be included to reduce cost and power requirement.

Keeping in mind the above considerations, a TMZ104 board from Tri-M System was chosen as controller. Additionally, a Utility Board (by Tri-M) and a Fabiatech FB1603B Ethernet 100BaseT board are used.

The resulting embedded system has the following features:

- Transmeta Crusoe TM 5500 CPU.
- Ethernet 100BaseT (Fabiatech Card).
- Dual RS232C serial.
- Embedded BIOS.
- Watchdog timers.
- Dual EIDE and floppy support.
- 128 MB of RAM.
- 48 MB DiskOnChip
- USB, parallel port
- PS2 keyboard & mouse connectors

The operating system for the IOC is a Linux-MZ[4]. The Linux image is based on a Slackware 8.1 distribution with a 2.4.18 kernel. The root file system uses about 14 MB and is loaded into RAM at start up. The system uses 6.8 MB of the DiskOnChip (DOC).

## IOC SOFTWARE

EPICS device drivers were written to control both the Camac crate and the PLC.

The environment to develop and deploy the application is based on Linux Slackware 8.1 OS with a Kernel 2.4.18. We used EPICS 3.14.7 core and the new Asyn 4.1 (Asynchronous Driver) model to develop the support for our devices. Asyn provides a standard mechanism for higher level EPICS software, such as Device Support and State Notation Language, to communicate with device-specific drivers. Facilities provided by Asyn include connection management, separate threads for asynchronous devices, trace and debugging control, standard interfaces and generic Device Support. The available Driver Support modules from the EPICS community are only supporting high level classes of Allen-Bradley PLCs and cannot be used with the Micrologix 1500. To interface to this device only two types of commands are possible: Unprotected Read and Unprotected Write. Using these two commands, the data exchange with the PLC has been achieved using an integer file type identified as N7. The M250 Camac Crate controller uses a TCP ‘streaming’ (not block-oriented) communications protocol, which is a human readable ASCII stream. Each ‘connection’ is a single, independent bidirectional data stream using a fixed command set. Through these connections every type of command can be issued, using the Ethernet communication like an RS232C terminal connected to the module.

The database realized is the result of the inclusion of three databases: m250control.db, mx1500.db, teststand.db (camac modules functions); these realize an application with 70 records described by table 1 below:

| **Nr. of Records** | **Scan (sec.)** | **Type** |
|---|---|---|
| 60 (teststand-multiplexers) | 0.1 | Longin-Longout-MBBO |
| 3 (m250) | 2 | Stringin |
| 8 (mx1500) | 1 | Longin |

Table 1: Databases

The EPICS databases are created and edited using standard tools from the EPICS toolkit (VDCT, MSI). Porting the application from the development environment to the PC104 was done taking into consideration the limited space available in the DOC. For this reason, we transferred to the target only the binary code of the application and the caRepeater module, required to manage Channel Access. To be able to move the binaries without recompiling everything on the PC104, the STATIC_BUILD switches were applied when building the EPICS base and the application. With these flags active, the binary image includes statically all EPICS libraries. Tailoring the Linux OS for the PC104 was not required since we used the ready-to-use image in the DOC provided by the manufacturer. Only few shared libraries (libpthread, libreadline, etc.) had to be added[4]. The device driver for the Fabiatech Ethernet Card was compiled like a loadable Linux module.

## OPERATOR INTERFACE

Traditionally, EPICS Operator Interface (OPI) clients were based on the Unix/X Window System; Sun workstations were often used to run the graphical interface.
Recently, most standard EPICS tools (such as the MEDM display manager) were made available for multiple platforms. Thus, PC workstations running Linux are now widely used as OPI machines for EPICS. With the EPICS Extensions for WIN32, many tools are available also for Windows based PCs: this allowed to further reduce the cost by reusing Windows machines already installed for the control of NUCL-EX experiment to run the graphic interface.
Fig. 1 shows the overall network structure for this application.

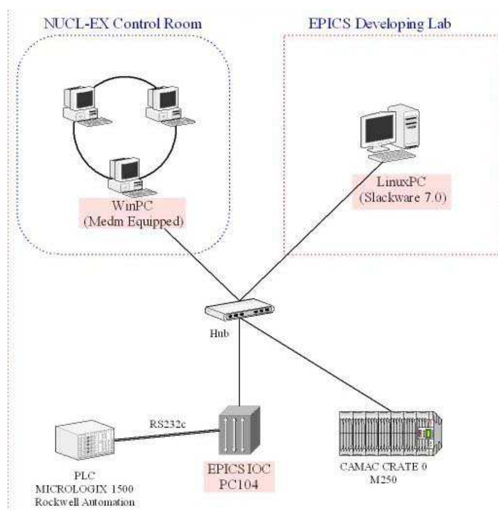


Figure 1: Structure of network used to develop and use this application.

## STATUS

A tiny IOC control system has been developed to set up and monitor a number of Camac multiplexer modules and to remotely acquire data from temperature sensors by means of a Micrologix PLC. A prototype module is currently being used in the experimental area of NUCL-EX.

## CONCLUSION

The use of PCs in the control of experiments is getting more and more popular. The choice of a collaborative, free, open source software lined up with state-of-art technology, allowed upgrading the control system of an experiment with limited budget and manpower resources. The core components of EPICS are now running on inexpensive hardware: the continuous commitment by the developers' community to extend its portability has significantly reduced the cost of deploying EPICS systems. The availability of COTS extremely compact controller boards is changing the scenario of control architectures towards high granularity systems based on a large number of small yet powerful network nodes. The PC104 is one of the industry standards that made this evolution possible.

## ACKNOWLEDGMENTS

Thanks for all the help we received from the EPICS community.

## REFERENCES

[1] F. Gramegna, http://www.bo.infn.it/nucl-ex/4pages.pdf
[2] A Epics Collaboration, http://www.aps.anl.gov/epics
[3] PC104 Consuntorium, http://www.pc104.org
[4] M. Giacchini, http://www.aps.anl.gov/epics/wiki/index.php/HowToPC104