10. OTHER ACTIVITIES ON CONTROL DATA ACQUISITION AND SIGNAL PROCESSING

J. Sousa and C. Varandas (Heads), A. Batista, A. Combo, C. Correia, M. Correia, N. Cruz, R. Pereira, P. Ricardo.

10.1. INTRODUCTION

This project included in 2004 two main research lines:

- Development of a low-cost, fully integrated, event-driven real-time control and data acquisition system for fusion experiments;
- Development of clusters of computers.
  This year adequate embedded development tools for the PowerPC processor have been assessed. A preliminary hardware control core of the system, implemented as a System on Chip (SoC), has been designed. Adequate software tools for programming a reconfigurable real-time processing system have been assessed and tested. A reconfigurable hardware PCI module has been developed. A water-cooled 16 computer cluster has been developed, tested and used for heavy parallel calculus.

10.2. Embedded development tools for Power PC

The performance and cost of the following embedded real-time operating system for PowerPC and cross-compiling tools have been evaluated: WindRiver VxWorks, MontaVista LinuxPPC and LinuxPPC with RTAI extensions. Results allow to conclude that the LinuxPPC with RTAI is the best choice since it has adequate real-time capabilities at a much lower cost. The Xilinx Platform Studio cross-compiling tools are appropriate due to the use of programmable logic devices from this manufacturer for SoC implementation. These tools have been tested to readily compile a version of the Linux kernel that has been specially developed for the target hardware platform.

10.3. Hardware control core of SoC

A basic hardware control core of the system on chip (SoC) has been designed, using the Xilinx Platform Studio (Figure 10.1). A version with only one processor has been successfully tested and implemented on a ML310 board which includes a Xilinx Virtex-II Pro FPGA.

10.4. Software tools for programming a reconfigurable real-time processing system

The reconfiguration of a SoC-based control and data acquisition system can be made at two levels: the processor and the programmable logic. While the former is readily programmable using a computer language compiler, the latter until now was lacking high level programming tools. Nevertheless there are at least two suitable tools for this task: the Mathworks Matlab (with the Xilinx System Generator or AccelChip) and the Mentor Graphics Catapult C Synthesis. This group of tools covers three different, complementary approaches for reconfiguring the programmable logic: integer and floating-point programming in Matlab and high-level language (C) programming.

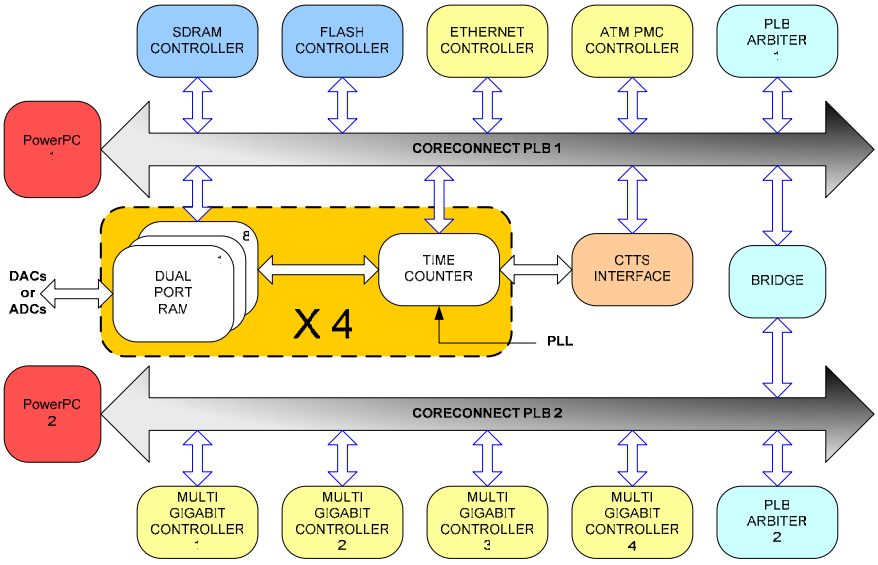

Figure 10.1 - Hardware control core of the SoC system

10.5. Reconfigurable hardware PCI module

A general-purpose multi-channel PCI module for applications on control and data acquisition has been developed. This module includes a reconfigurable DSP/FPGA combination providing up to 8000 MIPS DSP and 3 Million logic gates and can be coupled through a standard connector to new expansion boards.

For new projects the designer will target the application, by designing a specific expansion board and reconfiguring the FPGA equations and/or the DSP code.

The first application was an expansion board (Figure 10.2) with 8 channels, 2 MSPS, 16-bit waveform generator for use on the test-bench platform for the real-time project of JET. A control board with 12 ADC channels with 12-bit@50 MHz and 2 DAC channels as well as a timing generator and event router module is being developed.Generic DSP control software including a library of basic functions and a library of FPGA equations has been developed to facilitate the design of codes for the specific operation of new expansion modules and applied on the development of the waveform generator module.

10.6. Development of clusters of computers

CFN is developing clusters of standard computers (Orionte) (Figure 10.3), with the following main design requirements: (i) reduced cost; (ii) reduced space; (iii) reduced noise; (iv) high ratio between performance and cost; and (v) adequate parallel programming skills for plasma physics.

Orionte has been implemented in two editions with 8 and 16 processors, using exclusively commodity hardware (Table 10.1). In order to get the minimum cost and space, Orionte is made by modules of eight nodes. Each node is put together with one set of components except the switch, which serves up to three modules (Figure 10.4).

Employing water cooling copper blocks, manufactured at IST, it was possible to put eight nodes is one 6U rack. T 1GBit/s ethernet was chosen in order to satisfy network demanding parallel programs. Each module (8 boards) is equipped with one harddrive (120 Gb for Orionte 16), for 7 nodes boot remotely with DHCP and TFTP and mount the master's harddrive through NFS.

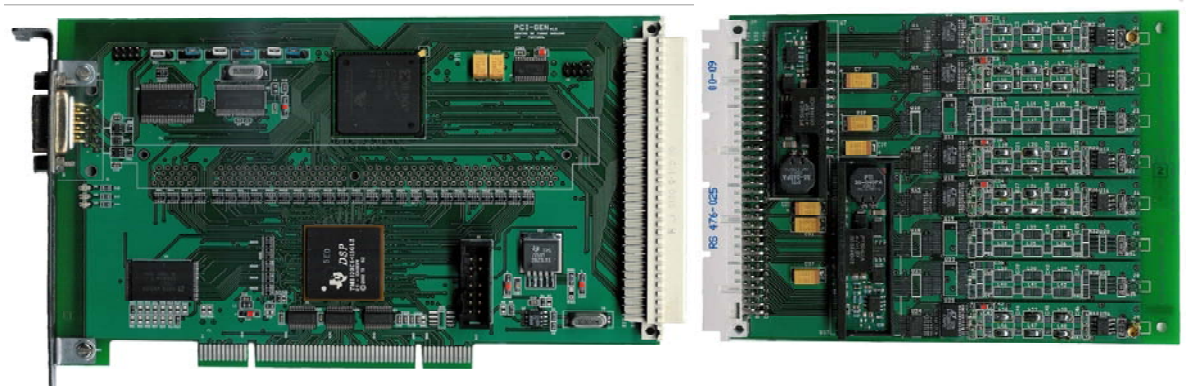

Figure 10.2 - Configurable hardware PCI module with a waveform generator expansion module.

	Orionte 8	Orionte 16
Processor	P4 2,4GHz FSB 533MHz	P4 3 GHz FSB 800MHz
Memory	1GByte (2 x 512Mb)	1GByte (2 x 512Mb)
Network interface	1Gb (onboard)	1Gb (onboard)
Motherboard	Intel D845GERG2	Intel D865GLC
Hard drive	80Gb	2 x 120 Gb
Switch	Ovislink 1Git 8 ports	USRobotics 1Gbit 24 ports

Table 10.1 – Components of Orionte 8 and 16

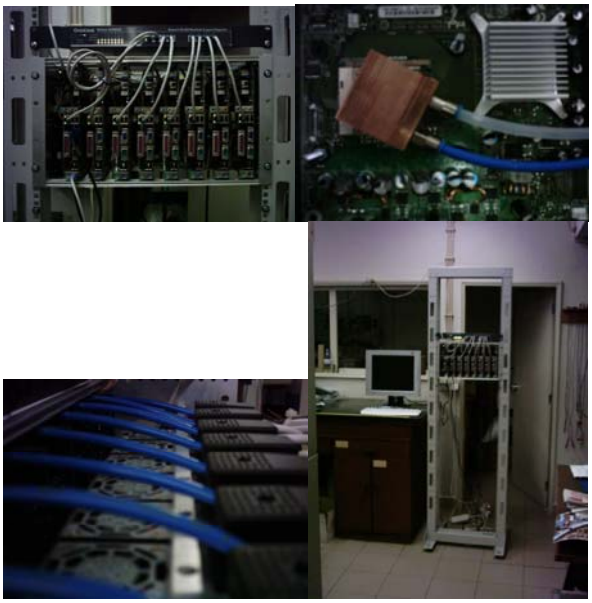

Figure 10.3 - Photographs of the Orionte cluster

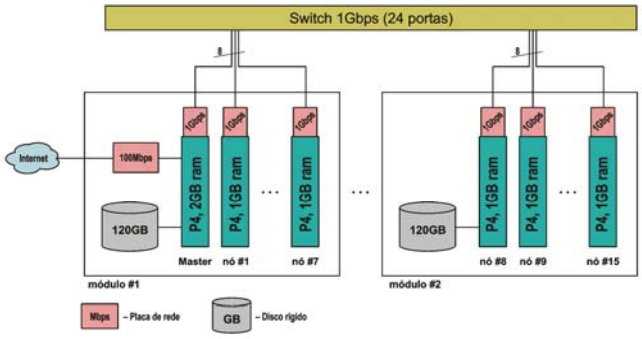

Figure 10.4 – Orionte eight board module diagram

Orionte is a"Beowulf" 2nd generation cluster, running linux Gentoo, with 2.4.28 kernel. Linux offers all the necessary tools for remote booting and simple cluster administration: DHCPD, TFTPD, SSHD, etc. The kernel is patched with OpenMosix, providing single-system-image (automatic migration of processes). An open source batch queue system (Sun Grid Engine) was also installed.

Orionte has many of the most used tools in computational physics (such as state-of-the art compilers for C, C++, Fortran and Java), high performance numeric libraries (like FFTW, Lapack-Atlas), tools and libraries for parallel programming (such as MPICH 1/ 2 or LAM/MPI), visualization tools and libraries (like OpenDX) and high-level technical computing language and interactive development environments (such as Matlab or IDL).

Orionte has been used to develop, test and run parallel applications in C,C++, Fortran and Matab using the MPI paradigm, as well as to run intensive traditional (many-runs) serial codes, graphical applications for physical visualization, computational signal processing and remote data processing and visualization. Several parallel codes have been developed and/or used by CFN Researchers, such as: AXISYMME (for magnetic-equilibria reconstruction in axisymmetric, large-aspect-ratio toroidal plasmas, taking as input the plasma-pressure and the reduced poloidal-magnetic-field profiles), CHOI (for parallel calculation and ready visualization of the Choi–Williams distribution for non-stationary fusion plasma signals processing), CRUNCH (a simplified filament current fitting code to evaluate current profile in ISTTOK tokamak from Mirnov probes), ESBEAM (based on a multidimensional paraxial Wentzel-Kramers-Brillouin approach to yield the distribution of wave potential for electrostatic lower-hybrid wave propagation in tokamak plasmas under current-drive conditions), GEM3 (a three-dimensional electromagnetic gyrofluid code to study turbulence in the boundary region of tokamak plasmas). RAYON (a parallel ray-tracing code for lower-hybrid wave propagation) and REFMUL (a 2D Finite-Difference Time-Domain electromagnetic code used in reflectometry modelling of fusion plasmas).