Available online at www.sciencedirect.com
SCIENCE DIRECT®
Fusion Engineering and Design 81 (2006) 1933–1937
Fusion Engineering and Design
www.elsevier.com/locate/fusengdes

JET real-time project test-bench software structure

N. Cruz [a,*], A.J.N. Batista [a], D. Alves [a], J. Sousa [a], C.A.F. Varandas [a], E. Joffrin [b], R. Felton [c], J.W. Farthing [c],

JET-EFDA Contributors [c]

[a] *Associação EURATOM/IST, Centro de Fusão Nuclear Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal*
[b] *Association EURATOM-CEA, CEA-Cadarache, 13108 St. Paul-Lez Durance, France*
[c] *Association UKAEA/Euratom, Culham Science Centre, Abingdon, Oxfordshire, OX14 3DB, UK*

Available online 6 May 2006

Abstract

A new test-bench for the JET real-time project was developed, being capable of generating analogue and digital stimulus signals to the control systems under test using previously stored JET pulse data. This platform allows systems to be more thoroughly tested in a wide range of scenarios before going on-line on the JET machine, reducing development and maintenance times and improving systems performance and reliability. This paper describes the real-time stimulus generator. Three layers of software which were developed to completely control 32 analogue output channels and 32 ATM virtual circuits as a real-time signal generator system:

- Signal processing on Digital Signal Processor (DSP) software directly accesses the programmable control logic, issuing all the necessary commands using a 64-bit control register and 8-channel rate change registers. Real-time data flow from local SDRAM to digital to analogue converter (DAC) channel circular buffers is also controlled by the DSP. Interrupt service routines (ISRs) were developed to Control Software variables, as well as DMA data transfers.
- Signal generation and operation as a Linux application controls the DSP in a client–server architecture. The most important functions of this software are: (i) access the JET database via MDSplus, (ii) data transfer to the local DSP SDRAM, (iii) issuing commands to the DSP state machine hardware controller, (iv) check DSP and hardware logic blocks status for errors and (v) the ATM link control.
- Remote control operation using HTTP server running CGI scripts receives the remote configuration and commands from JET operations management software interface and passes it to the high level Linux software.

Keywords: Real-time; PCI; ATM; MDSplus; HTTP; CGI scripts

1. Introduction

Complex and difficult-to-debug real-time applications are used in JET in many mission-critical tasks.

* Corresponding author. Tel.: +351 239410108; fax: +351 239829158.
E-mail address: nunocruz@lei.fis.uc.pt (N. Cruz).

0920-3796/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.fusengdes.2006.04.023

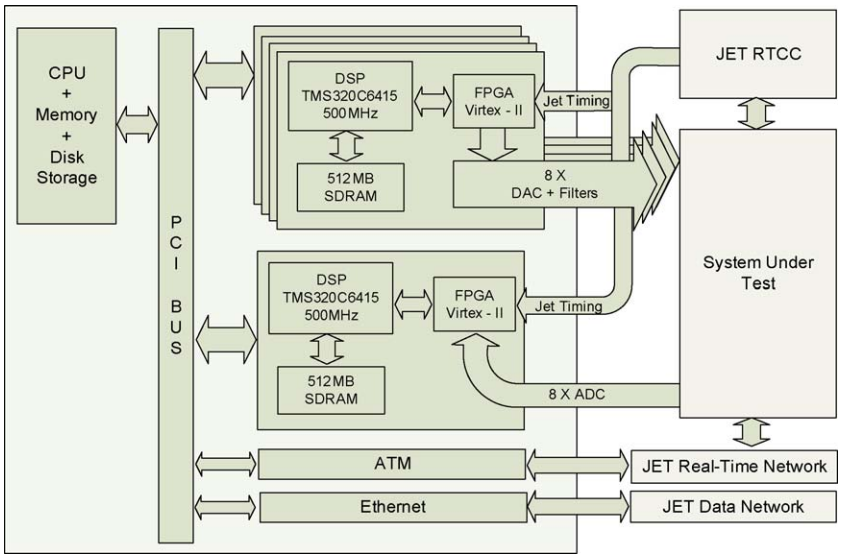

Fig. 1. System hardware block diagram.

They need to be carefully developed and tested using all possible means. Moreover, these systems are becoming components of larger and more complex systems that also need to be tested. Both functional and performance testing and fault diagnosis can be assisted by testing with realistic signals in the lab before installation and in situ. Such tests demand an accurate signal generator capable of a wide range of sample rates, synchronised to the JET timing, using user-defined simple data (e.g. sine, square and user-defined) or replaying signals from actual JET pulses. Where systems-under-test have outputs, these should be recorded and accessible to JET standard data presentation tools.

A test-bench, which generates the stimulus signals for testing a real-time control system and simultaneously recording its output variables, was developed for the JET real-time measurement and control tools [1]. The system allows testing the operation of a real-time control system by using as inputs the plasma variables previously stored on the JET pulse database or user-defined waveforms.

This paper presents the test-bench for JET real-time systems, focusing on its internal software structure and interfaces.

2. Hardware platform

The hardware is based on a standard PC running Linux with on site developed peripheral component interconnect (PCI) modules, standard Ethernet connection and off the shelf ATM module. The system uses purpose-built modules: four PCI-SG-S50M-M512 and one PCI-TR-C8I-S2M-M512. Fig. 1 shows the system block diagram.

2.1. The PCI-SG-S50M-M512 module

The PCI-SG-S50M-M512 (Fig. 2) is a new PCI hardware module with eight independent analogue output channels with 16-bit resolution and 32 MSPS refreshing rate. The module is based on a Xilinx

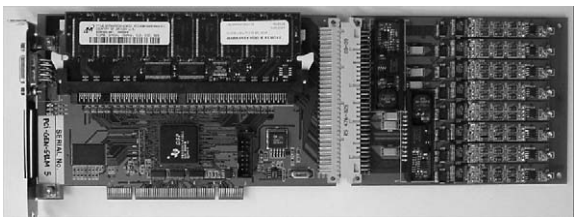

Fig. 2. The PCI-SG-S50M-M512 module.