

A large memory, high transfer rate VME data acquisition system for the JET correlation reflectometer

Nuno Cruz ^{a,*}, Rita Pereira ^a, Miguel Correia ^a, Luis Cupido ^a,
Carlos Correia ^{a,b}, C.A.F. Varandas ^a

^a Associação EURATOM/IST, Centro de Fusão Nuclear Instituto Superior Técnico, 1049-001 Lisbon, Portugal

^b Grupo de Electrónica e Instrumentação, Departamento de Física, Universidade de Coimbra, 3004-516 Coimbra, Portugal

Abstract

The current paper presents a large memory giving special emphasis to the architecture and suitability of the Linux OS for this type of applications. The system will be used on the joint european torus new correlation reflectometers composed by four microwave channels with four analogue signals each. All sixteen signals will be acquired by 12-bit VME fast acquisition channels at 2 Msamples/s achieving a total data rate of 60MB/s. Slower VME acquisition channels, with 16-bit resolution and sampling rates up to 250 ksamples/s, can also acquire these 16 signals. Two dedicated digital signal processors run an algorithm over this acquired data to avoid the use of the old KG8B root mean square analogue circuits as well as to control and to generate all the timing and acquisition clocks needed by the system. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Data acquisition; Digital signal processing; VMEbus; Transient recorder; Reflectometry; Real-time OS

1. Introduction

Correlation reflectometry with high temporal ($< 1 \mu\text{s}$) and spatial ($< 1 \text{cm}$) resolutions is an adequate tool for the study of the influence of electrostatic turbulence on improved operation regimes particularly during the occurrence of internal transport barriers [1].

The new joint european torus (JET) X-mode correlation reflectometry will have four microwave channels, each one delivering four ana-

logue signals. During the complete JET discharge 10–20 correlation measurement series shall be performed, each one corresponding to the acquisition of 10–20 frequency plateaux at a sampling rate of 2Msamples/s (MSPS) with at least 1 ksample acquired data per plateau. On the other hand, real-time plasma monitoring at a slower sampling rate is also required to detect the turbulence.

This paper describes the VME data acquisition system that fulfils the above-mentioned requirements. It has 16 fast and 16 slow input channels with large local memory, providing data transfer rates up to 60MB/s to accomplish minimum dead time between measurement series.

* Corresponding author. Tel.: +351-239-410108; fax: +351-239-829158.

E-mail address: nunocruz@lei.fis.uc.pt (N. Cruz).

2. System description and architecture

The dedicated VME data acquisition system is composed by:

- One CPU module, with a Pentium III processor running RTLinux, 256MB RAM, four RS232 ports, one 10/100Mbps/s Ethernet channel and one 4GB disk.
- Two in-house developed intelligent transient recorder modules (TRMs), with one digital signal processor (DSP) TMS320C31 running at 80 MHz, 8 input channels with 16 bit resolution at sampling rates up to 250 ksamples/s per module, and 64 kwords of memory per channel.
- Two in-house developed fast TRM units, with 8 input channels with 12 bit resolution at sampling rate up to 3 MSPS per module, and 1.5 Msample memory per channel.

Fig. 1 depicts the architecture of the system and the interface signals. The components in this architecture, as well as the way they interface with each other are described in this paper.

2.1. Intelligent transient recorder module

Fig. 2 depicts a block diagram of the intelligent TRMs.

- The main functions of these modules are
1. to provide the system acquisition clock frequency up to 3 MHz, and 1 ms time resolution to control the measurement series length;
 2. to acquire 16 channels at a slow rate for real time plasma monitoring in order to detect the turbulence;

3. to run a digital signal-processing algorithm over the acquired data to avoid the use of the old KG8B root mean square (RMS) analogue circuits.

The system acquisition clock and high-resolution counter are achieved using DSP internal timers and appropriate software interrupt service routines (ISRs). The two DSP timers are programmed to operate at different frequencies, the highest is the acquisition clock that may go up to 3 MHz (the maximum sampling rate of the analogue-to-digital converters of the fast acquisition modules) while the lowest is programmed with the desired time unit. Each clock cycle of the slowest timer generates an interrupt request (IR) to the DSP. The ISR that serves this IR, increments a counter (time stamp), which is used to start or stop the first timer for each of the measurement series.

For real time plasma monitoring eight channels are acquired in each intelligent module, data is processed by the local DSP and stored in local memory. The algorithm running in the DSP is currently under development in order to replace the existing analogue configuration and circuitry. Fig. 3 presents a schematic view of the current KG8B RMS circuits that will be replaced by the DSP real time routines. To have the data necessary to rebuild the signal with amplitude and phase information, data from two channels are acquired and processed (I and Q). The main difference between these two working modes is that by using KG8B analogue circuits the signals

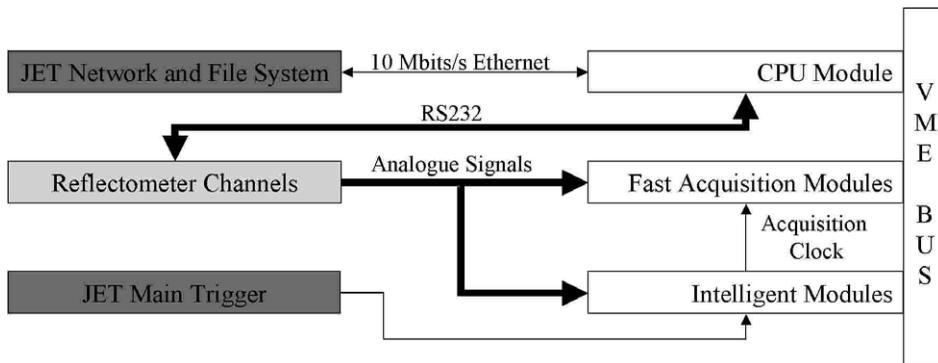


Fig. 1. System architecture and interface signals.

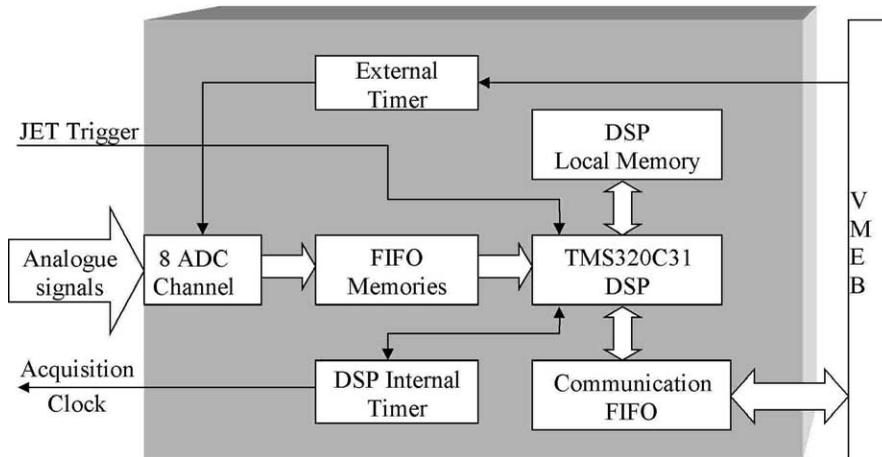


Fig. 2. Block diagram of the intelligent TRM.

were digitised after being formatted, while with this system signals I and Q are digitised and all data processing is done by the DSP in the digital domain.

2.2. Fast transient recorder module

Fig. 4 depicts a block diagram of one of the eight channels of the fast TRMs.

Each acquisition channel can store 1.5 Msamples of 12 bits, required to acquire a large quantity of data with no dead time (Section 4). To control the data flow from ADC (LTC1412) to the channel memory and to VME bus, an address generator was implemented in a programmable logic device, along with all the logic control.

The purpose of these modules is to digitise and store the incoming analogue signals from reflectometers at a 2 MSPS per channel rate.

The two fast TRMs can achieve an overall acquisition rate up to 96MB/s, which cannot be sustained when data is transferred from the acquisition boards to some other system resource due to the limitations of the VME bus performance. Optimal VME interfaces are required to minimise dead time between acquisitions. For this reason the transient recorder slave modules support Block Transfer Accesses and the master CPU module uses an industry leading VME controller.

The software controlling this device allows the

user to configure the acquisition periods and thus compromising the number of measurement series, its duration and the available memory. A set of working modes for different amounts of data is presented in Section 4.

2.3. JET network and system interfaces

The communication between the JET central control and data acquisition systems and this dedicated data acquisition system will be made using a 100Mbps/s Ethernet channel, to receive commands and send the acquired data. This choice was made based on the standardisation, performance and low cost of this technology.

The data acquired during one JET discharge is stored in the JET database after all diagnostic activities are over, at the end of the JET discharge. The diagnostic configuration will also be obtained from the JET computers through the network before the beginning of the discharge. This communication is performed as described:

- Supervisory software writes a configuration file to the acquisition system using FTP protocol, before the acquisition starts. All configurable parameters of the system are defined in this file.
- JET trigger starts the acquisition and the VME system acquires the data to a directory named after the discharge number, to files named after the channel numbers.

- When the acquisition is over, supervisory software will get the files from the VME system using FTP protocol.

The dedicated acquisition system will have a service running with a TCP/IP port where the status of the machine can be remotely checked. This service is responsible for checking all the hardware and software status to ensure that all information available is accurate and updated.

3. Real time and operating system considerations

During the last few years Linux has found increasing use in instrument-control applications, mainly due to its stability, integrity, versatility and low cost. An effort has been made to use Linux-based PCs controlling CAMAC and VME systems. Moreover, because Linux implements the same POSIX specification as commercial versions of UNIX, it supports a full range of features such as multi-user, protected-mode and multi-tasking [2].

One of the projects using VME and Linux is the VMELinux project [3]. It aims at building a Kernel Level Linux Device Driver that interfaces between the Unix Shell environment and the VMEbus. This driver is compatible with the Tundra Universe PCI–VME bridge integrated circuit and supported by various manufacturers and the device drivers developers themselves [4].

This software is governed by the GNU Free Software Foundation [5] license and includes:

- VMELinux Device Driver, which interfaces the user programs with the system device. The driver uses a standard set of calls, familiar to most programmers.
- VMEUtils Program, running in a Unix shell

which communicates directly with the Device Driver. It's an example to the programmer of how to interface the driver and what are its main features.

- VMEShell Utilities, which are a set of routines to be called from the Linux shell and which may be used to read data directly from VMEbus as well as to be used as examples to the programmer.

Another problem that usually comes along with fast data acquisition systems is the need for a real time operating system (RTOS). 'Real-time' is an overused expression with the common meaning of 'very fast', 'as soon as possible' or 'almost at the same time' (like real-time stock quotes over the Internet) among others. However, what real-time means is that a process must respond with an appropriate action within a specified time limit.

This system is designed to run in a real-time environment. This means that there are timing deadlines that cannot be overcome under any circumstances, at the risk of data loss. No matter how fast the system usually is, if it doesn't read a buffer before it is full, even if it is only one time, all data read out and stored may be compromised.

In Section 4.1 is described an implemented working mode in which memory is readout only after the acquisition has ended. In this mode a RTOS would not be necessary, however, the working mode described in Section 4.2 demands the use of a RTOS due to the specified time limits to read out the acquisition channels.

The choice of using RTLinux [6,7] was made because it is a real-time variant of Linux OS therefore allowing to control data acquisition systems and other time sensitive instruments and devices with accuracy. This RTOS is currently

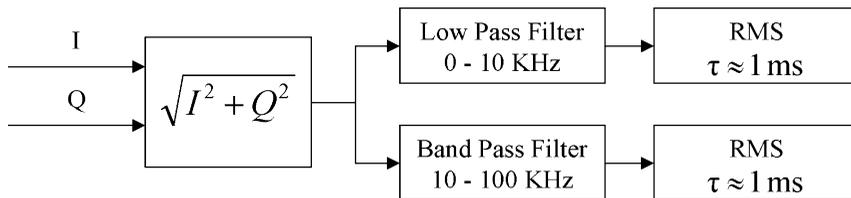


Fig. 3. Block diagram of KG8B RMS circuits which is the flowchart of the algorithm running on the DSP.

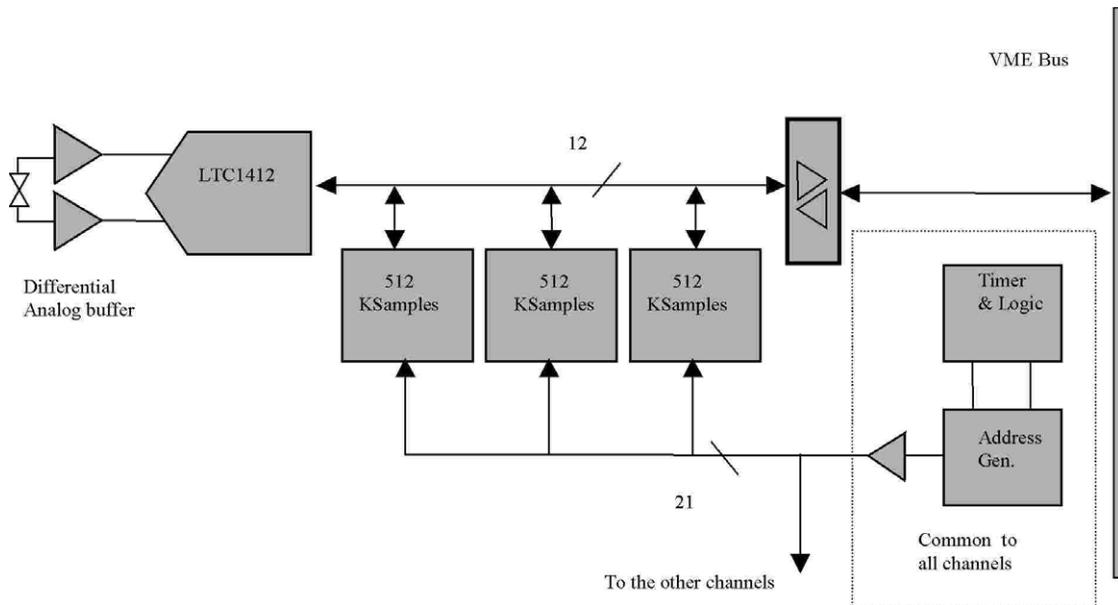


Fig. 4. Block diagram of one acquisition channel of the fast TRM.

being tested in this application and results will be available soon.

4. Results and system performance

This system is capable of acquiring 16 channels with rates up to 3 MSPS per channel directly to its internal memory at a total rate of 96MB/s. This rate cannot be sustained when data is transferred from the acquisition boards to some other system resource, due to VME bus performance limitations. The maximum VME throughput achieved using this system was 16MB/s. In spite of this particular constraint, several important cases can be described in which the overall performance of the system can be assessed and shown to be adequate to the envisaged application.

A few meaningful hypothetical application cases are described to illustrate the capabilities of the system.

Let us define a measurement series as the acquired data for all frequency levels. The data length acquired for each measurement series is attainable multiplying the number of frequency

levels by the data length acquired in each level. For instance, a measurement series that uses 10 levels of frequency of 8 ksamples each will have 80 ksamples of data length, taking 40 ms sampling period at 2 MSPS.

4.1. Case 1—acquisition to the channel memory

Let the acquisition rate be 2 MSPS per channel. Eighteen measurement series of 40 ms each (80 ksamples/measurement series) can be performed with no dead time between them, possible due to the large internal memory associated to each channel (1.5 Msamples). It is possible to make any combination of number and duration of measurement series providing that its total size is smaller than 1.5 Msamples per channel.

4.2. Case 2—read out to CPU memory using the VME bus standard access and the VME bus block transfer

This case shows the possibility of acquiring large amounts of data during extended periods of time and thus minimizing the dead time between measurement series.

Again, let the acquisition rate be 2 MSPS per channel. The system's total input rate is 64MB/s, given by 16 channels at 2 MSPS and 2 bytes per sample. Based on a standard VME transfer rate (using D32 Standard Access) of 4MB/s and VME D32 Block Transfer that enhances the data transfer rate in the VME Bus to 16MB/s, the readout to CPU memory is 16 times and 4 times longer than the acquisition period. This means that it is possible to have longer measurement series up to 1.5 Msamples knowing that the dead time is, in the best case, four times longer than the measurement series.

This data rate (16MB/s) is tested and was achieved; however it is not sustained for long periods of time, due to internal delays in the CPU board. It is still under development to achieve longer periods avoiding the PCI latency internal to the CPU board.

In this mode only the CPU DRAM (256MB) limits the amount of data to be acquired.

5. Conclusion and future work

In reflectometry diagnostics very large amounts of data must be acquired in very short time intervals. This fact leads to very high data acquisition rates and to the need of very large storage devices and amount of memory.

The need of simplifying analogue circuits that are a potential cause of system errors and malfunctioning leads to the development of digital processing units to replace the old circuits. The easy re-programming and ability to use different settings and algorithms for different situations is another advantage of digital signal processing.

The use of general public licence software like RTLinux and VMELinux Device Driver provide

low cost solutions keeping high quality standards and performance.

In this project all these features and technologies were gathered to develop a large memory, high transfer rate VME data acquisition system for the JET correlation reflectometer. This system is working in Portugal, where several bench tests have been performed. It will be installed in January 2002 in JET facilities, where tests will be performed in loco.

The choice for standard technologies with large use in industry, as well as the experience attained when developing previous VME modules helped to develop a high performance system.

Acknowledgements

This work has been performed under the contract number FU05 CT 2000-00242 between the Commission of the European Union and Instituto Superior Técnico, in the frame of the European Fusion Development Agreement.

References

- [1] M.E. Manso, F. Serra, B. Kurzan, I. Nunes, A. Silva, W. Suttrop, P. Varela, S. Vergamota, H-Mode studies with microwave reflectometry on ASDEX-Upgrade, Plasma Physics Controlled Fusion 40 (1998) 747.
- [2] D. Marsh, Linux a worthy OS for Real Applications, Test and Measurement Europe, May 1999.
- [3] VMELinux Organization. Available from <http://www.vmelinux.org>.
- [4] Commercial VMELinux. Available from <http://www.vmelinux.com>.
- [5] GNU. Available from <http://www.gnu.org>.
- [6] Real-Time Linux Organisation. Available from <http://www.rtlinux.org>.
- [7] Commercial Real-Time Linux. Available from <http://www.rtlinux.com>.