ARM BASED EMBEDDED EPICS CONTROLLER FOR BEAM DIAGNOSTICS IN CYCLOTRONS AT VECC

S. Sahoo[#], N. Chaddha, T. Bhattacharjee, R.B. Bhole, A. Roy, S. Pal, Amitava Roy
Variable Energy Cyclotron Centre, Kolkata, India

Abstract

Advanced RISC Machines (ARM) processor based embedded system is used to port EPICS for the development of input-output controller for beam diagnostics applications in K=130 Room Temperature Cyclotron and K=500 Superconducting Cyclotron at Variable Energy Cyclotron Center (VECC). The beam diagnostics system, used for efficient transportation of the accelerated ion beam up to the experimental area for both the cyclotrons, consists of many hardware devices to be interfaced into the EPICS based control system. The development of ARM based embedded controller card has replaced the existing PC based systems. This controller card can either be used as a stand-alone IOC or as a small plug-in module in one slot of a crate where many beam diagnostic hardware module with specific functionalities are placed side-by-side. This is having an obvious advantage of integrating the control system inside the one hardware crate thus reducing the cable and overall hardware complexities which were involved in the PC based system. This paper explains the steps involved in designing the ARM based EPICS controller for beam diagnostics and development of Graphical User Interface (GUI) for operators by using Microsoft Visual Basic (VB) and indigenously developed EPICS Channel Access embedded ActiveX components at VECC.

INTRODUCTION

The Beam Diagnostic systems of the beam line in the two cyclotrons at VECC, i.e. Room Temperature Cyclotron (RTC) and Superconducting Cyclotron (SCC) help to transport accelerated beam efficiently up to the experimental area at various caves. There are many parameters which are required to be controlled and monitored to obtain the information about beam position and intensity. The beam tuning often requires reducing the beam size as per the need of experimentalist. Beam diagnostic components like Beam Viewer, Faraday cup, slit etc. are used in the beam-line for this functionality. These components are required to be handled remotely in a control system. The functionalities of the required hardware to be interfaced with each beam diagnostic components are common in many respects and hence customized development of this hardware as per functionalities can reduce the cost and complexity at the same time unlike using commercially available devices. The modular design of the hardware also facilitates easy up-gradation and ease in problem identification during regular maintenance.

_____

ssahoo@vecc.gov.in

A crate, containing many modular hardware cards plugged in the backplane, was developed to cater beam diagnostic requirements at VECC [1]. These modular cards were initially controlled separately from a traditional PC based system and were not included in the EPICS based control system. However, the development of EPICS IOC for a PC based system by replacing the old control software will not be an optimized solution as it requires addition of extra hardware and field cables in the system. Moreover, the processing power and resource of a PC is not efficiently utilized if the IOC dedicated for beam diagnostic purpose, runs on it. With this motivation, an ARM based controller with embedded EPICS is developed for the beam diagnostics application and the beam diagnostic system is included in the EPICS based control system. This embedded controller can be inserted into the beam diagnostic crate as a small plug-in module with EPICS IOC running on it.

CONTROL ARCHITECTURE

The control system, for beam diagnostics in both the cyclotrons, follows three-layer software-hardware architecture. The upper layer is the front end control i.e. the operator interfaces (OPIs) or GUIs which are developed using EPICS Channel Access (CA) embedded Active-X components in Microsoft Visual Basic (VB) developmental platform. The EPICS IOC, runs on the ARM based Single Board Computer in the middle layer, acts as master controller and controls various functional modules for beam diagnostics hardware. The lower layer is the hardware modules which are directly connected with the actual beam diagnostic equipment in the beam line. Figure 1 shows the control architecture layout of the whole design.

Each hardware module is designed to have four building blocks viz. µ-controller, functional part, communication and power supply sections. AVR and C51 family controllers are used for the µ-controller part. Functional part is designed as per the common functionalities of beam diagnostics equipment. Interface of these modules to main controller card is through dedicated TTL RS-232 communication lines i.e. Rx (Receive), Tx (Transmit) and Ground.

At present there are six pluggable modules in the Beam Diagnostics crate which are listed as below:

1. ARM based Master controller card
2. Stepper Motor module
3. X-Y Slit module
4. Encoder Read-Out Module

5. Relay Module
6. I/O Module

All these modules are connected to the controller card through dedicated TTL RS-232 lines in the backplane as described earlier. Thus failure of any of these modules will not affect the function of others. Figure 2 shows the beam diagnostics crate with hardware modules and master controller card connected.
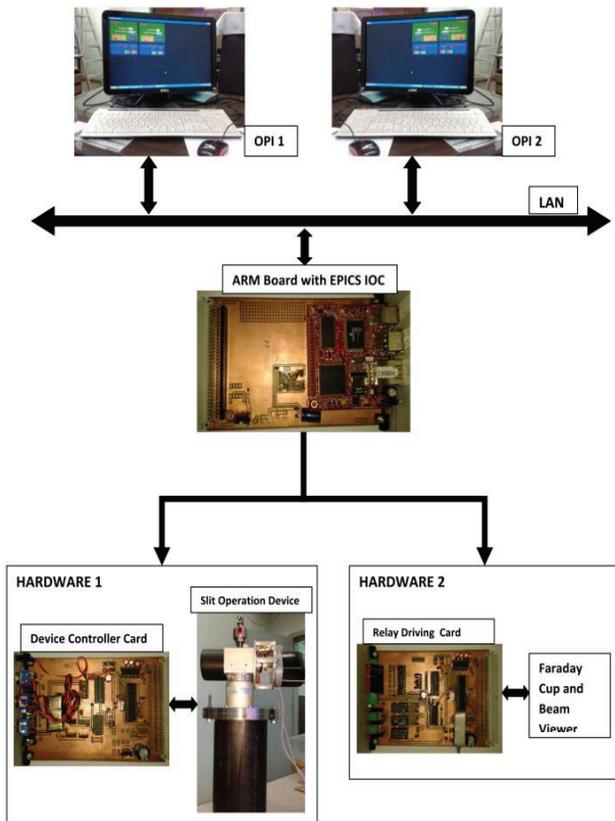
Figure 1: Control Architecture in Beam Diagnostics system at VECC.

Figure 2: Beam Diagnostics crate.

TS-7500 ARM SBC

The TS-7500 [2] is a compact Single Board Computer (SBC) based on a 250 MHz ARM922T CPU from Cavium Networks. This board, as shown in Fig. 3, is used to embed the EPICS. Table 1 lists the specifications of the TS7500 board in details. Other than one dedicated TTL UART interface, the board also contains eight TTL XUART interfaces designed on the on-board FPGA by Lattice Semiconductor. These XUART serial interfaces are used for communication with the different hardware modules described earlier.

Embedded Linux on TS-7500 ARM SBC

The TS-7500 SBC loads the Debian Linux OS based on Linux 2.6.24 kernel from the Secure Digital (SD) non-volatile memory card. The full featured Debian Linux (version Lenny Arm) distribution includes a complete GNU C/C++ embedded development environment. In addition, Apache Web Server, FTP, SSH, Telnet and Samba network services are available with C/PHP/Perl for embedded CGI development.
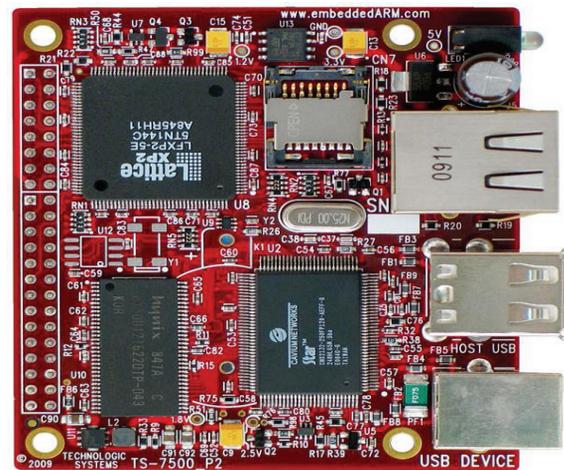
Figure 3: TS7500 Board.

Table 1: TS7500 Hardware Specifications

Item		TS-7500
CPU		ARM922T (250Mhz), From Cavium Networks
OS		Linux 2.6.24 (Debian)
Memory	DDR RAM	64 MB
	SRAM	512 KB
	Flash ROM	4 MB
	External SD Card	2 GB
Peripherals	DIO	33 Pin DIO
	USB 2.0	2 (Host) and 1 (Slave)
	Ethernet	10/100 Mbit
	UART	1 TTL serial console port and 8 XUART TTL serial ports.
Power	Low Power	400mA @ 5V

Software Development on TS7500

For application development, cross compiling the source files in standard PC based Linux environment is the approach that we have adopted to develop embedded EPICS on this board. For cross-compilation, gnu-tool-chains for ARM are used and the resulting binary and library files are copied to the SD card for execution on the TS7500 board.

STEPS FOR BUILDING EMBEDDED EPICS IOC

EPICS Device Driver

'Asyn' software module is used to develop the EPICS device driver as per the communication protocol of the hardware. The communication protocol of all the hardware modules are developed in-house following the standard ASCII MODBUS protocol. The format includes start character followed by unique Command ID, 4 bytes of data fields and LRC. The software coding is done according to the protocol to link the hardware parameters with Process variables (PVs).

EPICS Database

The database file (db file) describes the actual EPICS process variables associated with the beam diagnostics application. In our application, we have created a separate db file for each of the hardware modules to maintain modularity in design and easy debugging during maintenance. In total, there are about 20 process variables required for each hardware module which results a total of 100 records in our system for the beam diagnostics.

EPICS IOC for ARM Target

After proper installation of the ARM cross compiler on the developmental platform, application source codes are built for arm-linux target architecture. After successful compilation of the EPICS base and asyn module, the application (say "beamdiag"), is compiled to generate the required binaries, library files, database files and database definition files which are then copied into the SD-card of ARM board (TS7500). Some kernel tweaks have been done to load the EPICS IOC just after boot-up before logging into the user profile.

OPERATOR GUI USING VISUAL BASIC

Considering the operators' familiarity and compatibility with third party application software, MS-Windows is chosen as the platform for operator interface (OPI) in the control console of the cyclotron. Although there are standard X-windows based EPICS tools e.g. EDM, MEDM etc. for developing OPIs, yet using these OPI tools in windows platform require *X*-Server (e.g. *Exceed*, *Xming*, Cygwin/*X*) with expertise during configuration. Hence, Microsoft Visual Basic (VB) is chosen as the OPI developmental platform considering its object oriented structure, rich GUI library and less complicated coding

style. A set of indeginiously developed CA embedded Active-X components e.g. *CA Text, CA Image, CA Setpoint* and *CA button* [3] are used to develop the OPI for beam diagnostics application. Figure 4 shows some screen-shots of the present GUI, developed using these Active-X components in VB, at the operator control room.
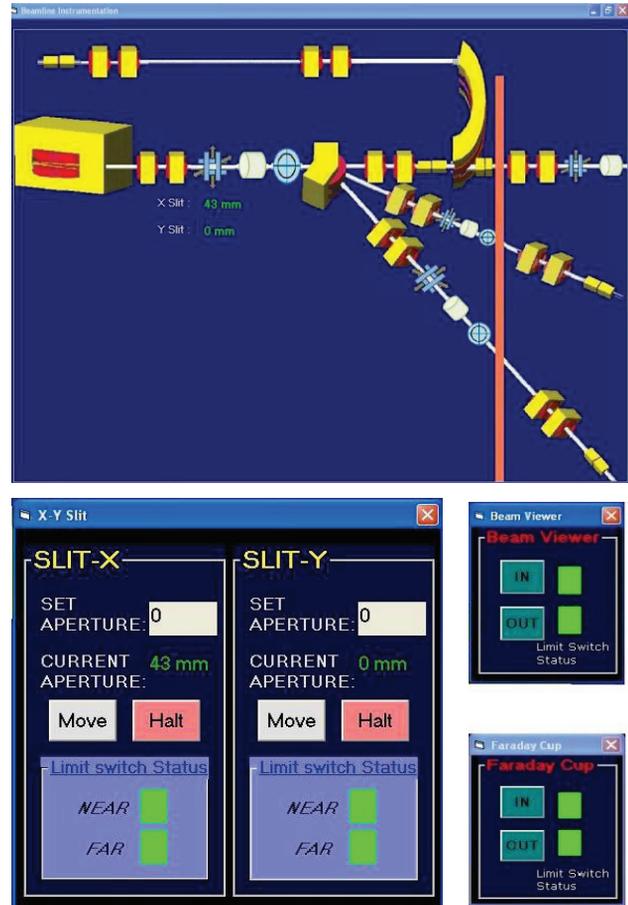
Figure 4: Operator GUI developed in Microsoft VB.

PROBLEMS DURING OPERATION AND PROPOSED SOLUTION

The developed system is running round-the-clock in room temperature cyclotron beam line. The major problem faced during its operation was frequent damage of the SD-card containing the operating system and IOC. The problem is addressed and it was found that radiation may be a reason for the damage of the SD-card as the crate was placed very near to the highest radiation zone in the vault. The complete crate is moved far away where radiation is considerably less as compared to the previous location by extending the field cables of the actual beam diagnostics instruments.

The ARM board presently does not have any network boot-up option and it has a proprietary boot code in the FPGA that is used to load the kernel from the desired medium which starts executing the kernel and ends up in

the fast-boot shell. The fast-boot shell is equivalent to traditional embedded boot-loaders like Red-Boot or U-Boot shell. The 4 MB on-board SPI Flash card contains two partitions viz. raw linux kernel and initrd/busybox. It is proposed to implement a net-boot option in it which will help us to replace the SD-card and the controller will act as a disk-less device and IOC can be loaded from a network disk.

FUTURE UPGRADATION

In future, it is proposed to replace the ARM9 processor with a soft core processor. As opposed to ARM9 processor which is fixed in silicon, a soft core processor (like Nios from Altera or MicroBlaze from Xilinx) can be customized for a given application and synthesized for an FPGA. Porting of Channel Access server on MicroBlaze processor and analyzing its performance is already done [4]. This technology, if implemented, integrates the IOC and digital control hardware within a single FPGA, thus reducing the overall hardware complexities of field devices.

CONCLUSION

We have successfully installed EPICS IOC on ARM9 processor and developed the controller card for beam diagnostics in cyclotrons at VECC. This controller card is interfaced with different functional modules required for diagnostics purposes and individual device drivers have been developed. The OPI is developed using EPICS Channel Access embedded ActiveX components in Microsoft Visual Basic (VB) development platform. The system is running round-the-clock successfully.

Although the beam diagnostic system does not demand real-time requirement, however, the soft real-time response of EPICS can be achieved with this controller by proper configuration of hardware and software during development. This will facilitate the use of this embedded EPICS controller in applications which require deterministic response.

REFERENCES

[1] N. Chaddha, R.B. Bhole, S. Sahoo, P. P. Nandy, S. Pal, "Modular Beam Diagnostics Instrument Design For Cyclotrons", p. 154, PCaPAC'12, Kolkata, December 2012.

[2] TS-7500 Documentation, Technologic Systems," http://wiki.embeddedarm.com/wiki/TS-7500

[3] A. Roy, R. B. Bhole, S. Pal, "Development of EPICS Channel Access Embedded ActiveX components for GUI development", p.54, PCaPAC'12, Kolkata, December 2012.

[4] S. Sahoo, T. Bhattacharjee, S. Pal, "Development and Performance analysis of EPICS Channel Access Server on FPGA based Soft-core Processor", p.274, PCaPAC'12 , Kolkata, December 2012.