PROSCAN CONTROLSYSTEM STATUS REPORT

D. Anicic, G. Dave, T. Korhonen, A.C. Mezger

.

INTRODUCTION

Three control systems comprise the PROSCAN control system: COMET control system as provided by the supplier ACCEL, Machine Control System (MCS) and User Control Systems (UCS) for each Gantry, Optis and Experimental area. Here we report the status of our part, the MCS, and its integration with the other two systems.

The MCS is basically a copy of the High Intensity Proton Cyclotron (HIPC) control system, with some additional components, in particular the Beam Allocator (BALL) System, handling the requests of the UCSs. Here we report the progress in development and installation of hardware, software and BALL.

THE MCS HARDWARE

The hardware for the MCS consists mainly of standard control systems components that are already in use in the other PSI accelerators. The hardware is based on the VME64x standard and the extensive use of mezzanine cards like Industry Packs. CPU cards from Motorola and the Industry Pack (IP) carrier boards from Hytec Electronics are used. Control of devices that need an intelligent low-level controller (for example to be able to monitor the device status and generate interlock signals) has been implemented using intelligent IP carrier VICB8003 from Hytec.

The control system infrastructure is in place as far as the construction allows and the components of the first section (up to degrader) have been connected to the devices and configured with corresponding firmware.

The components to be controlled in the first section are the magnets, the drives for the beam blockers and the profile monitors and the energy degrader.

The degrader has two axes that have to drive synchronously. The drive is implemented with stepper motors and the amplifier developed for SLS was selected, with a small modification for the use with brakes. The degrader controls have been extensively tested and the requirements for speed and accuracy have been fulfilled.

The kicker magnet is used for fast beam on/off switching. For the kicker, a switch that allows the safety systems (RPS and PaSS) to be connected was developed and installed.

The Run Permit System (RPS) has been implemented in VME hardware using IP modules and corresponding transition modules to connect to the signals. The firmware logic for the first sections has been implemented and is under testing at the time of writing.

The signals for the Patient Safety System (PaSS) have been defined and the wiring prepared for the existing components. The PaSS implementation is underway; the structure has been defined and is being refined and undergoing analysis for safety.

So-called safety switch boxes to interface the safety systems (RPS, PaSS and PSA) to the components that are used to switch off the beam have been developed by ACCEL to our specifications and have been installed, tested and connected.

THE MCS SOFTWARE

The MCS software is the copy of the existing HIPC control system. For its implementation, we have installed the server and the operator workstation computers as well as the data acquisition and control computers (IOC).

The server and the operator workstation computers are running a Linux (Fedora Core 1) operating system. There are actually four servers: the IOC boot server, the file and the development server, the database server and the archival server. Installed, is also a dedicated workstation/server for the BALL functionality. We have currently two operator consoles, one in the control room (Fig. 1), consisting of two workstations with four screens each, accompanied by one “knobbox” (the four-knobs operator convenience tool). Another one is a workstation with two screens installed near the PROSCAN HW equipment for test and commissioning purposes. At this moment, only two IOCs are necessary. One is dedicated for connection to the ACCEL SPS based control system and one for the first section of the beam-line up to the degrader.

All servers are fully functional. The boot server boots the IOCs and contains their configuration and log files. It also acts as a Dynamic Host Configuration Protocol (DHCP) server in the PROSCAN MCS sub-network. The file server contains the control system client software (for operators, physicists and HW experts). The Database server runs the ORACLE database and the corresponding data entry and retrieval applications. It also distributes the necessary config-files for the IOCs and the workstations. The archival server runs the data-archiving, and also the centralized message-logging and the set-value-logging applications.

The IOC software [1] supports all hardware and functionalities being used at this moment. The additional drivers will be implemented as soon as needed and as additional hardware becomes available. Access control has also been implemented, but will not be used until it becomes necessary. The BALL has still to be extended for this functionality.

Fig. 1: PROSCAN control room console.

The IOCs already have an almost complete list of the data acquisition and control parameters for the ACCEL SPS section driving the COMET accelerator and also for the starting beam line section up to and including the degrader. There is still no write (set values) access to the SPS (COMET) parameters, because it has not yet been documented by ACCEL. The COMET control system will be commissioned with its own SCADA based system. Afterwards, MCS will take over.

THE BEAM ALLOCATOR SYSTEM

The primary task of the Beam Allocator is to handle the requests of the beam users (Gantry1, Gantry2, Optis, Experiments) and to control the beam.

To implement this functionality, various software components were developed, as shown in Fig. 2. In particular, the components: Beam Allocator, Machine Configurator and Tune Manager, which communicate with the underlying MCS.

The BALL system is a Client/Server design, using multi-processing and multi threaded techniques to decouple the components and handle them separately. This gives the flexibility of being able to run the components on separate computers if needed. The components communicate with each other using an interface protocol of request and response sequences. The communications between the components are established using TCP/IP and UDP protocols. The Beam Allocator starts individual 'Listeners' to handle and control each new UserClient as they establish contact. The Beam Allocator controls the UserClient throughout its life cycle. Should there be a break or loss of communications, the Beam Allocator will disengage gracefully, cleanup and recover safely. The StatusClients for data and information display are handled on the same basis. The StatusClients can also be broadcast site wide. The interlock aspects interact with the BALL through communications with the RunPermitSystem.

The Beam Allocator does the Tune setting by using the component Machine Configurator for high speed response. The requested Tune is processed and the devices are configured through the IOC's.

The Tune Manager is used to develop and manage the Tunes and their relevant files. The Tune Manager's file editor allows file selection, management and interpolation functions. Individual or multiple devices can be set, and on selection, a particular Tune can be set to configure the beam characteristics. Individual or multiple magnets cycling facility is also available. The magnet cycling profile is specified using a cycling definition file. The selected Tune files are loaded in the shared memory area for the Machine Configurator to access at high speed for Tune setting.

Through the set of available commands, some of which are, Login, RequestMastership, ReleaseMastership, ReadDevice, SetDevice, DoTherapy, SetOperationsMode, SetTopology, the user performs the required tasks. The beam is requested by using Beam settings for a particular 'Tune' with certain energy, intensity, emittance, gantry angle and other parameters. These are specified in XML type Tune files which have a particular name format which identifies a particular Tune and its characteristics precisely.

To facilitate the user interaction with the Beam Allocator, an application programming interface has been developed in C++.

The primary functionalities of each of the components have been developed and demonstrated in the development environment. Each component has been demonstrated individually. The Tune Manager is being run and tested in the operational environment. The systems integration phase for the components working together has also been achieved and demonstrated. The whole system has now been transferred to the PROSCAN machine environment to establish run ability and to develop and establish a test environment and a formal release environment.

The detail development of each of the components is proceeding. As functionality increases and modifications become incorporated, each of the system components will progress incrementally through to the release environment.

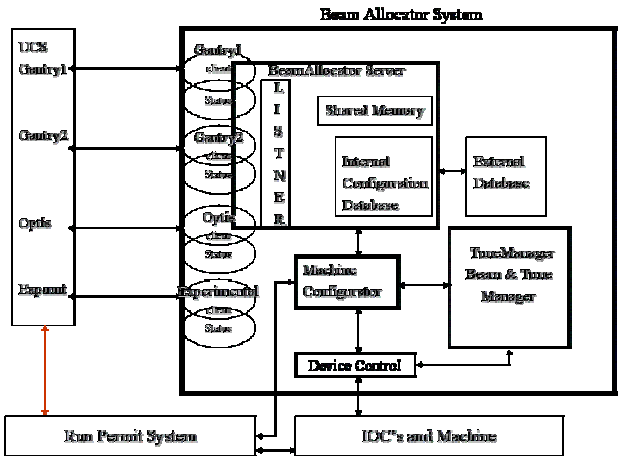

Fig. 2: The Beam Allocator environment.