The Tandem Control System

L. Beck, L. Rohrer, and H. Schnitter

The control system for the accelerator and related systems works with microcomputer crates as local control units (nodes) and personal computers, connected by a local area network. In most applications the local units act as servers, while the desktop PCs are the clients. But, as appropriate (e. g. high voltage stabilization system), each node can act as a client: send requests to other nodes to control remote devices or to get measurement readings.

Each of the approx. 50 local crates is equipped with I/O-boards according to the devices to be controlled (digital parallel or serial and analogue). As program language and communications language Forth is used. Each microcomputer is programmed via LAN so that it can perform its specific control or monitor task. Program code and actual data are stored in a non-volatile RAM and reloaded after a cold start. So a microcomputer resumes its job from the latest status after a failure.

The application programs running on the personal computers (C++ and Visual Basic) were subject to substantial changes when we changed from Windows 3.11 to Windows NT or Windows 2000. Part of the modifications, particularly of the transport layer between the LAN and the application programs, and many useful ActiveX-components were done by FORTecH Software, Rostock. Most of the application programs (Visual Basic) were redesigned in our laboratory. All application programs refer to a common Access database ONF32.MDB. A revue is given below.

'Rackviewer' is our standard control desk designed for quick and easy configuration. The operator can select by menu picking screens with 12 standard cells each. A cell represents a device. It has on/off controls, graphic and numeric analogue display, and data input by mouse or keyboard, as appropriate. To compose the screens and the menu structure a designer program is available. Using this one can assign cells and menu items by drag and drop. The result is stored in the common data base and is effective in all rackvievers.

A more complex control desk 'BildKp' shows the controls not in container cells but individually sized and positioned on a graphic presentation of the equipment. For the rest the graphic desktop has the same functionality as the rackviever. A designer, similar to the VB-tool for design of VB-forms, is available. In addition to this configurable graphic desktop a fixed version is in use showing radiation levels and beam lines, indicating the actual beam by colour.

'Sequencer' can be used to repeat easy series of operations. It is used to degauss the switching magnets with alternating currents of a decreasing amplitude in time. Conditioning the tandem for high voltage is an other important application. Here the charging currents of the pelletron chains are increased or decreased in dependence of the level of X-ray emission.

Two programs are available to record measurement readings: 'Spektrum' displays the beam current versus ion mass. For this purpose the inflection magnet current of the ion injector is varied slowly while the magnetic field and the beam current are measured. 'Schreiber' emulates a strip chart recorder with 8 channels. A visualisation program for our beam scanners is not yet converted to Windows NT or Windows 2000. The redesign is in process.

The program 'Beam' calculates initial set values for the accelerator and the beam transport system. It uses an Access database containing an atomic mass table, tables with stripper data, and tables with parameters of devices to be controlled. The program has access to the machine so that the calculated values can be set immediately.

With the program 'Logb' one can take readings of all relevant data. The status of the machine is saved in a database, and selected data can be printed in adequate formats. Data from earlier readings can be extracted from the data base and displayed or printed. 'Setlog' has access to the nodes in the LAN, so the accelerator facility can be set according to an earlier run.

Finally, some maintenance tools for the control system are worth mentioning. 'Terminal' connects a remote node to a text window. 'DeviceTester' displays the properties and methods of a device and tests whether it is controllable. 'DownloadManager' supports programming remote nodes and keeps track of source code versions. 'DeviceBrowser' lists all available devices and the garbage collector 'GC' removes dead devices.