JAERI FEL CONTROL SYSTEM

Masayoshi Sugimoto, Ryukou Kato, and Ryohji Nagai
Japan Atomic Energy Research Institute
Tokai-mura, Ibraki-ken 319-11, Japan

Abstract

The control system is developed for JAERI Free Electron Laser facility which aims to generate far infrared laser using the superconducting linac. It consists of the networked PCs and the stand-alone CAMAC systems for device control. The software is operated under Microsoft Windows and the object-oriented approach is employed.

Introduction

JAERI has constructed the Free Electron Laser (FEL) facility to produce the far infrared (20-50μm) laser using the superconducting linac of 499.8 MHz frequency [1]. The control system of the accelerator components are designed [2] and implemented on the peer-to-peer network of PCs which access the devices through the CAMAC systems. Microsoft Windows is employed as the standard interface to the operators and all controls and monitors are carried out by using the graphical representation.

Hardware configuration

The basic unit of the control hardware is a "local unit", which is a stand-alone CAMAC system with a PC controller. It works as a "console unit" when the devices are remotely controlled, and also works as a software development machine by assuming the virtual devices. The present status of the network configuration and the major components of the facility is shown in Fig. 1.

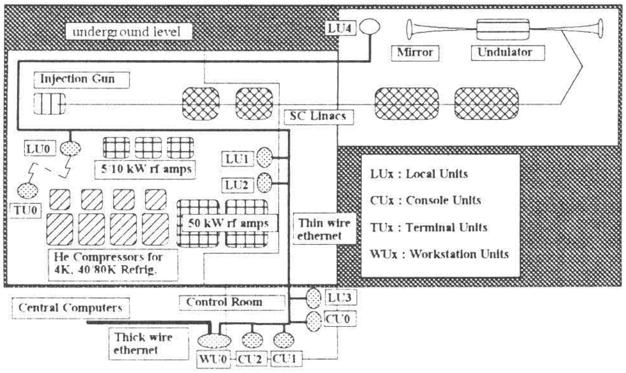


Fig. 1. Present status of the network configuration of the control system. Thin wire ethernet is used as a control network and thick wire ethernet is connected to the central computers through a workstation.

All nodes at every place in the facility can communicate with a thin wire ethernet using the proprietary protocol, which is designed that only a single packet is enough for the most communications. A terminal can be connected using the wireless modem as the portable node, which are convenient to monitor the status during the maintenance or local adjustment of the devices. A workstation is used to access the central computers through the laboratory-wide ethernet. At present, there are five local units and three console units, and the total number of nodes can be doubled without a degradation of the performance. It is the most cost-effective choice for such a relatively small system.

Software layer

The control system software is designed to have three abstraction layers: (1) the device layer, which includes the device handler and the corresponding (real or virtual) object database, (2) the controller layer to abstract the common actions (on/off, setting, reading, etc.) for the devices, and (3) the host layer, which includes several actors (e.g. "panel", "logger", "simulator", "inspector", and "helper"). Such actors can access any real and virtual objects in the device layer through the controller layer ("virtual controller"). These can be independently developed based on the object-oriented approach for easy maintenance.

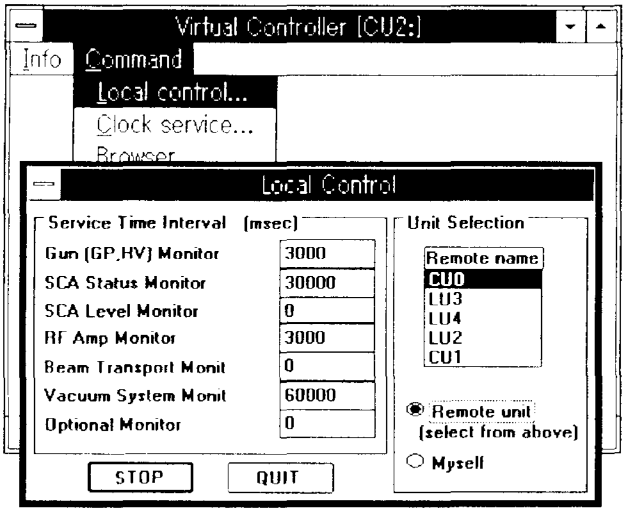
Device layer

The device drivers are implemented as Windows dynamic link libraries, which can be loaded and accessed from any other applications only when they need. The real devices of a controller are a CAMAC and an ethernet interface boards.. The modules in the CAMAC station, the GPIB devices through the CAMAC-GPIB interface module, and the other controllers are all abstracted objects. Such objects may exist, actually or virtually (within the software simulator).

Controller layer

The virtual controller program handles the typical messages to the devices, which may reply answers or not. The actions to the messages are embedded in the object methods in the device layer. The message issued from one controller is delivered to other controller which is responsible to take an action for the message. The addresses are written in the device configuration files, the property lists of the instances of the device class, and they are easily modified. Some messages can be delivered through a gateway which transfers them to the other node by rewriting the device configuration locally.

As shown in Fig. 2, a hardware diagnosis about the interface boards are carried out, and the behavior of the remote nodes, such as the service time of the monitoring, can be redefined. These are very convenient for keeping the system performance as the optimal one.



Service Time Interval (msec)

Gun (GP,HV) Monitor	3000
SCA Status Monitor	30000
SCA Level Monitor	0
RF Amp Monitor	3000
Beam Transport Monit	0
Vacuum System Monit	60000
Optional Monitor	0

Unit Selection

Remote name: CU0, LU3, LU4, LU2, CU1

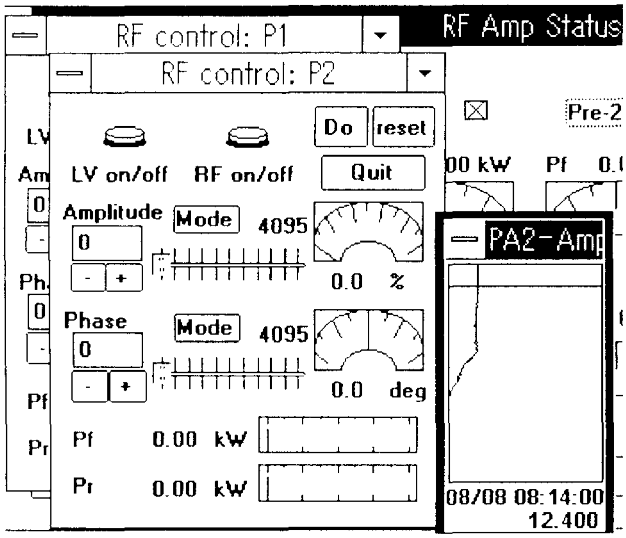
● Remote unit (select from above)

○ Myself

Fig. 2. The virtual controller window and a local control dialog to show the status of the remote nodes. The service times can be modified and sent to each node.

Host layer

Any application programs can become as the actors in the host layer when they support the dynamic link library calls. We developed a bunch of "panels" which are similar to the diagrammatic display panels. Fig. 3 shows an example of such panels, the control panel for rf amplifiers. The devices controlled by the single panel in Fig. 3 are distributed over three nodes, actually. A strip chart attached to each meter can be used to confirm the status trend in recent several hours.



Amplitude 0 — 4095 — 0.0 %

Phase 0 — 4095 — 0.0 deg

Pf 0.00 kW

Pr 0.00 kW

08/08 08:14:00 12.400

Fig. 3. An example of the panel program consists of a main window and control dialogs for the rf amplifiers. A dialog contains two switch buttons, two sets of control monitor for a low level rf amplifier, and two power meters. A strip chart associated with each meter shows the trend of the parameters in a short term.

The "logger" produces three different types of the data logs: (1) long term trend data of each parameter, which are similar to the recorder charts and recorded automatically, (2) data sheet form to view all data at the specified instances, and (3) single line log to present the status of the routine work briefly.

The "simulators" are independent programs to evaluate the current status of the beam dynamics and the performance of the laser oscillation. They can collect the input parameters from the control system to produce the on-line and realistic results. The beam dynamics code developed for designing the FEL system before [3] is converted to work with the Windows and to get the input data from the control system. As the simulation of the laser oscillation is time-consuming, the summary of the results are stored in the database and the appropriate data are retrieved when they are requested.

The "inspector" program is no more than a diagnostic expert system to watch the data trend and to warn the operator about the malfunctions if necessary.

The "helper" is a Windows help system to offer the help information to the operator.

Finally, a commercial programs which support the dynamic link library calls, such as spreadsheets, word processors, and database managers, can be also the actors of the control system by adding the interface software modules for each. For example, a spreadsheet program can show the data log in a sheet form, and operator can modify the value cells to send the new data.

Conclusions

We developed the control system for JAERI FEL using the networked PCs operated under the Windows. The CAMAC is used as the device interface and a graphical user interface is employed as the operator interaction. By applying the object-oriented approaches to the design and the implementation stages, the system is highly configurable, easily modifiable, and extensible to more sophisticated one by compiling the operation experiences.

Acknowledgments

We would like to thank to Dr. N. Shikazono, Dr. Y. Kawarasaki, Dr. Y. Suzuki, and the laboratory staffs for the helpful discussions and the strong interests on the software development.

References

[1] M. Sugimoto, et al., "Progress of the IR FEL development at JAERI", Nucl. Instr. and Meth. A 341, ABS 41 (1994).

[2] M. Sugimoto, "Control system with a software simulator for JAERI-FEL", Nucl. Instr. and Meth. A 331, 340 (1993).

[3] M. Sugimoto and M. Takao, "Beam transport program for FEL project", Nucl. Instr. and Meth. A 318 335 (1992).