PB20

Analysis of the PLC Object Model in the PF Linac

Isamu Abe, Akihiro Shirakawa, Kazuo Nakahara and Masahiko Tanaka

KEK, National laboratory for High Energy Physics
Oho1-1, Tsukuba-shi, Ibaraki 305, Japan

Abstract

Device controllers (front end) for the KEK PF Linac, which has been operated for more than ten years, had to be renewed, because there is no longer support from hardware manufacturer. To reduce maintenance costs, device controllers are planed to be replaced by PLC (Programmable Logic Controller). Investigations launched in the late 1994s for the PLC [1] have carried out in order to make sure of the I/O stability and network possibilities. In this paper, the software for the PLC is discussed based on the OOP concept.

1. Introduction

An Object oriented approach to a PLC control(device control) system analysis and design was adapted to the Linac. The old console system has a graphic display written using FBHG (Fujitsu High grade modular Basic). It has not been very simple to change the display and it's programs, which were written in structured FBHG. To change the conventional language to the new OOP language is a big paradigm shift, and there are two aspects which are good and not good. One aspect is high productivity at OOP, though more learning cost for users is needed. We are therefore providing a top class which can inherit objects for users without any programming on the GUI windows. Users can make a control and display windows very easily and quickly by just copying a visual object from it's mothers class inheriting method and properties. After naming the object and changing some properties if necessary, it can be run as a user's program.

2. About the PLC

The performances of the PLC is almost good enough for controlling the magnet power supplies, vacuum systems and klystron modulators as a slow speed control. Mostly, Since PLC can be connected to Ethernet these days, it is now possible to be controlled from computers, even though each maker has its own network. In the PF linac, the stability was actually measured and compared with the products of a couple of makers. Although the PLC is not for a high-speed multi-purpose machine interface, DAC and ADC modules are stable enough for magnet control. The PLC commercial product has been commonly used in chemical plants and other wide fields. The price is cheaper than other interfaces, like VME or CAMAC, and maintenance is more easy.

3. PLC networking and speed

The PLC which we have selected has two CPUs: one is for I/O, and the other is for communications. TCP/IP and UDP are available protocol used to communicate with other computers on Ethernet. The PLC has its own programming and tools inside, and allows itself to be an intelligent controller using a ladder program. For all channel scans on MS windows3.1 with a pentium 90MHz CPU, 10 to 20 refreshes are proceeded. That is a sufficient scan speed for an operator when he adjusts the magnet current or other. In the system of the PLC and host computers, the PLC is the bottle neck at communications specified in a data/command transfer. The host computer has several device-control network segments, which are managed in serial; there would be no collisions in it's communication packet.

4. GUI object model

The common classes for linac control were analyzed and implemented using MS Visual C++ on Visual BASIC based on OMT (Object Making Technique) [2,3]. In the accelerator domain it is possible to be defined in two major classes as the device class and the generic class. The device class may have several derived sub-classes; the other super classes, like communications, components and other related classes. These classes were developed as a control pack which appear on the toolbox for Visual Basic when it is in the interpreter mode.

The basic model of top classes and it's relations are shown in figure-1.

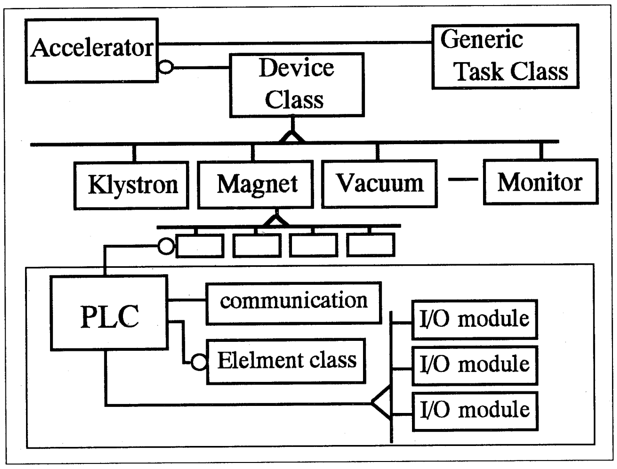

figure-1

5. Device class of the Linac

Device class; klystron, magnet, vaccum, gun, monitor and others are defined and coded as a super class on the control pack. The instances (real object) are derived in the GUI windows from its control pack: for example, the magnet class has its sub class (such a Q-magnet, STC-magnet and Bending-magnet). Each device class may use a PLC object for communicating between them. It is defined so that the PLC class is related the other super class.

6. PLC class

PLCs are produced by several companies, and there are no standards among the manufactures. A PLC support many I/O modules: such as DI/O, DAC, ADC and IRQ-module. The PLC is the super class, and the I/O modules are defined as belonging to the sub-class. The PLC top class can communicate to the PLC communication class. These PLC classes should be installed on the device layer. In the PF Linac, device classes are set on the Human interface layer, and the device class of the GUI should communicate with PLC device class.

7. Implementation

It is being planned to install the PLC for the Linac device controller, which will be connected with about 400 sets of magnets and 200 sets of vacuum systems and 60 sets of klystrons which are slow devices in control. In October 1995, a prototype is going to control the vacuum system at the 2nd sector of the Linac. In advance to the hardware system, software development has completed it's first version based on the Object models shown in the OMT. Since Visual BASIC has a simple inheritance system, a Mother class has been coded using visual C++, and was made as *.VBX tool on Visual BASIC. Visual BASIC has limitations in memory size and execution time when there are many objects derived from the mother class. PLC and other objects are shown in figure-2.

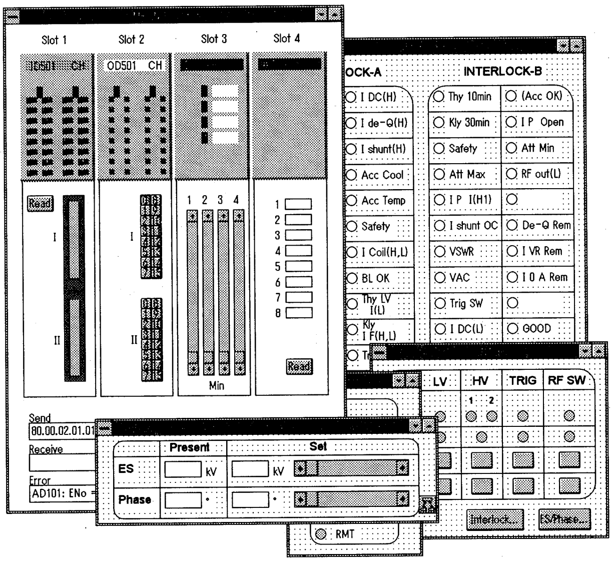

figure-2

8. Results and conclusions

Sufficient speed to control the magnet current setting was achieved based on Ethernet communications between PC and PLC. There are a reasonable number of PLC connections we can run in series because of PLC communications speed. We could actually make standard class for the PLC and GUI device object or class. The standard class brings out a nice reusable object, which makes the control system very flexible. OOP also gives users an easy way to make modification and to produce a highly productive control system for accelerator users or machine operators.

references

[1] Feasibility study for PLC as a device controller, A. Shirakawa
Linac conference 1995, Osaka

[2] GUI object model in Accelerator control
I.abe, K.Nakahara, M.Mutoh, Y.Shibasaki
Linac conference 1995, Osaka

[3] Object-Oriented Modeling and Design
James Rumbaugh. Prentice Hall, Inc.