

# The Control Software Based on Windows

Gao Weizheng, Huang Xinmin

Institute of Modern Physics, Chinese Academy of Sciences

P.O.Box 31, Lanzhou 730000,China

## 1 Introduction

This year, the HIRFL (Heavy Ion Research Facility in Lanzhou) control system will be transferred from VAX to PC, running under Windows. In this paper, we introduce some new technologies for developing the control system software.

## 2 The OOP method

The C++ language is used as a development tool in the kernel of our software. C++ provides a new object-oriented programming method. It deals with everything as whole subject, and abstracts into a data type, named as class. The characteristics that belong to the object are described. It gives some functions that can be done by this class of object, and some properties that inherit from its parents or are inherited by its children. The abstracted mankind, for example, has some characteristics, such as name, height, weight, age and so on, and can eat, walk, sleep and work.. If we define a class named as Mankind in our programs and realize all functions that should be done by a man, when we attach some characteristics to this class we generate a man in our software. Everything in our programs can be generated by defining a class. If we can look into a object, abstract it as a class, and this class is properly written, a object of this class can be easy generated . In the OOP view, the programs will be assembled by many classes , just like a machine is assembled by many components.

We see that the object-oriented program has many advantages. First, the analysis and implementation of a program is easier. The analysis is important for designing of a large program. Second, the program debugging is easy, because the program is assembled by classes. We can look for the bugs in each class, so that much time can be saved in the program debugging period. Third, the program's code can be reused since the code is used by all the objects of this class, and also by objects of a relative class that is derived from this class.

The object oriented programming method has been introduced into our control software designing. The accelerator is composed of many components. The components that will affect beam quality are our control objects. We abstract those components into classes. Take the power supply as example, we have defined a class that manages all power supplies.

Many power supplies are used in accelerator most systems. A power supply has some attributes, such as the input current value, the maximum value that can be bear, the status and the site in an accelerator (we always label it by its location). The power supply can change its current value according to the command given by the operator,

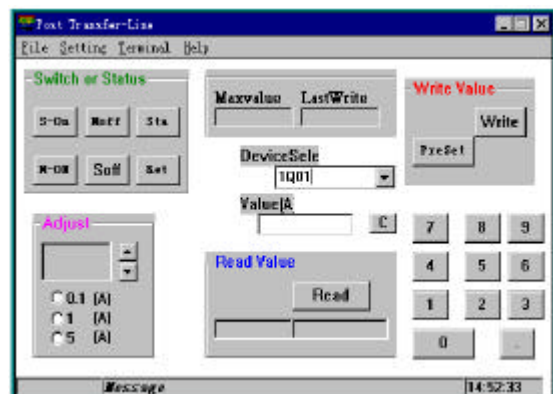
report its current value to operator, or report its status to an operator. So we can define a PowerClass as follows (in C++ language):

```
Class PowerClass:Public CamacAccess
{
    protected:
        char        PowerName[8];
        unsigned    PowerNo;
        float       WriteValue;
        float       ReadValue;
        int         Status;
        float       MaxValue;
    Public:
        PowerClass();
        bool WriteData(float value);
        float ReadData();
        int ReadStatus();
};
```

There are hundreds of power supplies in the accelerator system. If the PowerClass is properly written , much time can be saved in software development.

## 3 The windows program designing

The program that runs under Windows is more complex than DOS. Developing Windows program should follow the Windows rules. Visual Basic was chosen to develop the man-machine interface in our control system. Visual Basic is a powerful Windows program develop tool. It orients the object and provides many control components for the programmer. If you design a Windows interface, the work you need to do is pasting control components from its toolbox. We developed an interface as follows:



At first, we selected a device from the "DeviceSele box", then read the device's value, while will display in the box below the "Read" button, or write the value into this device by pushing the "Write" button. The Visual Basic can call

some functions written in C++ through the DLL (dynamic link library). A function, called through the DLL, should be declared as follows:

```
[Public|Private] Declare Sub|Function name Lib "libname"  
[Alias "aliasname"][(arglist)]
```

In this module, we call the functions, named as "write current" and "read current", through the DLL. The declaration is:

```
Private Declare Function hsy_wrt% Lib "\Power\hsy.dll"  
(Byval number%, Byval current&)
```

In the C module, we should declare a library entry that is LibMain() in win16.

Visual Basic can also communicate with other programs by DDE (Dynamic Data Exchange). The Windows DDE is managed by the DDEML (DDE Manage Library). If you want use DDE to exchange data among some programs, you should obey the rules defined by Windows. Only some examples are given in this paper.

The power supply's status is monitored by running a program written in Borland C++. If a power supply status is changed, in other words, it has some problems, the monitoring program will notify the control software (written in VB) by DDE. In Visual Basic, some control components have DDE properties, such as Static box and Edit box. We use the Static box as our DDE box. It has four properties used to DDE: LinkItem, LinkMode, LinkTimeout, LinkTopic. In the initiation of the program, we set the four properties. The LinkItem and LinkTopic must be the same as defined in the monitor program. In a C

subroutine, we define a procedure to deal with the DDE messages. The procedure should be declared like this:  
HDDDEDATA EXPENTRY \_export DDECallback (WORD wType, WORD wFmt, HCONV hConv, HSZ hsz1, HSZ hsz2, HDDDEDATA hData, DWORD dwData1, DWORD dwData2);

The message sent to DDE is XTYP\_CONNECT, XTYP\_REQUEST, XTYP\_EXECUTE, XTYP\_ADVSTART, XTYP\_ADVREQ and XTYP\_XXX. This procedure is loaded by calling DdeInitialize function. We can also call some DDE functions provided by Windows to deal with some DDE events.

#### 4 Summary

At present, the programs developed by us that run under Windows are in a test stage. We have just finished a part of a control system. There are many hard tasks to do in the next step. In the future, we want to transfer those programs to a PC network.

#### References

- [1] <<Microsoft Windows 3.1 Programmer's Reference>> Microsoft Corporation
- [2] <<Borland C++ 3.1 object-oriented Programming>> Ted Faison
- [3] <<Visual Basic 3.0 for Windows Programming Guide>> Zheng Qinghua