The Design and Implementation of the HIRFL- Cryopump Temperature Monitoring System

Li-li Li¹, Yan-yu WANG¹, Jia YIN¹ and Jun YIN¹

¹Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou, China

Keywords: HIRFL, Temperature monitoring, Protocol converter.

Abstract. This paper mainly presents a HIRFL-Cryopump temperature monitoring system by describing the architecture of the whole hardware system and the designing of its software system. We use a self-designed protocol converter as a bridge to connect to a vacuum device. This converter can convert serial port signals to network signals which we need in the central control room. The software part is created in ForceControl configuration platform, so the driver development is one of the most important jobs in the software design process. The driver codes are debugged and passed in Visual C++. Through two years running, it shows that the driver is correct and efficient. Overall, we believe that this system provides a cost-effective monitoring service with acceptable user experience.

Introduction

HIRFL is a large-scale heavy ions research facility, which can accelerate various ions and possess high energy. It is comprised of the ECR ion source, Sector Focusing cyclotron (SFC), Separation Sector cyclotron (SSC), radioactive ion beam line, Cooling Storage Ring(CSR) and so on. As one of the most important devices in HIRFL, the vacuum unit can maintain high vacuum condition inside the accelerator and ensure the HIRFL running effectively. Cryopump as an important equipment can create vacuum and have been widely applied as a reliable tools in ECR, SFC and SSC [1]. In the whole running time, a large number of experiments proceed in their proper order. Staff can’t attend to check whether the cryopumps work successfully. So a cryopump temperature remote monitoring system is urgently needed.

In this system we can obtain the real-time data from the field device via the Intranet in the central control room. We can also monitor the temperature curve and query the historical data through the remote monitoring. When the temperature is anomalous, this system can generate alarm signals and store some vital messages in the database so as to query them at any time. After practical running for two years, this system had shown its accuracy and reliability.

Architecture of the Whole Hardware System

The Model 218 is an eight input temperature monitor that can be used with diode or resistive temperature sensors. The measurement input was designed for the demands of cryogenic temperature measurement. The Model 218 has eight constant current sources (one for each input) that can be configured for a variety of sensors. The inputs can be configured from the front panel or via a computer interface, and are grouped in two sets of four. Each set of four inputs is configured for the same sensor type (i.e. all 100 Ω Platinum or all Silicon Diodes, etc.).

The Model 218 has a 9 pin D-Subminiature plug on the rear panel for serial communication [2]. The serial interface used in the Model 218 is commonly referred to as an RS-232C interface. RS232C is a standard of the Electronics Industries Association (EIA) that describes one of the most common interfaces between computers and electronic equipments. In order to receive vital data in central control room via TCP/IP protocol, we must find a new way. A protocol converter is a nice solution. The protocol converter can convert the serial signal from the Model 218 to the network signal, and the LAN switch transmits the temperature data to the first end server.

As we mentioned, the self-designed protocol converter named LOW_TEMP_NET has its unique IP address and port number, and can achieve to the function that convert serial port (RS485, RS232 and RS422) signals to network signals. As Fig. 1 is the picture of LOW_TEMP_NET.
In this system, there are 29-point, including 18 cryopumps, need to be monitored. Their communication rate with Model 218 is 9600bps. The front end Server linked to LOW_TEMP_NET through LAN switch, send commands to the Model 218 and receive the temperature from all cryopumps. We can open the browser to get all monitoring data in central control room. The topology of this system is shown in Fig. 2.
Design of the Software System

For the software system, it uses ForceControl configuration software that developed by Sunway Corporation as the development platform. This is a set of professional software that aimed to collect real-time data from the field device and control the communication process. It can communicate with various I/O devices in the domestic and overseas at the same time, including Distributed Control System (DCS), Programmable Logic Controller (PLC), Field Bus Control System (FCS), USB, Intelligent Module, Board, Intelligent Instrument and Variable-frequency Drive, etc [3].

In addition, it provides a FIOS Software Development Kit (SDK) in 32-bit Windows environment and allows users to develop a variety of I/O device drivers under the ForceControl software platform. We just need to write several implementation codes of the scanning function relying on the specific communication protocol or the driving interface instruction, and then proceed to do debugging and testing. It is the most important job in our system to develop new I/O device driver because of lack of a useful one.

FIOS SDK is mainly comprised of 4 files (named Iodevcfg, Ioitemui, Ioapi and lserver) [4]. It encapsulates lots of technical details which developer needn’t care about. Refer to the communication protocol of the Model 218, we write several codes in Visual C++ and debug and pass. In the next paragraph it is the parts of the codes.

```c
INT OnReadData(CPacket* pPacket, LPTSTR lpszSendString, INT& nCmdLen)
{
    CDevice* pDevice = pPacket->GetDevice();
    CChannel* pChannel = pDevice->GetChannel();
    pChannel->ClearAcceptBuffer();//clear the buffer
    IOITEMDEF* pItemStru = pPacket->GetItem(0)->GetItemStru();
    //create read command
    nCmdLen = 0;
    lpszSendString[nCmdLen++] = 0x4B;
    lpszSendString[nCmdLen++] = 0x52;
    lpszSendString[nCmdLen++] = 0x44;
    lpszSendString[nCmdLen++] = 0x47;
    lpszSendString[nCmdLen++] = 0x3F;
    lpszSendString[nCmdLen++] = 0x0D;
    lpszSendString[nCmdLen++] = 0x0A;
    return SEND|WAIT;
}

BOOL ChangeASCII(CDevice* pDevice, char* chResponse, int nResponLen)
{
    BYTE bChar;
    for(int i = 0; i < nResponLen - 1; i++)
    {
        bChar = BYTE(chResponse[i])&0x7f;
        chResponse[i]=bChar;
    }
    return TRUE;
}
```
The GUI software on console is shown in Fig. 3. When the temperature value detected by the software is not in threshold, the corresponding alarm light will blink with ringing to warn the on-duty person. At the same time, the alarm message will be shown in the Alarm Message interface. The GUI software of this system also shows the real-time temperature curve in the T-Curve interface so as to observe the difference in one hour directly. The history temperature values and warning messages will be stored in the database so that we can inquire them at any time.

Conclusions

Through two years practical running, it shows that the system has good stability and accuracy, completes to monitoring the HIRFL-Cryopump Temperature successfully and meshes with vacuum department smoothly. Due to the versatility and flexibility of the self-designed protocol converter, it can also been applied to many other systems, including the dipole magnets temperature monitoring and interlocking protection system of HIRFL-CSRm and so on.

References

[1] Wu Dezhong, Zhou Lijuan. The Design and the Thermal Entropy of the New HIRFL-800 Cryopump. CRYOGENICS.2002.3. [In Chinese]

[2] 218_Manual.pdf, LakeShore, USA, 2004.

[3] Database and Web Development Manual. ForceControlV7.0. In Chinese.

[4] http://www.sunwayland.com.cn.