DEVELOPMENT OF HANDY TERMINAL USING PDA

Y.Ishizawa*, M.Takeuchi, T.Ohata, T.Fukui,
SPring-8, Hyogo 679-5198, Japan

Abstract

In SPring-8, VME-based distributed control system is used to realize stability, scalability and versatility. When the network is stopped at the time of maintenance, various equipments cannot be controlled from remote terminals. We planned to develop a handy terminal as a local controller. The handy terminal requires a compact size, long buttery backup time and user-friendly interfaces. We selected a Linux-based PDA, SharpZaurusSL-C860 (Zaurus). The MADOCA (Message and Database Oriented Control Architecture), which is a control software framework developed in SPring-8, was ported on it. The network technology that implements IP protocol to the physical layer of USB, usb-net, is adopted, which provides a peer-to-peer network communication between the VME system and the Zaurus, even if the network stops. We made a GUI using Qt/Embedded to fit local operations for maintenance and set up. We completed the PDA controller system with which we made Beamline component maintenance in January in 2005.

INTRODUCTION

A classical specification of communication interface, GP-IB, had been used at many facilities. However, it is not suitable for the infrastructure at a synchrotron radiation facility, because GP-IB is unstable on continuous operation and slow data transfer speed. In SPring-8, we discarded GPIB motor controllers. When the network is stopped at the time of maintenance, we cannot control many motors. Because we lost the local operation panel which is located on a front of stepping motor controller with GP-IB interface. We need a replacement of the control panel for the time of maintenance. For this purpose, we selected local controller with portable and ease of use. Because local controller is used for long time in the place where like athletic field.

As the other requirements, identification of equipments is also important. In SPring-8, nearly 50 beamlines are under operation. The identification prevents from the erroneous operation to wrong equipments, because many similar equipments can be found in a large facility. A workstation which is designated for a beamline has an ability of authenticate with an access control list. The access control list is specified for a beamline. A handy terminal is not specified for a beamline and needed to use for various equipments. We selected a peer-to-peer communication between a controller and a VME. In this paper, we would like to present porting of the MADOCA[1], is the software framework of the SPring-8 control system into PDA that is driven by Linux and also show the other development of Ethernet over USB and Qt GUI.

REPLACEMENT OF A LEGACY INTERFACE

In SPring-8, the beamlines are made up various experiments and are being conducted to each purpose and mission. Equipments had a lot of stepping-motor controlled by a 16-axes stepping-motor controller, PM16C [2]. A PM16C has local control panel and can be controlled by a GP-IB. Figure 1 shows a schematic view of old control environment for stepping-motors. This system is working well for tabletop experiment. However, GP-IB has not enough stability for continuous operation.

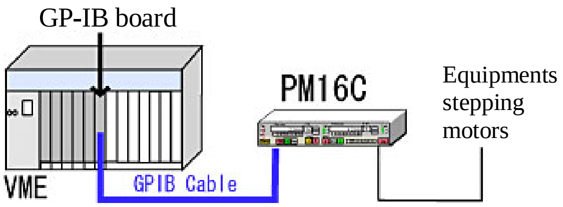

Figure 1: Schematic view of old former control system for stepping motors.

We replaced numerous GP-IB board and PM16C to VMEbus board stepping motor controller, Advme2005 [3] for stable operation. Advme2005 has historical results in our control system. However Advme2005 has no local control panel and we lost local control for maintenance of stepping motors. Figure 2 shows a schematic view of new control environment for stepping-motors.

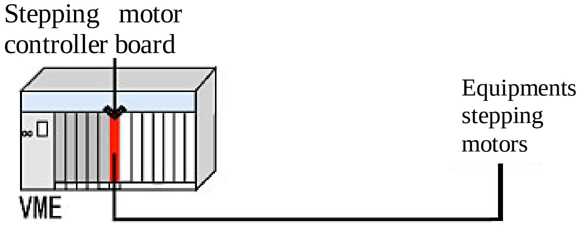

Figure 2: Schematic view of new former control system for stepping motors.

Most of the equipment in SPring-8 can be maintained at shutdown periods. The control system such as network should also be maintained at this time. When network system is out of service, it is not possible to maintain any equipment. We need a local controller that is free from network system and fit in well with existing control system. We develop local controller to maintain various equipments.

*yasu@spring8.or.jp

REQUIREMENTS FOR A CONTROLLER

For the maintenance work, one-handed operation, large screen size, toughness and long battery life are required. The local controller is used in the place where like athletic field so that it need one-handed operation and large screen to display a large amount of information. The local controller must be able to operate during the working hour of 7 hours. As a result, we selected the Sharp Zaurus SL-C860 (Zaurus) [4], which is operated by Linux OS(2.4.x). Zaurus is powered by Intel XScale PXA255/400MHz processor with 64 MB random access memory. And 3.7inch size, 65536 colors LCD with 640x480 resolution is supplied. It has USB port, SD and Compact Flash extension card slot. Zaurus has functionality of IP over USB.

PORTING MADOCA TO ZAURUS

MADOCA is a control framework which was originally developed in SPring-8. MADOCA is portable and scalable to various UNIX OS. The MADOCA consists of Message Server (MS), Access Server (AS) and Equipment Manager (EM). Figure 3 shows the diagrams of MADOCA. MS uses System V IPC, message. GUI application sends control messages to MS. MS sends messages to the appropriate AS. AS is client process of ONC/RPC. AS sends messages to the Server process (EM) running on each VME system. EM operates the VME boards according to the received message and replies the execution result to the client. AS returns the result to MS. MS returns to GUI application.

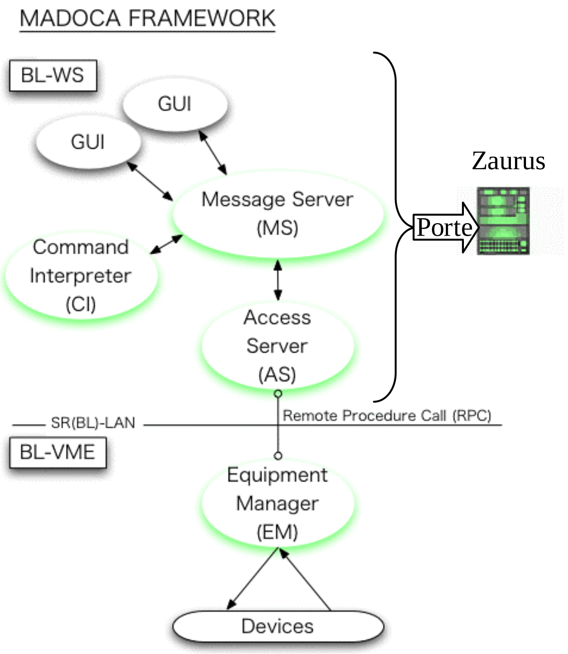

Figure 3: Framework of MADOCA.

At first, we built up the development environment for Zaurus on RedHat Linux.7.3 For existing VME system assets, we ported MS and AS to Zaurus, developed GUI application by Zaurus.

DEVELOPMENT OF ETHERNET OVER USB

VME system has several communication ports such as USB, Ethernet and RS-232C. USB and Ethernet are better by reason of the communication speed. Wireless LAN has security problem. The network to communicate between Zaurus and VME systems should connect peer-to-peer so that it can protect from the erroneous operation by miss understanding. Because there are lots of similar component and have similar name in beamline of SPring-8. Therefore we selected the USB for a physical layer of the network.

We developed a device driver of Ethernet over USB (usb-net) on Solaris9 (x86) operating system. The usbnet is able to communicate between Zaurus and VME systems by using ONC/RPC, which is required by MADOCA. The usb-net consists of two modules, USBnet module and USBzaurus module. The USBnet module transforms Ethernet packets to data frames of USB. And the USBzaurus module implements the hot-plug capability on Solaris9. A layer structure of usbnet is shown in figure 4. These modules are located in the Data Link layer in the OSI model. Because the USBnet module has an interface of DLPI (Data Link Provider Interface) version2, any application program can communicate with Zaurus without any change. Solaris provides a skeleton device driver of DLPI, called GLD (Generic LAN Driver). GLD is a multi-threaded, clonable, loadable kernel module providing for Solaris local area network (LAN) device drivers. We made the USBnet module based on GLD. The USBzaurus module communicates with USB core driver instead of any Ethernet devices. Connecting Zaurus and VME system by a USB cable, Solaris9 recognize Zaurus as a network node.

For a runtime and development environment of the usb-net, USB Driver Development Kit Version 0.9a (DDK) is needed. This DDK is distributed on the web site of Sun Microsystems. As a compiler, gcc3.4.1 was used.

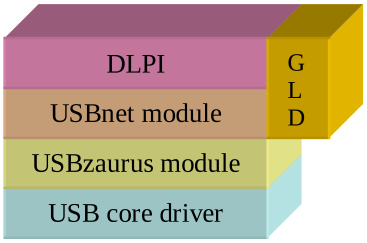

Figure 4: Structure of the driver.

DEVELOPMENT OF GUI

We made the cross-platform development environment on Linux (Redhat7.3). The tools of cross-platform environment are able to download from Sharp web site.[5] The tools that we used are as follows.

- gcc-cross-sa1100-2.95.2-0.i38 : Cross Compiler
- glibC-arm-2.2.2-0.i386 : Library
- linux-headers-arm-sa1100-2.4.6-3.i386 : Header File
- binutils-cross-arm-2.11.2-0.i386 : Utility

We developed two GUI application programs on the Zaurus suitable for our maintenance work. One is mstool. Other is GUI application for maintenance. We reconstructed the local controller panel image of PM16C on Zaurus as a GUI application programs. Figure 5 shows a screenshot of Zaurus. As a utility tool, we ported one of an application program, mstool, of MADOCA to Zaurus. The mstool can manage any kind of messages on MADOCA. Qt is adopted as a GUI application framework, which is implement in Zaurus. Qt is a cross-platform C++ GUI application framework developed by Trolltech [6] and enables to one-handed operation of Zaurus. Zaurus provided Qt and Java for GUI application program environment. We chose the Qt, because JAVA is located on Qt in Zaurus architecture. So, JAVA is very slow. As a GUI builder Qt Designer was used. Qt Designer is de facto standard GUI builder for Qt that is also prepared by Trolltech. Qt Designer generates XML-based codes that includes user interface information. We can build GUI applications program easily.

We used Zaurus to check setting of hardware limits and behaviours of equipments as maintenance work in March 2005.

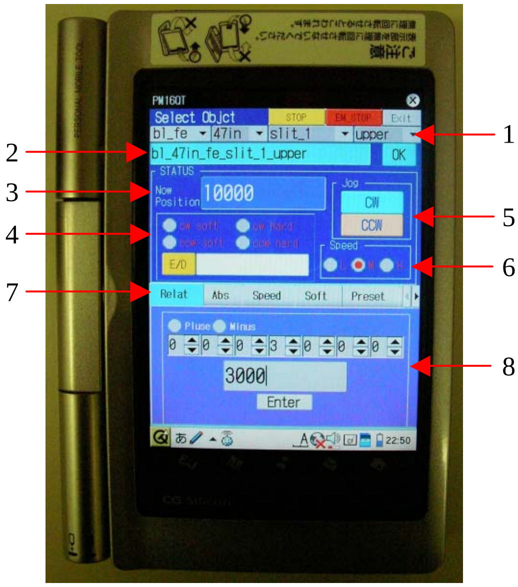

Figure 5: GUI application for maintenance.
1: Select the stepping motors.
2: Display MADOCA command.
3: Display position of stepping motors.
4: Display hard and soft ware limits status.
5: Operate up/down of stepping motor.
6: Select the speed.
7: Select the menu.
(relative, absolute, speed, soft limit, preset, etc.,)
8: Set the value.

SUMMARY

We developed a handy controller for various maintenances by using PDA. When the network is stopped at the time of maintenance, various equipments cannot be controlled from remote terminals. Because we reconstructed the local controller panel image of PM16C on Zaurus as a GUI application programs. We can replace unstable GPIB equipments to VMEbus stepping motor controller with PDA as local control panel. As a communication, we developed the device driver of Ethernet over USB on Solaris9 with plug and play capability. It makes easy connectivity between PDA and VME system without erroneous operation. The PDA provided local controllability, and reduces work load at equipment maintenance time in SPring-8.

REFERENCES

[1] R.Tanaka et al., “Control system of the SPring-8 Storage Ring”, Proc. Of ICALEPCS ’95, Chicago, USA, 1995, p.201.

[2] http://www.tsuji-denshi.co.jp

[3] http://www.advanet.co.jp/en/index.html

[4] http://www.sharp.co.jp

[5] http://developer.ezaurus.com/

[6] http://www.trolltech.com