

PC as a touch-terminal controller

Norihiko Kamikubota, Hirofumi Akimoto*, and Kazuro Furukawa
High Energy Accelerator Research Organization (KEK)
1-1 Oho, Tsukuba, Ibaraki, 305-0801, Japan
Mitsubishi Electric System and Service Co. Ltd. *
Umezono 2-8-8, Tsukuba, Ibaraki, 305-0045, Japan

Abstract

Touch-terminals and knobs have been used extensively at the KEK e-/e+ linac since its commissioning in 1982. They were connected directly to minicomputers through dedicated CAMAC modules. A new controller for a touch-terminal and a knob was developed with a DOS-based personal computer in the early 90s. A TCP/IP remote execution environment was installed to utilize resources of the linac control system. Tools for managing touch-sensitive areas were developed for easier maintenance. So far, this system has been used for the daily operation of the linac, which implies high reliability of a DOS-based PC with the TCP/IP protocol as an IO controller.

1 INTRODUCTION

The KEK e-/e+ injector-linac, with an energy of 8/3.5 GeV for electrons/positrons, provides beams to both the KEKB rings and the Photon Factory ring. Since the commissioning of the linac in 1982, five touch-sensitive terminals and knobs, which were installed at the control desks in the linac control room, have been used. These touch-terminals and knobs provide basic functions to control the linac devices with easy-to-understand graphic interfaces.

Since the early 90s these touch-terminals and knobs have been controlled by DOS-based personal computers (hereafter PC). A TCP/IP remote execution environment was installed to utilize resources of the linac control system. Roughly 50 screens have been developed and extensively used for the daily operation of the linac. Even in the recent Windows era, present touch-terminals and knobs are the operators' favorite.

In this article, first we describe the details of a PC controller for a touch-terminal and a knob (Section 2). A discussion follows in Section 3. In Section 3, the possibility to use old office PCs as controllers for touch-terminals and/or knobs is discussed. Finally, conclusions are given in Section 4.

2 PC CONTROLLER FOR TOUCH-TERMINAL AND KNOB

2.1 Short History

At the time of the linac commissioning in 1982, the control system consisted of eight minicomputers; two of them were in charge of controlling man-machine interfaces including the touch-terminals and the knobs [1]. The touch-terminals

and knobs were connected directly to two minicomputers through dedicated CAMAC modules (Figure 1). This old system has been used for roughly ten years.

By the end of the 1980s, the capacity of the minicomputers became inadequate for increasing control demands. Thus, we introduced two PC-based systems for man-machine interfaces. One is an operator's console system, which started around 1987 with DOS-based PCs and the MS-network [2]. Now, this system has been reconfigured with a modern Windows NT environment [3]. Another is a new touch-terminal system, which is described in this article. The development of this new touch-terminal system started around 1990. The first one was installed in 1991 [4]. By 1998, all five touch-terminal systems had been replaced with new ones.

Both the operator's console system and the touch-terminal system have been used continuously, even after the main control system was replaced with Unix-based workstations in 1993 [5].

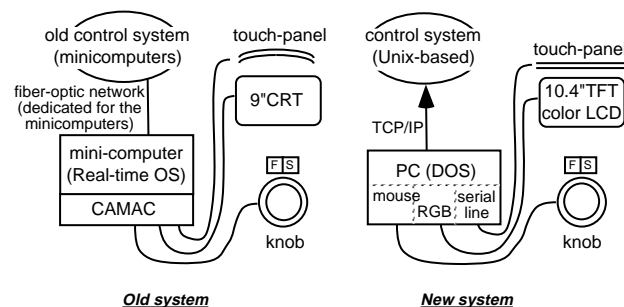


Figure 1: Schematic drawings of the old and new systems.

2.2 Hardware

As shown in the schematic drawing of Figure 1, the new system has the following upgrades: a) a 9-inch monochromatic CRT was replaced with 10.4-inch color-TFT LCD, b) controlled by a PC instead of a minicomputer, and c) communicate with the control system by the TCP/IP protocol rather than a dedicated network. A view of the system is given in Figure 2.

The following gives the detail descriptions for each part of the system:

(1) color display with a touch-sensor

A color-TFT LCD with a size of 10.4 inch is used for the

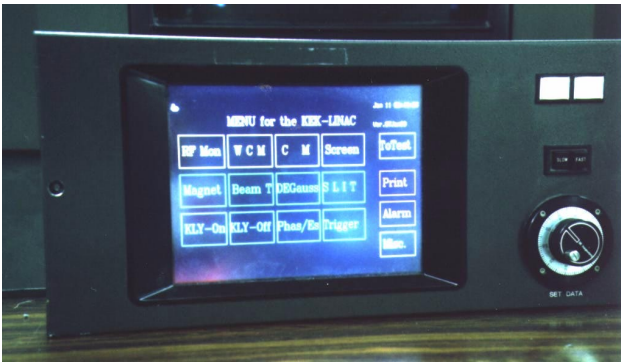


Figure 2: View of the new touch-terminal system.

new touch-terminal display. Since this type of display has been widely produced for PC notebooks, we could purchase them at lower cost. Actually, we introduced a commercial color-TFT LCD with a touch-sensor [6]. The resolution of the touch-sensor is 1024x1024, which is high enough for our purpose. The touch-sensor is connected to a PC by a serial-line.

(2) knob

A commercial knob [7] has been used for both the old and the new system. The signals from a knob is fed into a mouse-port of a PC through a simple differential circuit.¹ This idea has two advantages: a) popular mouse-functions can be used to control the knob signals, and b) no additional card is needed.

(3) personal computer

The touch-terminal controller is a popular PC (NEC PC available in Japan). A typical one has a 80486 chip at a 66MHz clock, and 12MB memory (used as a RAM-disk). An ethernet card is necessary for communication with the control system, but no hard disk is needed. The PC runs with the DOS operating system (V3.3) and the PC/TCP [8] environment.

2.3 Touch-zone Control Functions and Tools

(1) overview

Various functions and tools for controlling touch-zones were developed with Quick C (V2.0). These tools (or a C-program with the touch-zone functions) are invoked from a DOS-style batch program. An example of batch program is given below. This batch program produces the screen shown in Figure 3.

```

_tzload layout.txt layout.dat -i
_tzsel layout.txt layout.dat -vABC
if errorlevel 68 goto UNDEF
if errorlevel 67 goto QUIT
if errorlevel 66 goto LABEL_B
if errorlevel 65 goto LABEL_A
goto UNDEF
:LABEL_A
  # do some specific action for Zone A
  goto QUIT
:LABEL_B

```

¹Only X-axis is used to accept a knob signal. Right-button is assigned for the radio-button which enables faster encoding.

```

  # do some specific action for Zone B
  goto QUIT
:UNDEF
  # show error message
:QUIT

```

(2) drawing screen

Usually, screen drawing is made by referring two files (a layout-text and a layout-data). The tool '_tzload' converts them to a graphic presentation (see Figure 3).

(3) touch action

When an operator touches one of the valid touch-zones (in the present case, zone A or B or C), the tool '_tzsel' works to execute a specified action. For example, a touch to the zone C will terminate the batch program.

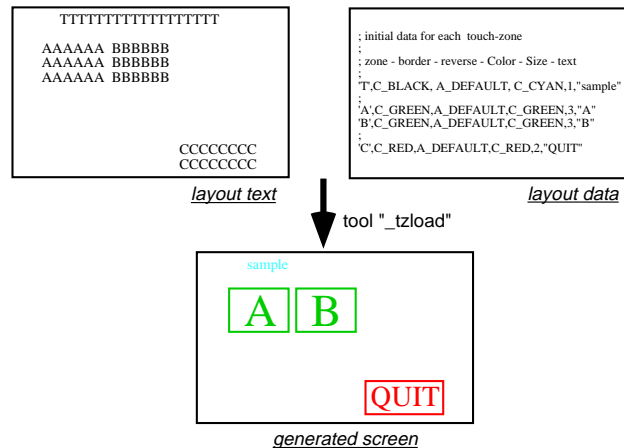


Figure 3: The '_tzload' creates a drawing by referring two files.

2.4 Operation and Administration

Up to now, roughly 50 screens (50 batch programs) have been developed. They are frequently used in the daily operation of the linac. Figure 4 is an example of the screens.

One PC is in charge of program maintenance. All of the programs (batch or C) are modified (or newly created) with this PC, then stored at a RAID-disk of the linac control system. When a touch-terminal PC starts, it downloads the newest version from the RAID-disk to a local RAM-disk automatically.

3 DISCUSSION

3.1 Long-term Reliability

Two of five PCs, which we bought before 1991, frequently cause sudden stops (i.e. once a day). Experimental studies have suggested that an interrupt from a serial line (corresponds to an operator's touch) is sometimes misunderstood, for example, as an interrupt from a hard-disk. Later troublesome PCs were replaced with others. Such a miserable problem was never experienced with the PCs in later years.

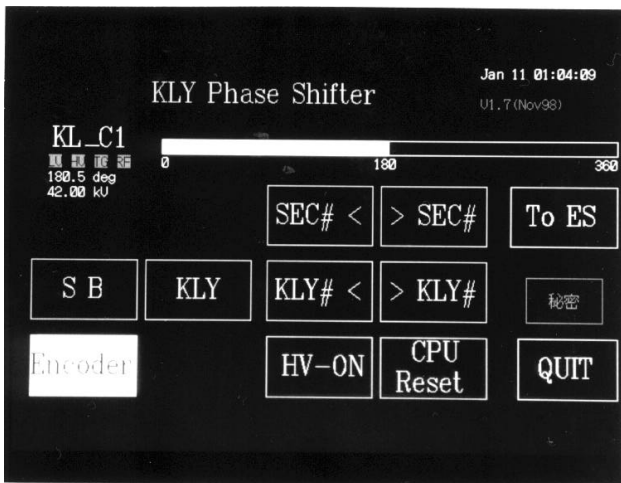


Figure 4: Screen to control the klystron phase.

After this initial problem, the PC controllers have been satisfactory stable over the past 8 years.

3.2 Performance and Problems with TCP/IP

At the beginning of development (around 1990), round-trip communication with the TCP/IP protocol always stopped at around 70000 loop counts. Later, this problem was assumed to be a bug in the libraries provided by a company. After we changed the library to [8], the problem disappeared.

The round-trip times between a PC and the linac control system were measured, and are summarized in Table 1. The observed times are short enough, since most of our applications expect for the refresh rate at a few Hz. Note that the popular clock-cycle of PC was 10 MHz when we started development, and 40 MHz is fast enough for our purpose. This fact enables us to buy a PC controller at an extremely low cost, since we do not need the latest high-end models.

In 1990–1991, we developed a network communication library useful for general data-exchange between Unix, VMS, and DOS [9]. This library was used in the development of touch-terminal applications. It is worth noting that the use of the library enabled us much easier development of TCP/IP applications at the PC.

Table 1: Observed round-trip times between a PC and the linac control system.

PC (year)	UDP	TCP
16MHz 80386 (1991)	30 ms	100 ms
40MHz 80486 (1994)	11 ms	18 ms

3.3 Use of Batch Program

The DOS-style batch is the only possible script-language for our PC-based controllers. One of the problems with the present system is that the changing-time from one screen to next is slow. With a PC with a 66 MHz clock, it takes 1–5 seconds, depending on the selected screen. This slowness originates from the slowness of the batch program at a DOS system. An easy (but non-elegant) solution is to introduce a PC with a faster clock. In addition, the DOS-style batch is not suitable for a large-scale program with many conditional branches.

3.4 Future Availability of DOS-based PCs

Recently, all of the available PCs are Windows-based. This implies that it will be difficult year by year to buy a DOS-based PC. Although the DOS operating system is expected to survive, we will suffer higher costs in the near future.

On the contrary, we will have many old PCs in the near future, which are currently used for office work. These PCs will be unable to be used with the latest Windows-based operating system. However, based on our experiences, they will be able to survive as the DOS-based controllers for touch-terminals and/or knobs.

4 CONCLUSION

A controller for a touch-terminal and a knob was developed with a DOS-based PC with the PC/TCP environment. They have been used in the daily operation of the KEK injector linac over the past 8 years. Though typical PCs have as slow clocks as 40–66 MHz, they are fast enough for our purposes. The PC controllers have been satisfactory stable, except for the initial problems. This fact implies that a DOS-based PC with the PC/TCP protocol has fairly high reliability as an IO controller.

5 REFERENCES

- [1] K. Nakahara, I. Abe, R.P. Bissonette, A. Enomoto, Y. Otake, T. Urano and J. Tanaka, Nucl. Instr. Meth., A251(1986)327
- [2] K. Nakahara, I. Abe, N. Kamikubota and K. Furukawa, Nucl. Instr. Meth. A293(1990)446
- [3] I. Abe, this workshop
- [4] N. Kamikubota, K. Furukawa and H. Akimoto, Proceedings of the 17th Linear Accelerator Meeting in Japan, 1992, p.276
- [5] N. Kamikubota, K. Furukawa, K. Nakahara and I. Abe, Nucl. Instr. Meth. A352(1994)131
- [6] ARGB-N41A-BK, San'oh Kogyo Co. Ltd., Mishima Japan
- [7] HPG-5J, Hagiwara Denki Co. Ltd., Nagoya Japan
- [8] PC/TCP Release 2.2 (Nov.1992), ftp Software, Inc., MA U.S.A.
- [9] N. Kamikubota, K. Furukawa, K. Nakahara and I. Abe, Proc. Int'l Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'91), Tsukuba Japan, November 1991, KEK Proceedings 92-15 (Dec.1992) p.318