

Utilisations of an Embedded System in a Physics Laboratory

Wattanapong Kurdthongmee

Lecturer, Division of Physics, Institute of Science

Walailak University, Tha-sa-la, Nakorn-si-thammarat 80160 Thailand

ABSTRACT - Majority of laboratory-designed instruments in an experimental Physics still rely on utilizations of standard electronic devices. The instruments absolutely perform with respect to the requirements of the design. They are, however, difficult to re-implement, replicate, maintain, and repair. In the last few years, many modern electronics devices have been commercialized with the benefits of smaller size, low power, low cost/performance ratio, and user-programmability. These devices are microcontrollers, CPLDs (Complex Programmable Logic Devices), FPGAs (Field Programmable Gate Arrays), and digital signal processors (DSPs). Unfortunately, the exploitation of these devices in Physics laboratory is limited. As part of our research, we have improved and/or redesigned the new laboratory instruments used both in teaching and research purposes to employ these modern electronic devices. In this paper we present the two instruments resulted from the utilizations of a microcontroller. These are an embedded system to automatically sample an experimental data and an embedded system to interface between a numerically displayed scientific instrument and a computer.

Keywords - microcontroller-based system, scientific instrument

บทคัดย่อ - โดยทั่วไปแล้วเครื่องมือทางวิทยาศาสตร์สำหรับใช้ในการทดลองทางฟิสิกส์ที่ถูกออกแบบขึ้นเองในห้องปฏิบัติการยังคงใช้อุปกรณ์อิเล็กทรอนิกส์มาตรฐาน เครื่องมือเหล่านั้นสามารถทำงานได้ตามความต้องการของการออกแบบ แต่การจัดสร้างเครื่องมือเพิ่มเติมตลอดจนการซ่อมแซมนั้นสามารถทำได้ค่อนข้างยาก ในช่วงระยะเวลาไม่นานมานี้ อุปกรณ์อิเล็กทรอนิกส์ชนิดใหม่ๆ ได้รับการวางตลาดด้วยจุดเด่นคือ มีขนาดเล็กลง ต้องการกำลังไฟฟ้าต่ำ ราคาถูกในขณะที่มีความสามารถในการทำงานสูง และยังสามารถในการโปรแกรมการทำงานได้เองโดยนักออกแบบระบบ อุปกรณ์เหล่านี้ได้แก่ ไมโครคอนโทรลเลอร์ ชิปวงจรรวมชนิด CPLD และ FPGA และชิปประมวลผลสัญญาณดิจิทัล อย่างไรก็ตามอุปกรณ์เหล่านี้แทบไม่ได้รับการนำมาประยุกต์ใช้ในห้องปฏิบัติการทางฟิสิกส์เลย ส่วนหนึ่งของงานวิจัยที่ผู้เขียนได้กระทำอยู่คือ การปรับปรุงประสิทธิภาพการทำงานและการออกแบบเครื่องมือทางวิทยาศาสตร์เพื่อใช้ในการทดลองทางฟิสิกส์ทั้งในการเรียนการสอนและงานวิจัย โดยมุ่งเน้นการนำเอาอุปกรณ์อิเล็กทรอนิกส์สมัยใหม่มาใช้งาน ในบทความนี้ ผู้เขียนจะได้นำเสนอผลงานส่วนหนึ่งที่นำเอาไมโครคอนโทรลเลอร์มาใช้ในการควบคุมการทำงานของเครื่องสุ่มอ่านข้อมูลทางวิทยาศาสตร์ และการใช้ไมโครคอนโทรลเลอร์ในการสร้างส่วนเชื่อมต่อระหว่างเครื่องมือวิทยาศาสตร์ที่แสดงผลเชิงตัวเลขกับคอมพิวเตอร์

คำสำคัญ - ระบบควบคุมโดยไมโครคอนโทรลเลอร์, เครื่องมือวิทยาศาสตร์

1. Introduction

Most of scientific laboratories, especially in experimental Physics which the author involves directly, rely heavily on utilizations of electronic instruments. The examples of such instruments are a data logger, a process controller, a radiation counter, a programmable logic controller (PLC), a thin-film thickness monitoring instrument, etc. While commercial scientific instruments have been constantly developed to keep up in pace with modern electronic devices, laboratory-designed instruments, in contrast, still rely on utilizations of standard electronic devices; i.e. standard TTL 74XX/74LSXX/74HCXX/74HCTXX or CMOS 4XXX families of digital devices and standard family of linear ICs. The instruments perform absolutely properly with respect to the requirement of researchers/experimentors. They are,

however, difficult to re-implement, replicate, maintain, repair and are very high power consumption. This arises from the fact that as the instruments are getting more and more complex as a result of the demand for higher functionality of the instruments, the circuit complexity also increases.

In the last few years, many modern electronics devices have been commercialized with the benefits of smaller size, low power consumption, low cost/performance ratio, and user-programmability. These devices are high performance microcontrollers, CPLDs (Complex Programmable Logic Devices), FPGAs (Field Programmable Gate Arrays), and digital signal processors (DSPs). Unfortunately, the exploitation of these devices in laboratories, both teaching and research, is limited. As part of our research, we have improved and/or redesigned the laboratory instruments used both in teaching and research purposes to exploit the benefits

of these modern electronic devices. In this paper we present two instruments which are the result from the utilizations of a microcontroller.

The organization of this paper is as follow. In section 2, we present a detail of an embedded system to automatically sample an experimental data. An embedded system designed to open a connectivity between a numerically displayed scientific instrument and a computer, another utilization, i

presented in section 3. Finally, section 4 concludes this paper and gives the future plan of our work. It is noted that in section 2 and 3, we focus more on the underlying ideas and algorithms to be used rather than specific applications. The given examples is only for demonstrating the proposed ideas. We hope that the readers may exploit these ideas in their own applications.

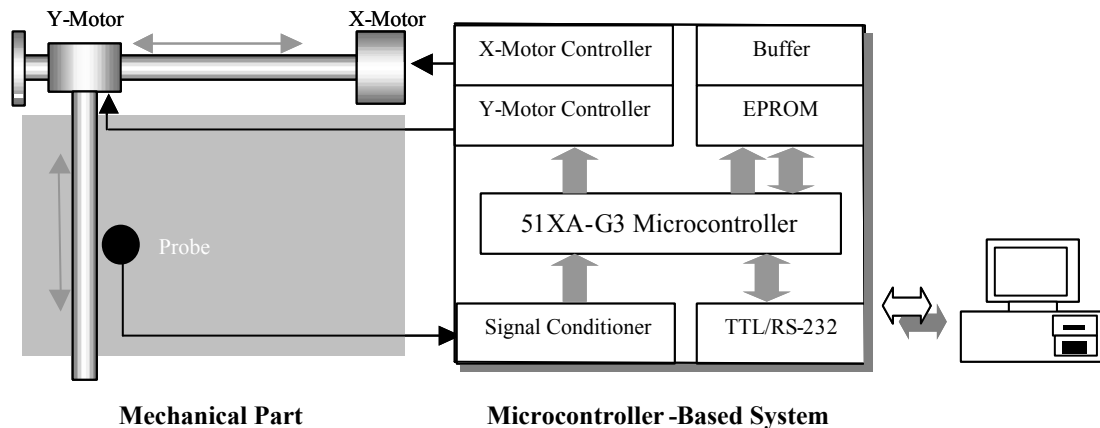


Figure 1. Illustrates the diagram of the microcontroller-based automatic sampling system

2. A Microcontroller-Based Sampling System

The motivation behind the design and implementation of our microcontroller-based sampling system came from the first year introductory physic laboratory: the experiment to study an electric field mapping [3, 4]. According to the experimental procedures [8], students have to prove the existence of electric fields in a medium by sampling values of electric potential from uniform grids with equal separations. These are then used to construct equipotential lines by connecting points with approximately equal potential values. These experimental procedures take time and only one configuration of electrodes can be performed by our students in a three-hour laboratory session.

Luckily, this limitation can be resolved by utilisation of a microcontroller-based system to control a mechanical part that is used to automatically sample potentials from a uniform grid within a medium. A fairly simple implementation of an automatic sampling system, with the capability to change grid separations was designed and constructed with an easy adaptation to another application in mind. In this case, the system, together with a moderate performance computer and a computer program, is successfully used to demonstrate experimental procedures to students.

The automatic sampling system consists of four main parts as illustrated in Figure 1: a mechanical part, a microcontroller-based system to control the mechanical parts, a data acquisition subsystem and an input/output subsystem. We

avoid describing the details of the latter two parts which are very specific to this experiment and only focus on the mechanical and microcontroller based system. These are detailed in the following subsections.

2.1 The Mechanical Part

Two stepping motors have been used as mechanical parts to precisely move a probe to locations to sample potentials. To change from a motion around the motors' axis to a linear motion along the x and y arms, a synchronous belt is used to connect between the rotating axis of a motor and a freely-rotating end at the opposite side of the arm. One end of the y-axis arm is attached firmly to the synchronous belt of the x-axis arm and is free to move back and forth along this axis. The y-axis arm is supported by an aluminium rod firmly fixed to the base of the system parallel to the x-axis arm in order to ensure that it does not oscillate during motion. The probe is attached to the y-axis arm at a location that allows it to move freely in the y-direction under control of the y-stepping motor. With this type of connection, one step of rotation, 1.8° , is changed to $1/256$ inch equally in both axis along the arms which are the accuracy of the sampling system.

2.2 Hardware and Software to Control Mechanical Parts

To make the stepping motors produce enough torque and rotate as smoothly and accurately as possible, a unipolar driving circuit [5] is utilised in our prototype. With this type of circuit, one end of each phase of a stepping motor is

connected to an open-collector switching power transistor while the other end is connected to the power supply. The power transistor is logically controlled by an output port of the microcontroller. As we use 4-phase stepping motors, we need 4 bits from the output port to control each transistor separately. In order to rotate stepping motors in a half-phase mode, the following 4-bit binary code sequences need to be sent out to the output port: 1001, 1000, 1010, 0010, 0110, 0100, 0101 and 0001.

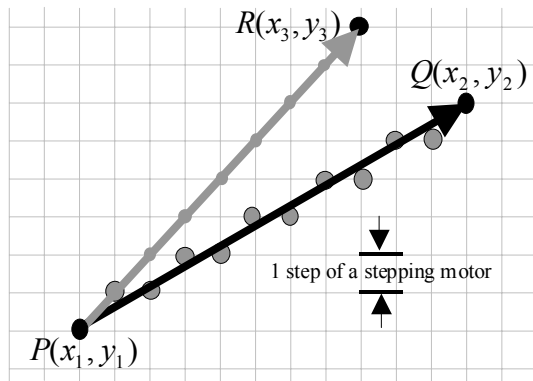


Figure 2. Illustrates Two ideal motion paths from $P(x_1, y_1)$ to $Q(x_2, y_2)$ and from $P(x_1, y_1)$ to $R(x_3, y_3)$

We employ a Brascenham's line algorithm [1] to use in controlling the stepping motors. The algorithm, which is actually a fundamental computer graphics algorithm to scan-convert straight lines on a raster screen, is an efficient method in that it uses only integer addition, subtraction and multiplication by 2 [6]. Therefore, it is fairly easy to code in an assembly language for use in a microcontroller-based control system.

The modified Brascenham algorithm [3, 4], shown in Figure 3, to control the stepping motors of our sampling system works as follows. Assume that we would like to move the probe from $P(x_1, y_1)$ to $Q(x_2, y_2)$ as illustrated in Figure 2. The best approximation to the ideal motion path is described by those coordinates in the discrete domain that fall the least distance from the ideal motion path. Call the distance to those abscissas lying above the ideal path NE_i and the distance to those directly below it E_i . The modified Brascenham's line algorithm identifies the decision variable, $d_i = NE_i - E_i$. When $d_i < 0$, the closest abscissa in the discrete domain will be the abscissa below the ideal motion path. Conversely, when $d_i \geq 0$, the abscissa immediately above the ideal motion path is closest. In figure 2, Two ideal motion paths from $P(x_1, y_1)$ to $Q(x_2, y_2)$ and from $P(x_1, y_1)$ to $R(x_3, y_3)$ are shown by solid lines. The gray dots are the intermediate positions of the probe under a condition of constant increment in the x-axis.

The technique provides an efficient method for moving the probe in directions less than or equal to 45° from the x-axis. For directions greater than 45° and less than 90° , the coordinate (x, y) must be interchanged prior to passing it to the procedure and must be interchanged again prior to moving the probe. The result of the interchange will cause just y or both x and y to be incremented at each step.

2.3 The Computer Software

A major role of the computer program to the sampling system is to define a group of coordinates to be sampled, group into a communication protocol and send to the automatic sampling system via a serial communication port.

The program divides the width and height of the sampling region into grids with equal separations along each axis at the required sampling resolution. The coordinate of each grid is coded by combining it with the protocol header and sent out to the sampling system. To avoid unnecessary waiting for incoming data to arrive, which is against the rules of Windows programming, we implemented the serial communication interrupt routine to accept the potential values from the sampling system. By this approach, the computer is free to do another job while data gathering is in progress. The incoming data is also in the form of a communication protocol and needs to be decoded to remove the protocol header from the potential value field. The decoded data is then stored in the two-dimensional array at the appropriate row and column.

2.4 Implementation and Result

All of the approaches and algorithms described in subsection 2.1 and 2.2 were implemented into the 51XA-G3 16-bit microcontroller system as shown in figure 5 by use of the 51XA-specific assembly language. The computer program described in subsection 2.3 were coded in Visual Basic.

The whole system has been successfully used in an introductory physics laboratory to sample potentials from different configurations of electrodes. The number of grid points, which is actually programmable, is set to 600 (30×20) with a grid separation of 5 mm. The times taken to sample potential values from all points in all experiments are approximately 5 min. Along with experimental demonstration to students, we also point out the benefit of exploitation of embedded system to physics experiments.

3. An Intelligent Unit to Interface a Computer to a Numerically Displayed Instrument

Most of scientific measurement instruments display data in a numeric form. Some of them, especially obsolete models, do not have a capability to interface to a computer. This factor limits the exploitation of such instruments in an experiment that must be performed automatically under computer-controlled. In this section, we present another approach to open a connectivity between a computer and a numerically displayed instrument. The benefit of this approach is that it can be done without interfering any complex parts inside the instrument. This approach has been successfully employed in order to open a connectivity between a radiation counter, used in nuclear physics laboratory, and a computer.

As our approach relies on a modification of a display unit of the instrument, in the following subsection, therefore, two configurations of this unit of the instrument are briefly described. Then, the interfacing techniques of both type of display unit are presented.

```

// An array to hold the logical codes, in 4-bit binary format, used to drive motors (send these codes to an output
// port to which the stepping motor connected.
STEP_TABLE : ARRAY(0..7) = (1001, 1000, 1010, 0010, 0110, 0100, 0101, 0001)
// Indicate which member of the STEP_TABLE is currently used to drive the stepping motor
CURRENT_STEP_X, CURRENT_STEP_Y : INTEGER
// Hold the current location (coordinate) of the probe with respect to the sample space
Xcurrent, Ycurrent : INTEGER
PROCEDURE MOVETO_XY(Xend, Yend : INTEGER)
BEGIN
    CURRENT_STEP_X := 0
    CURRENT_STEP_Y := 0
    dX := Xend - Xcurrent
    dY := Yend - Ycurrent
    incE := 2 * dY
    incNE := 2 * (dY - dX)
    d := incE - dX
    X := Xcurrent
    // X is a main axis of motion, loop until the probe has already moved to the end point at Xend
    WHILE X < Xend DO
        // No need to step in the Y direction
        IF d <= 0 THEN
            d := d + incE
        // Need to step in the Y direction
        ELSE
            d := d + incNE
            // STEPPING_MOTOR_Y is an address of the output port that controls the Y motor.
            OUTPUT(STEPPING_MOTOR_Y, STEP_TABLE[CURRENT_STEP_Y])
            // Update a member in the STEP_TABLE currently used to drive the Y motor.
            CURRENT_STEP_Y := CURRENT_STEP_Y + 1
            IF CURRENT_STEP_Y > 7 THEN
                CURRENT_STEP_Y := 0
            ENDIF
        ENDIF
        // STEPPING_MOTOR_X is an address of the output port that controls the X motor
        OUTPUT(STEPPING_MOTOR_X, STEP_TABLE[CURRENT_STEP_X])
        // Update a member in the STEP_TABLE currently used to drive the X motor.
        CURRENT_STEP_X := CURRENT_STEP_X + 1
        IF CURRENT_STEP_X > 7 THEN
            CURRENT_STEP_X := 0
        ENDIF
        X := X + 1
    ENDWHILE
    // Update the current location of the probe
    Xcurrent := Xend
    Ycurrent := Yend
END

```

Figure 3. Illustrates the modified Brasenham's algorithm to control the stepping motors in our sampling system

3.1 The Configurations of the Display Unit

The electronic circuits of nearly all laboratory instruments are complicate and need to be calibrated appropriately by the manufacturers. Moreover, the details for these circuits vary from one instrument to another. Even two instruments with similar functionality may have completely different electronic circuits. We have found that only the display unit of the instruments is unique and most suitable to be modified and extended in order to interface it with a computer. Broadly, the display unit can be classified into 2 categories: LED (light emitting diode) multiplexing digital display and direct-drive LED display [2].

In LED multiplexing configuration (see the shaded area in figure 4), the same segment pins from all digits of the LED are tied together and connected to the *segment bus* of a microcontroller/microprocessor or special function chips; i.e. 74C911, 74C912 or 74C926. The common pin of each LED is controlled digit by digit separately by a switching

transistor. To display a numeric value at any digits of the LED, all switching transistors are inactivated first. This causes every digit of the LEDs to display nothing. The segment data to be shown is then sent out through the segment bus and the switching transistor corresponding to the active digit are activated causing the numeric value to appear. Repeating the same procedures sequentially from the first to the last digit of the display unit at approximately 20 Hz of refresh rate causes all digits to display their numeric values continuously.

The display unit employing the direct-drive technique, on the other hand, uses as many BCD (Binary Coded Decimal) counters and BCD to 7-segment decoders as the required number of digits. Refer to the shaded area in figure 5, the output from a BCD counter is converted to a numeric form by a BCD to 7-segment decoder. The output from the decoder is, in turn, used to drive the corresponding digit of the LED directly. By this technique, every digit of the LED displays its own numeric value continuously.

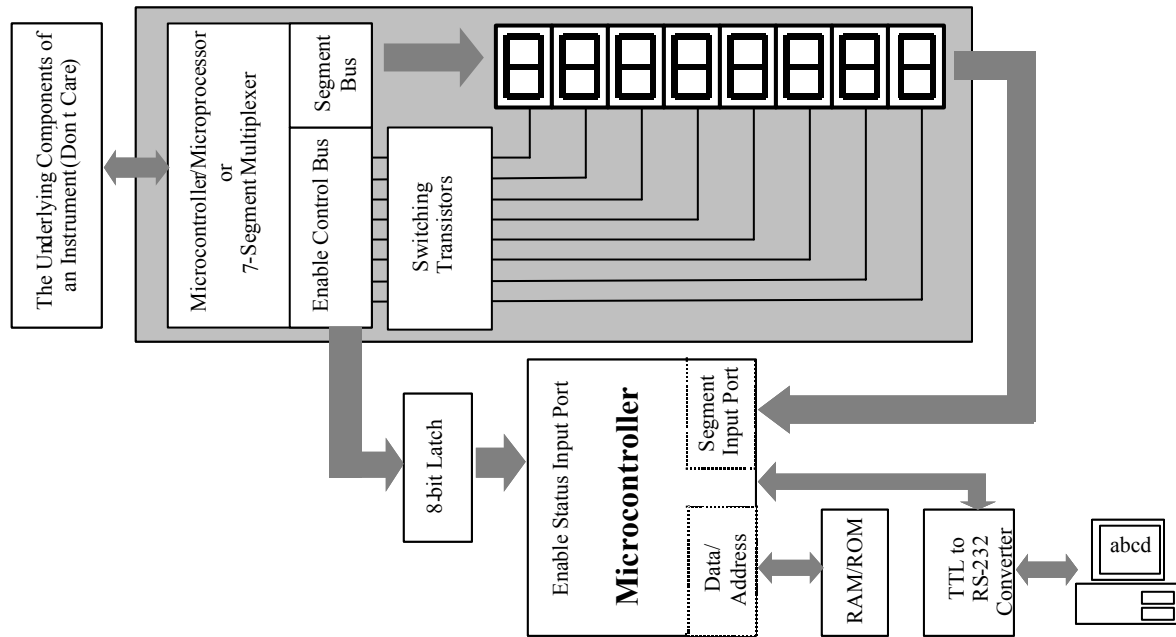


Figure 4. Illustrates The schematic of the instrument utilizing the LED multiplexing configuration and the modification.

3.2 The Modification Approaches

The most suitable electronic device to interface between a numerically displayed instrument, in both configurations of display unit, and a computer is a microcontroller. Employing this device makes the appearance of the interfacing circuit less complex as many components can be substituted by software. In addition, a microcontroller extends the functionalities of the system at no extra cost since many functionalities can be added at a software level. In this section, we present the details of the microcontroller based interfacing unit between the display component of an instrument and a computer in both configurations.

To automatically read data shown on the display unit of the instrument utilizing the LED multiplexing technique, the active digit at any instance of time must be known first. By examining the status of the enable signals, it is possible to point out to which digit the data at the segment bus is currently sent (only one digit of the LEDs is active at any instance of time). Then, the numeric data of the active digit is read from the segment bus and stored appropriately. Refer to the schematic in Figure 4, the enable signals from the enable control bus are stored in the 8-bit latch before connecting to the microcontroller. In order to avoid unnecessary hooking the microcontroller only to check, most of the time, unchanged state of the 8-bit latch, the microcontroller's timer interrupt technique is employed. The interruption causes the microcontroller to periodically identify the active digit by sampling the logical state from the enable control bus. During the interrupt process, the microcontroller samples data from the segment bus via the segment input port. This data, in 7-segment format, is converted to the corresponding BCD format and store in a suitable memory location.

The pseudo-code of the interrupt service routine described earlier is as follow.

DigitData : **ARRAY** [0..7] **OF** **BYTE**

// Determine the active digit, read the segment data and
// store in an appropriate location in the array DigitData

TIMER INTERRUPT readActiveDigit

// Read the current active digit from the 8-bit latch.

ActiveDigit := **INPORT**(8_BIT_LATCH)

// Use look-up table (LUT) to change to BCD format.

DigitIndex := **BINARY_TO_BCD**(ActiveDigit)

// Read seven-segment code from segment input port.

DigitDataSegmentCode :=

INPORT(SEGMENT_INPUT_PORT)

// Change from seven-segment code to the corresponding

// BCD and store in the array.

DigitData[DigitIndex] :=

SEGMENT_TO_BCD(DigitDataSegmentCode)

END INTERRUPT

Although, a radiation counter with the direct-drive LED display component appears to have fairly complicate circuit, the microcontroller interfacing system as shown in Figure 5 is easy to understand and implement. From the figure, the BCD signals from every digit of the BCD counters are input to the 8-to-1 multiplexer. There are four multiplexers in the schematic, each of them accepts only one bit of the BCD signals; D (2^3), C (2^2), B (2^1) and A (2^0). Each bit must be arranged in order from digit 1 to 8. The output signals from the multiplexers are, then, connected to the BCD input port of the microcontroller. The digit to be read into the

microcontroller's BCD input port is controlled by the selection signals.

In this configuration, the procedure to read data is trivial and can be implemented without utilizing any interrupt service routine. The pseudo-code of this procedure is shown below:

DigitData : **ARRAY** [0..7] **OF** **BYTE**

PROCEDURE ReadBCDData

// All digits of display

FOR digit := 0 **TO** 7

// Send out digit selection signal to the selection

// signal port of the microcontroller.

OUTPORT(SELECTION_SIGNAL_PORT, digit)

// Read BCD of selected digit.

dataIn := **INPORT**(BCD_INPUT_PORT)

// Store data in an array

DigitData[digit] := dataIn

END FOR

END PROCEDURE

Apart from acquiring data from the display unit and storing it in the array, the additional roles of the microcontroller in both configurations is to transmit these data to a computer via an appropriate communication channel; e.g. RS-232. Furthermore, the microcontroller can be programmed to partially/fully control the instrument. For example, the microcontroller in our sample implementation, a radiation counter interfacing system, has additional tasks to:

- release a command to the radiation counter in order to start gathering data at the specified, sampling time,
- release a command to stop gathering data and possibly reset the radiation counter,
- store data in its buffer if it detects that the computer is fail to receive data.

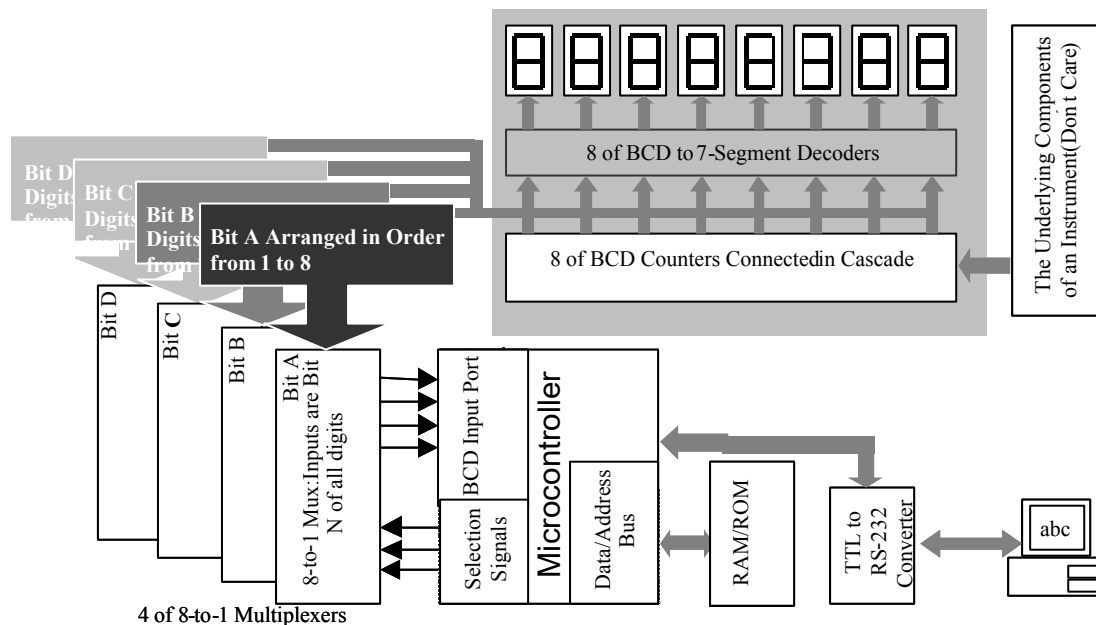


Figure 5. Illustrates the schematic of the instrument utilizing the direct-drive configuration and the modification.

3.3 Implementation and Result

We have implemented the proposed interfacing techniques in an embedded system under control of the MCS-51 family of 8-bit microcontrollers. The system was designed to interface to the ST-350 radiation counter [5] in an experiment to study a decay time of radio active materials. Apart from being able to read data from the radiation counter, we added features to make the system possible to directly and automatically control the radiation counter; i.e. start/stop collecting data at a specified time and period, without any extra hardware. The system is used in conjunction with a moderate performance notebook. In the experiment, we programme the system to gather data once a day. As the system has its own memory,

data are, therefore, mostly stored in the memory and transferred to the computer only once a week. It is, however, definitely obvious that employing a microcontroller system and a computer and a few modifications of the instruments, only in the least complex unit, is viable and can ease experiment designs.

4. Conclusion and Future Work

We have presented two examples that exploit the benefits of microcontrollers to Physics experiments. The examples are: the automatic sampling system and the intelligent unit to interface between a computer and a numerically displayed instrument. Obviously, the application of microcontrollers to these experiments not only eases the design process of the

instrument and it also increases the functionality of the instrument. This comes from the fact that many complex parts of the instrument can mostly be substituted by software instead of hardware implementation which is a basic building block in nearly all traditional laboratory-designed instruments. It might be argued that the learning curve of modern electronic devices is very steep as most of them rely on using a computer program to design and simulate. This is in contrast to the traditional hardware-based approach to use only standard TTL/CMOS/OP-AMP devices. However, we have found that although learning to use the modern devices is fairly difficult at the first place, this difficulty will be paid off in a short term.

In order to further adopt modern electronic devices to Physics experiment, we have planned to employ CPLDs (Complex Programmable Logic Devices), FPGAs (Field Programmable Gate Arrays) and microcontroller with Ethernet connectivity in the new implementation of many laboratory-designed instruments. Currently, we have finished the CPLD implementation of an electronic accelerometer. In the implementation, a dozen of standard TTL devices is substituted by only a medium density CPLD with higher functionality/programmability than the former implementation. The result of this implementation will be presented elsewhere.

References

- [1] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principle and Practice*, 2nd Edition, Addison-Wesley, Reading, 1992.
- [2] P. Horowitz and W. Hill, *The Art of Electronics*, 2nd Edition, Cambridge University Press, 1989.
- [3] W. Kurdthongmee, *Experimental Study of an Electric Field Mapping by Use of a Computer and an Automatic Sampling System*, European Journal of Physics, September 2000, pp. 441-450.
- [4] W. Kurdthongmee, *Design and Implementation of an Automatic System to Construct Electric Field Maps by Use of a Microcomputer and Microcontroller*, Songklanakarin Journal of Science and Technology, Vol. 23, No. 1, January-March 2001.
- [5] J. B. Peatman, *Design with Microcontrollers*, McGraw-Hill, New York, 1988.
- [6] R. A. Plastock and G. Kalley, *Schaum's Outline Series: Theory and Problems of Computer Graphics*, McGraw-Hill, New York.
- [7] *ST-350 Radiation Counter Instruction Manual*, Spectrum Techniques, November 1994.
- [8] H. D. Young and R. A. Freedman, *University Physics*, 9th Edition, Addison-Wesley, Reading, 1996.



Wattanapong Kurdthongmee received his Ph.D. degree in Computer Science (Computer Graphics and Solid Modelling) from Brunel University, UK. in 1998. He received his bachelor degree and master degree both in Physics from Prince of Songkla University, Hatyai campus, Thailand in 1987 and 1994, respectively. Currently, Dr. Kurdthongmee is a faculty member in Division of Physics, Institute of Science, Walailak University, Thailand. His research interests include real-time embedded system, utilizations of electronic devices in physics experiment, and computer graphics.