Computer-Interfacing Development for Propeller-Anemometer

[*]Nor Hayati Saad, [**]Zuriati Janin, [**]Ruhaidawati Mohd Ali Piah

[*]Faculty of Mechanical Engineering, Universiti Teknologi MARA,
40450 Shah Alam, Selangor, Malaysia
(Tel: ++6-03-55435149; E-mail: norhayatisaad@salam.uitm.edu.my)
[**] Faculty of Electrical Engineering, Universiti Teknologi MARA,
40450 Shah Alam, Selangor, Malaysia
(Tel: ++6-03-55436070; E-mail: zurri_72@hotmail.com

Abstract: A Propeller-Anemometer is an instrument used specifically, to measure the wind speed. The accurate measurement of the wind speed is vitally important such required by any weather stations. In this research, the measurand of the instrumentation was the rotational speed of the propeller and the instrumentation result or output data was wind velocity. The speed measured was recorded digitally in the computer by using specific software. A specific sensor used to measure a variable by converting information of the variable (rotational speed of the propeller) into a dependent signal such as electrical signal in form of voltage. The development of Propeller-Anemometer involved few sets of instrumentation process and equipment. It included three major parts, mechanical, electronics and computer. The main instrumentation processes were physical and signal interfacing, signal conditioning, logic interfacing, data transmission to computer and processing the data. Generally, this paper presents the overall concept and design of Propeller-Anemometer Instrumentation. However, an emphasis was mainly in designing and building the interfacing system, hardware and software. Basically, for the first phase of the development, this project designed and built the RS232 terminal using Peripheral Interface Controller (PIC), PIC16F873. The hardware can be interfaced to computer or other compatible devices. This routine converted input voltage from the circuit to speed (velocity) and transmitted them afterwards to the target device by using the RS232 transmission protocol. This implementation implied a computer display as visual interface. For the purpose of this paper, RS232 data transmission was carried out using a Microsoft Visual Basic software routine.

Keywords: Interfacing, anemometer, velocity, Peripheral Interface Controller, Visual Basic,

1. INTRODUCTION

The development of Propeller Anemometer (PA) involved with few functional elements. Refer to Fig. 1 and 2. In this instrumentation, the physical parameter being measured was the rotational speed of the propeller shaft. The first phase of computer-interfacing development for PA was to design and build the RS232 terminal using Peripheral Interface Controller (PIC), PIC16F873. The appropriate software and hardware was developed in order to display the speed parameter on the personal computer. PIC16F873 microcontroller was used as the control device for the whole system. The simple programming using Visual Basic was developed in order to display and analyze the output in the computer.

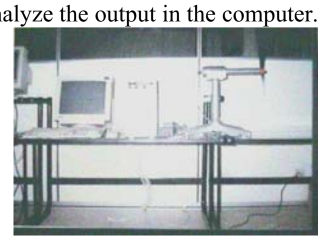


Fig. 1 The First Phase of the PA Development

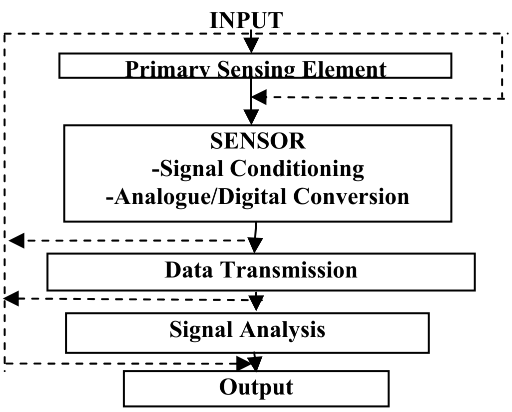


Fig. 2 Building Blocks of PA Instrumentation

2. HARDWARE DEVELOPMENT

2.1 Interfacing Hardware

There are many types of Peripheral Interface Controller (PIC) in the market. Each PIC has a specific number of inputs and outputs. PIC has all the CPU (Central Processing Unit), memory oscillator, watchdog and I/O incorporated within the same chip. The PIC controller saves space, the circuit can be compactly made, easy instruction set, protection bit for security, and so on. The PIC16F873 was chose because it has a special function called USART (Addressable Universal Synchronous Asynchronous Receiver Transmitter), which is one of the two communication ports equipped with PIC16F873. USART is also known as a Serial Communication Interface or SCI. The USART can be configured as a full duplex asynchronous system that can communicate with peripheral device such as personal computers. In the asynchronous mode communication of USART, the RX port was used for receiving and the TX port used for transmitting; so, it is possible to send and receive at the same time (Full duplex). In the asynchronous mode, it used RC6/TX/CK (17 pins) for the data transmission and RC7/RX/DT (18 pins) was used for the data receiving (Refer to Fig. 3). In order to use the USART the mode of the input/output of the TX port, the RX port must be set to the TRISC register. Bit 6 of C port was set to output mode (0) and bit 7 was set to input mode (1). To use RC6 and RC7 as the USART port, the SPEN bit of the RSCTA register must be made “1”. The signalling speed of the USART was controlled by BRG (Baud Rate Generator). The clock frequency of the PIC and signalling speed can be calculated by the following formula in Table 1.

Table 1 Calculation of PIC Clock Frequency

Mode	Low-speed (BRGH=0)	High-speed (BRGH=1)
Asynchronous (SYNC=0)	**Fosc/(64(X + 1))**	**Fosc/(16(X + 1))**
Synchronous (SYNC=1)	**Fosc/(4(X + 1))**	

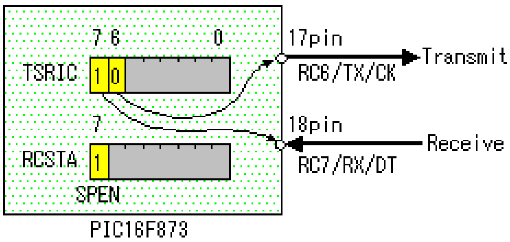


Fig. 3 Transmitting and Receiving Pin of PIC16F873

2.2 System Hardware

The hardware device was used to control the whole operation**.** The hardware consisted of several components, which were: DC power supply, Serial Port, RS232 Interface IC, PIC16F873. Fig. 4 shows the schematic diagram of the hardware interface device. Basically, a range of 4.5V to 5.0V DC was fed to the hardware Interface device. This voltage was used as the power supply for the RS232 interface IC and PIC16F873. The serial port used in the hardware interface device consists of 9 pins. It referred to the port that has been connected to the COM port of the personal computer for the data transmission. The data then was transmitted to the personal computer.

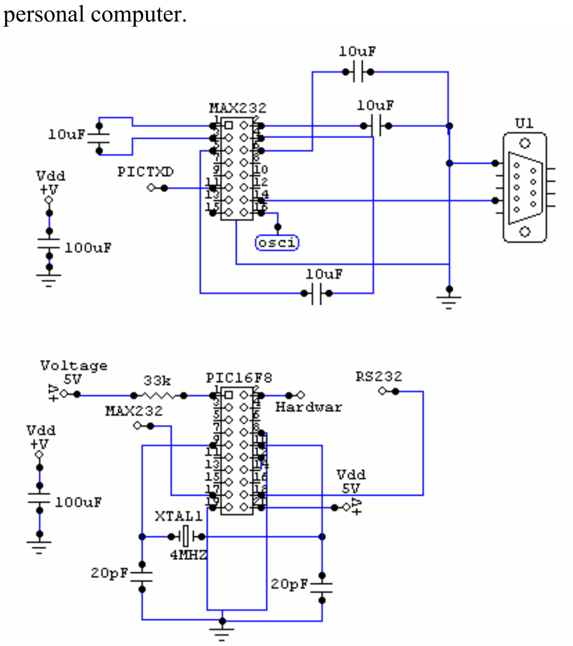


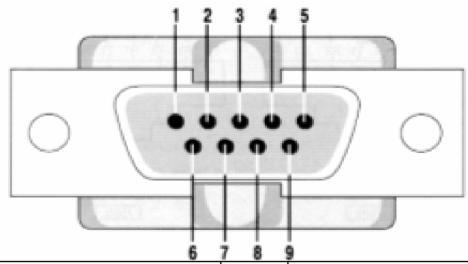
Fig. 4 Schematic Diagram of the Hardware Interface Device

2.3 MPLAB IDE

The PIC assembly language programming was developed using MPLAB IDE software. MPLAB IDE uses Integrated Development Environment IDE. The IDE provides the flexibility to develop and debug Firmware for PIC microcontroller families. MPLAB IDE is a Windows based development platform for the Microchip Technology PIC microcontroller families and runs under Microsoft Windows operating system. MPLAB IDE offers a project manager, program text editor, a user configurable toolbar containing four predefined sets and a status bar, which communicates editing and debugging information [1].

2.4 RS232

Serial communication links utilize a single data line to transmit information. The standardization of the RS232 serial port as part of a computer system and it compatibility has led to this communication interface being used in the project to be interfaced to the computer. The Electronic Industries Association RS232 standard was developed for serial data interface. It defines the interface between the Data Terminal Equipment (DTE) and Data Communication Equipment (DCE) employing serial binary data interchange [2]. Fig. 5 shows the serial 9 pins (PC) connector.



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

Fig. 5 The serial 9 pins (PC) connector

3. SOFTWARE DEVELOPMENT

3.1 Visual Basic 6.0

It was used for the display of the RS232 Debug Interface. Microsoft Visual Basic is good software for developing window interface; it involves fragments of Basic Code when the user performs certain operations on graphical objects. There are four types of Visual Basic code; sub procedure, function procedure, variables and constant. Their main functions are to execute in responding to an event, write the arguments, store the values and declare the same value of variables to constant, respectively. Refer to Fig. 6 for the process flow of the display development.

3.2 Execution Flow of the Display software

Software development systems allow programs to be developed with assistant of special software and peripherals and later to be transferred into the application programming interface on which they will run. The developed program was running in real-time. Refer to Fig. 7 for the developed display software. The program will first makes a declaration of the variables. The program will set the timeout for the data to appear. Then it will detect the serial port for the transmitted signal. The programs then will check if the data has arrived or not. If the data is arrived it will convert PWM input to speed-reading in meter per second or otherwise the program will end. The Visual Basic program will wait for the data transmissions from the interface device of the hardware. Then when the data is received it will triggered the program to operate. The final result of the display for the anemometer is shown in the Fig. 8.

4.0 CONSTRUCTION OF THE PIC ASSEMBLY PROGRAM

4.1 PIC Program

The PIC assembly language programming was developed using MPLAB IDE software. The IDE provides the flexibility to develop and debug Firmware for PIC microcontroller families. Fig. 9 shows the MPLAB IDE window. The steps in the construction of the PIC assembly program were:

a) The definition part of the hardware.
b) The definition part of the register files.
c) The initialization of processing part.
d) The main processing part.
e) The source ending.

4.2 Execution Flow of the PIC Software

The main hardware interface PIC Software flowchart was shown in Fig. 10 below. The program will start with the initialization of the hardware and RS232 port. Then the program will check the scan pattern reception flag. If ‘No’ or Clear means scan pattern reception flag received ‘Low signal’ it will loop. If ‘Yes’ or Set means scan pattern reception flag received ‘High signal’ it will go to hardware scan pattern decoding subroutine activities. At this stage, the program will detect any input on the hardware. Then the program will wait for any changing of the input at the hardware. Then the hardware will send the input and the program will read the input. Before ended, the program will transmit and display the input to the personal computer across RS232 serial link.

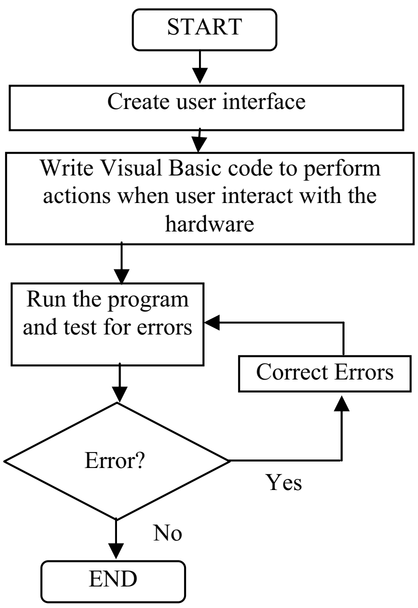


Fig. 6 Process Flow of the Display Software Development

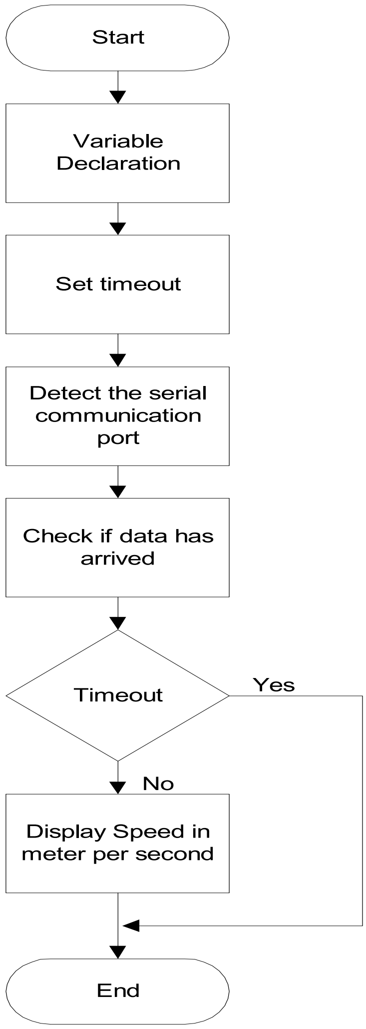


Fig. 7 Execution Flow of the Visual Basic Display Software

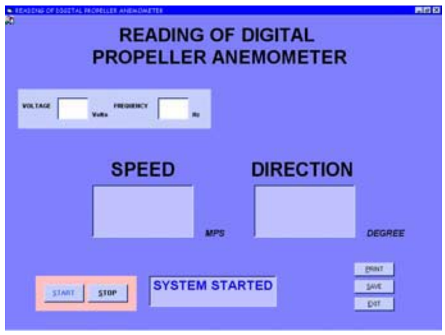


Fig. 8 The result of the Visual Basic programming

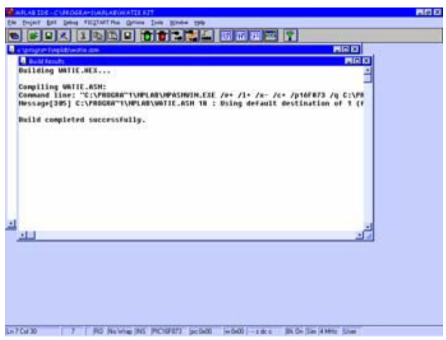


Fig. 9 The MPLAB IDE window

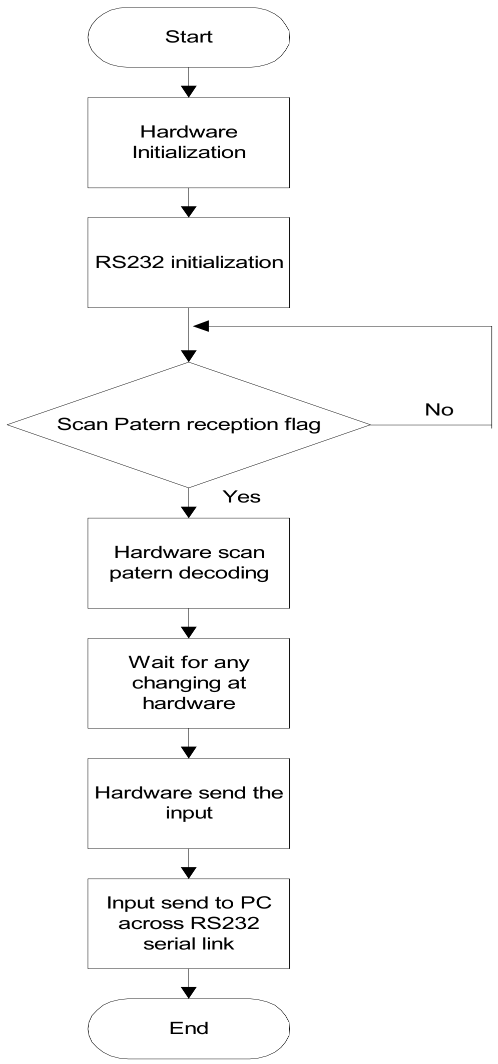


Fig. 10 The main Hardware Interface PIC Software flowchart.

5. PIC CALIBRATION AND RESULT

5.1 PIC 16F873 Calibration

In the experiment, the range of input frequency from the frequency generator was fed into the PIC16F873. The PWM (Pulse Width Modulation) and output voltage was observed. Fig. 11 shows the relationship between input frequency and output voltage and Fig.11 shows the relationship between input frequency and the PWM output. From Fig. 12 output voltage of 5 Volt was measured at the maximum frequency of 255 Hz. It can be observed that a PWM output was same as the input frequency value while the output voltage was 4.61 volt when the input frequency reaches the maximum (255Hz).

5.2 Result

As the speed of the propeller increased, the width of the output PWM also increased, as well as the output voltage. The speed of the propeller is equal to the output PWM. The final relationship was shown in Fig. 13.

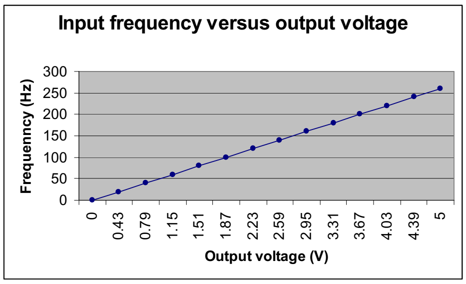


Fig. 11 Input Frequencies versus Output Voltage

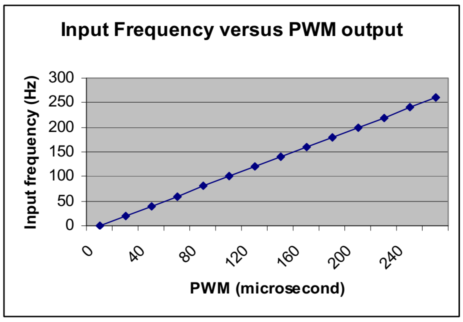


Fig. 12 Input Frequencies versus PWM (Pulse Width Modulation) Output

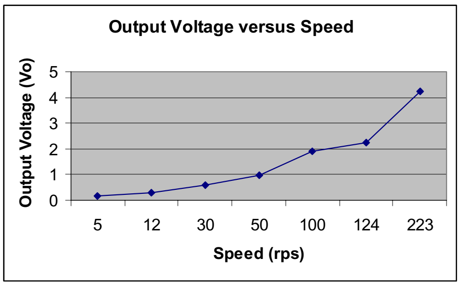


Fig. 13 Relationship between output voltage and speed

6. CONCLUSION

The Propeller Anemometer is an instrument used to measure the wind velocity. For the first phase of development, the simple interfacing software and hardware was developed on how the physical signal can be interfaced with the computer. Visual Basic was used as a language. The project was successful. The result of the speed was visualized on the screen of computer. The advantages of using PIC microcontroller are multipurpose, multifunctional and can be reprogrammable. PIC is convenient because its calculation part, memory and I/O are incorporated in one IC. The instrumentation of PA can be considered as digital instrumentation. Few advantages of digital instrumentation:

a) The interpolations are not required, thereby minimising errors caused by misinterpretation.
b) Digital signals are inherently noise-resistant.
c) The nature of digital signals and circuitry permits the signals to be regenerated and reconstituted from point to point throughout the processing chain.
d) Digital circuits imply relatively low operational voltages; 5 to 12 V.

ACKNOWLEDGEMENTS

The authors would like to thank to IRDC, UiTM for sponsoring the project and to faculty of Electrical Engineering for the use of the laboratory and equipments.

REFERENCES

[1] Gadner, Nigel, "A Beginner Guide To The Microchip PIC", Revision 2.0. 2nd Edition, Microchip Technology Inc, 1995

[2] Brey, "Microprocessors and peripherals; Hardware, Software. Interfacing and Applications", Merrill Publishing Company, Colombus, 1990.

[3] Ekedahl, Micheal and Newman, William, "Programming with Microsoft Visual Basic 6.0: An Object Oriented Approach, Course Technology", International Thomson Publishing company, Cambridge, 1999.

[4] Ronald L.Krutz, "Interfacing Techniques In Digital Design with Emphasis on Microprocessor", John Wiley & Sons, Inc, Canada 1988.