A LOW COST REMOTE DATA ACQUISITION SYSTEM BASED ON INTERNET AND MICROCONTROLLER FOR WIND MEASUREMENT

Manish Kumar Das* **and Thananchai Leephakpreeda**

Received: July 09, 2011; Revised: July 24, 2012; Accepted: July 27, 2012

Abstract

There are more and more demands for measuring variables in various engineering activities and collecting the measurement data from distant observation sites such as wind measurements from different far-away masts. In turn, a remote data acquisition system is required for such various customized tasks. It is a fact that microcontrollers featuring high-performance I/O hardware and software are quite flexibly developed in product markets while Internet technology is now available and easily accessible for a world-wide network. For a technical contribution, this paper presents the complete development of a low cost, easy to implement,remote data acquisition system embedded on a microcontroller so as to gain measurement data from remote measurement sitesthrough the Internet network with the techniques on the Transmission Communication Protocol/Internet Protocol (TCP/IP) and File Transfer Protocol (FTP). Additionally, the experimental results of real-timeimplementation show that theproposed techniqueshave not only been flexiblydeveloped by usingan available low-cost microcontroller but alsoare feasible with current Internet traffic for acquiring any remote measurement data where an Internet connection is available.

Keywords: Wind measurement, microcontroller, Internet, remote data acquisition system

Introduction

A data acquisition system is widely used in the fields of measurement and control for scientific research and industrialproduction. Briefly, the data acquisition system measures some physical variables of the processes in the real world and transfers those quantities in the form of digital/analog signals to analyzing/storing devices like computers for any technical user's purposes. Physical properties such as wind speed, wind direction, temperature, etc. at different attitudes are a few examples of analog signals in a data acquisition system of wind measurement (Bishop, 2008). In thetraditional method, the measurement data

School of Manufacturing Systems and Mechanical Engineering, Sirindhorn International Institute of Technology, Thammasat University P.O. Box 22, Thammasat Rangist Post Office, PathumThani, Thailand, 12121. Tel.: 0-2986-9009 Ext.: 2204, Fax.: 0-2986-9009 Ext.: 2201; E-mail: thanan@ siit.tu.ac.th.; ermkdas@yahoo.com

* *Corresponding author*

of wind arelogged in standalone equipment and collected periodically from the distant sites. Remote data acquisition, based on a microcontroller, ispresently interesting to product developers and researchersas a more convenient, less costly and more flexible method of measurements compared withthe traditional method.It can be seen that demands for a remote data acquisition system are increasing nowadays in many scientific and engineering activities. In the case of wind energy study, researchers need time-series records of wind speed, wind direction, and temperature of wind in a sampling interval of preferably 10 minutes from different remote sites in studies (Risø National Laboratory, 2009). A remote data acquisition system is important for the Inland Revenue Department (IRD) of the Nepal Government to acquire instant production data from beer manufacturersto monitor for taxation purposes. Some studies in the field of measurement have considered a microcontroller in a data logging system due to its easy availability, cheapness and its flexibility in functions & features and it is used along with the Global System for Mobile Communications (GSS)/General Packet Radio Service (GPRS) and radio frequencies (RF) for the transmission of data (Akkaya and Kulaksiz, 2004; Belmili *et al.*, 2010; Benghanem, 2009, 2010). This paper describes the apparent development of a remote data acquisition system by using a microcontroller and techniques on the TCP/IP and FTP for transmitting remote data through the Internet network in a practical scenario of wind measurement.

Methodology

The complexity of a remote data acquisition system increases withthe number of remote stations and the number of sensors in each station. Basically, the selection of a data acquisition system for a particular site is first based on the numbers and types of sensors, the accuracy of measurement, and the sampling rate of data acquisition. For instance, the number of input channels is to be greater than the number of input signals and the types of the input channels are to correspond to the types of input signals such as analog input or digital input. In the case of wind measurement, there are 3 analog quantities, namely, wind speed, wind direction, and temperature of the air at a given altitude which are required for measurement. At present, some microcontrollers available on themarket have up to 16 input channels for an Analog to Digital (A/D) converter with a resolution of 10 to 12 bits (Microchip Technology Inc., 2011). Additionally, there are advanced auxiliary components available now to strengthen the applicability ofthe remote data acquisition system such as a large storage capacity for the measurement of data and terminals required for the communication system, which allow the data acquisition system to havean in-built capability forsending data to a network in order to minimize the total cost of the system itself. However, data storage facilities are to be provided at each remote site for data safety in the case of failure of the network. In fact, measurement data can be transferred from remote sites to a server computer at a monitoring site via the Internet network. Currently, an Internet modem is also connected with the terminals of the data acquisition hardware via wireless communication. Finally, the server computer is to be equipped with-appropriate software to acquire measurement data from all the remote sites and store thedata as well as display analytical results graphically.

It should be mentioned for alternative extended works as per conveniencethat there are different ways for data communication between the remote sites and the monitoring site. The data communication can be chosen according to the available facilitiesat the remote sites. If a mobile network or telephone service is available at the remote sites,then GSM/GPRS modems or simple telephone modems can be used instead. In the worst case, satellite communication is needed in the absence of the mentioned options. Figure 1 illustrates some options, namely a wireless Internet network or wired network GSM/

GPRS modem network for data communication.

Remote Data Acquisition System

Hardware Assembly

In this study, a microcontroller, 3 sensors (wind speed, wind direction, and temperature) and an Internet network (TCP/IP and FTP protocol for a data communication system) are integrated to functionas a remote data acquisition system of wind measurement for real-time implementation at the sites.

To grasp how to practically implement the remote data acquisition system of wind measurement, a typical PIC16F877A micro-controller is used for a concrete example of the development of this work. It has an in-built 14 channel ADC, 256 bytes EEPROM Memory, 3 independent timers/counters, 35 input/output pins and communication supports for RS-485, RS-232, and synchronous serial port support. It runs witha speed of 0-20MHz and requires 2.0-5.5 V power supply (Microchip Technology Inc., 2010).

Figure 2 shows a block diagram of a PC based data acquisition system where the PIC16F877A microcontroller is connectedwith the sensors, an LCD (Liquid Crystal Display) and computers via an RS232 to COM port. In the absence of a COM port, the microcontroller can be also connectedthrough a computer with an RS232 to USB converter device.

One temperature sensor (LM 35 DZ), wind speed sensor, and wind direction sensor are connected to the analog input port A of the microcontroller as illustrated in Figure 3. These sensors' inputs are set to provide analog input between 0-5 V. The LM35 DZ is a three-pin analog sensor that can measure temperature to within 1°C accuracy in the temperature range between 0°C and 100°C (Ibrahim, 2008). Three pins are connected to the 5V DC source, the ground, and the analog input of the microcontroller, respectively. In this study, the microcontroller is operated by an 8 MHz external oscillator connected to pins 13 and 14. The RS232 serial communication is a form of asynchronous data transmission where data aresent character by character. Each character

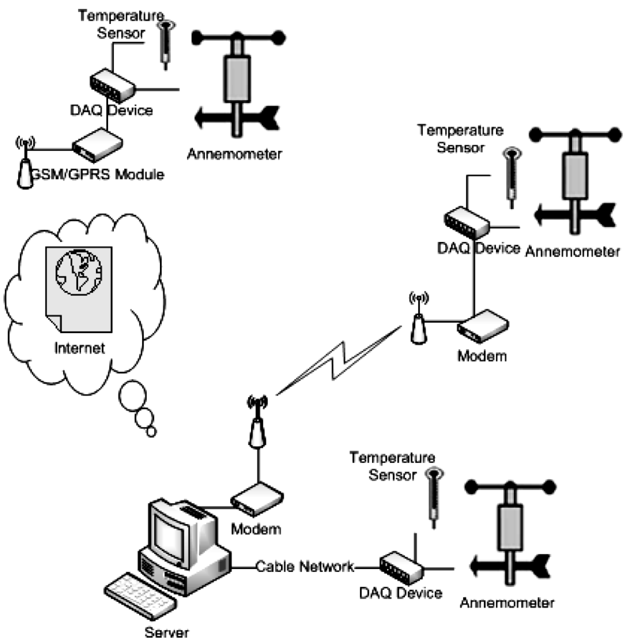

Figure 1. Remote data acquisition system for wind measurement

is preceded with a start bit, 7 or 8 data bits, an optional parity bit, and 1or more stop bits. The most common format is 8 data bits, no parity bit, and 1 stop bit (Axelson, 2007; Ibrahim, 2008). The MAX 232 chip is connected to port C of the microcontroller so as to provide the RS232 terminals with-connectivity to other devices likea GSM/GPRS modem and computer. The LCD is connected to port B of the microcontroller to display measurement data at instantaneous time. The LCD is also connected to a separate potentiometer to adjust its display contrast.

Firmware Development

In order to operate the microcontroller, a firmware is developed which mainly consists of coding for the timer, ADC (Analog to Digital Conversion), and UART (Universal Asynchronous Receiver Transmitter). The

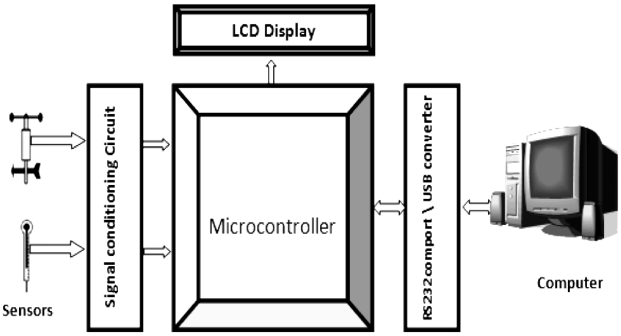

Figure 2. Structural architecture of microcontroller based data acquisition system

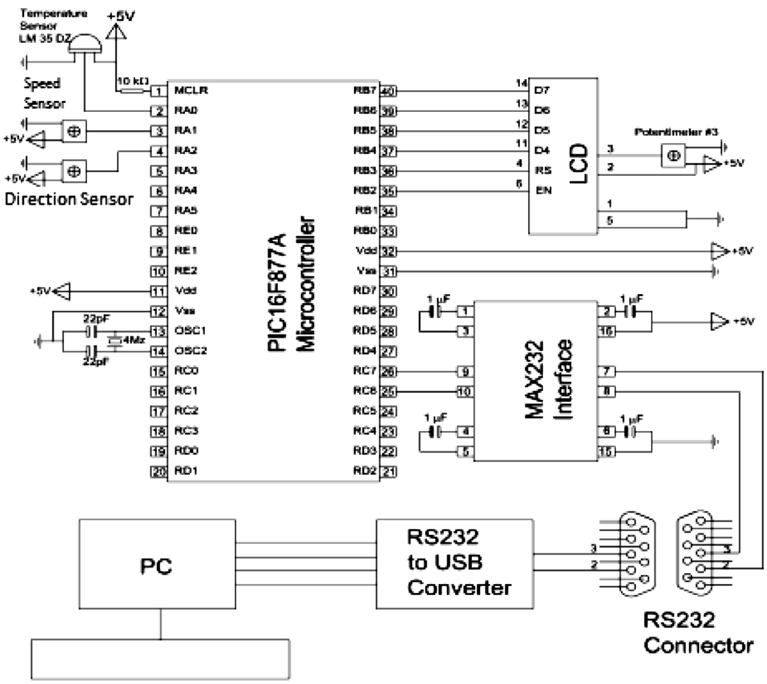

Figure 3. Circuit diagram of PIC16F877A microcontroller for data acquisition system

development of the firmware starts with writing the application program. The program for the microcontroller is written in C programming language, which is translated into executable task by Mikro C software from Mikro Elektronica (Verle, 2009). The compiler converts the program file to the hex file. Finally, the firmware is loaded into the Read Only Memory (ROM) of the microcontroller chip by a Microchip In Circuit Debugger (a special hardware tool to copy coded instruction into the memory of a microcontroller). A computer is connected to the Microchip In Circuit Debugger through a USB port.

Flow Chart of Developed Firmware

Figure 4 summarizes the operating of the microcontroller for the data acquisition system. A real time clock is important for the microcontroller to execute instructions in the predefined time interval. The microcontroller initiates the timer to implement the real time clock for the data acquisition system. The ADC is initiated for analog to digital conversion. The UART module is initiated to implement the RS232 COM port communication. The time interval for data acquisition is set in the next step. The microcontroller runs in an infinite loop after defining the time interval for measurements. When the predefined time has elapsed then the ADC converter capturesthe analog signal and converts it into corresponding digital values for each input. If time has not passed then the microcontroller does not do anything and sits in the idle position.Each digital value is then scaled toan appropriate value as per the range of measurements specified in sensors' manual. Finally, all measurement values along with a time stamp are sent to the RS-232 port in the form of an array of characters. The microcontroller returns back to the start of the cycle and waits for time to pass for the next measurement. The value sent to the RS-232 port is read by the connected device to send to the remote server.A GSM/ GPRS modem, infrared communication port, and radio frequency device may be used as the capturing device. In this study the computer is connected to the COM port to read measurement values from the microcontroller via the client application software.

Software Development

Two software applications are developed as follows by using the Visual Basic 2008 to test the performance of the remote data acquisition system as mentioned in the

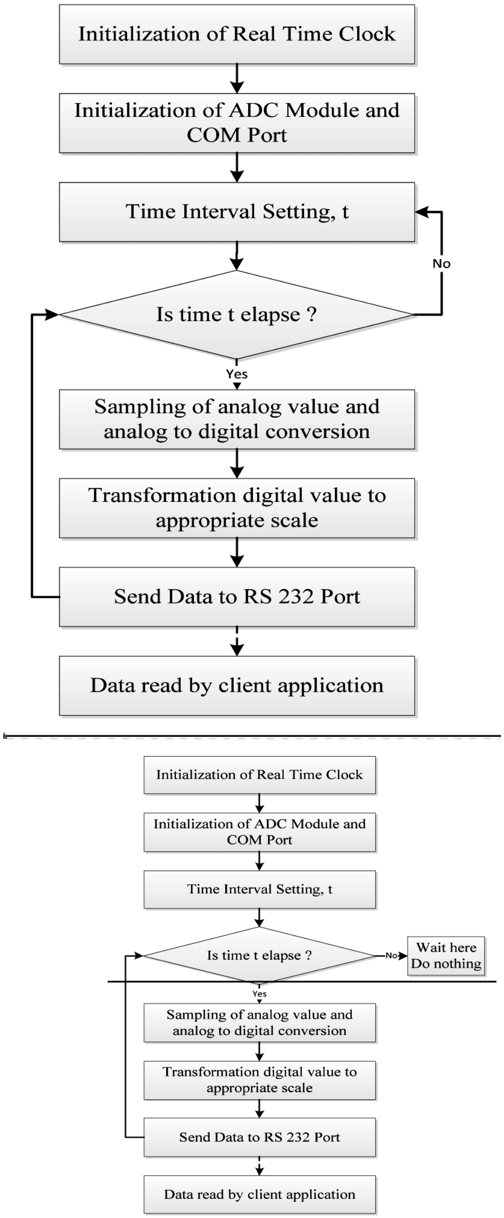

Figure 4. Flow chart of operation of microcontroller

proposed technique. These applications are based on Server/Client architecture which is based on TCP/IP protocols. The first one is Let us considerthe client side application. Firstly, it opensthe communication (COM) port to communicate with the microcontroller. The functions for opening the communication port and reading the values from the microcontroller are given in Table 1 and Table 2, respectively (Axelson, 2007). The client application captures data from the microcontroller, saves data to a hard disk, and displays data in a monitor as well as sending data to remote servers. It usesthe TCP/IP protocol for communication with the server.In addition, the client software also uploads the saved XML file to the web server in the predefined interval using File Transfer Protocol. The XML file is easily integrated into the web page to display data in tabular format. So, the measurement data can be accessed from anywhere in the world using Internet browsing. The codesfor uploading the data to the remote server, saving data tothe local hard disk, and uploading data to webpages are given in Table 3, Table 4 and Table 5, respectively (Reid, 2004; Evjen *et al*., 2008; Herman *et al*., 2008; Stephens, 2008). Figure 5 and Figure 6 show the snapshots of the remote site application and snapshots of the displayed data in the web site, respectively.

Forthe latter, at the monitoring site, ‘Remote DAQ: Server’ is the server application as shown in Figure 7. The server needs a unique IP address and separate port to listen to client applications. Its main function is to receive data via TCP/IP protocols, display the data that is sentfrom the remote data acquisition system, and record data trends in a hard disk of the server. This software should be run on a

Table 1. Visual Basic 2008 code for opening Client COM port communication

```
‘Opening DAQ serial port for communication
Private Function openSerialPort(ByValcomPort As
_ SerialPort, ByValportName As String, ByVal_
BaudRate As Integer) As Boolean
        Dim index As Integer = -1
        Dim nameArray() As String
nameArray = SerialPort.GetPortNames
        Do
index += 1
        Loop Until ((nameArray(index) = portName) or _
(index = nameArray.GetUpperBound(0)))
        If (index = nameArray.GetUpperBound(0)) Then
        If (nameArray(index) = portName) Then
        Else
MsgBox(“Port not found”)
openSerialPort = False
                    Exit Function
               End If
        End If
comPort.PortName = portName ‘ BoudRate, _
Parity.None, 8, StopBits.One)
comPort.BaudRate = BaudRate
comPort.Parity = Parity.None
comPort.DataBits = 8
comPort.StopBits = StopBits.One
comPort.Open()
‘MsgBox(comPort.IsOpen)
openSerialPort = True
        End Function
```

machine having a unique public IP address in order to be recognized by all the remote stations' computers.The server application uses the multithreading approach to listen to all clients in parallel. When a client is connected with the server then the server creates a separate thread to handle each client.Data from each site are started with their identity name (Station code) and followed by date stamps and measurement values, respectively. Therefore, the server recognizes each client easily. The Visual Basic code for starting the server to listen to a client and receive data from a client are given in Table 6 and Table 7, respectively (Evjen *et al*., 2008; Herman *et al*., 2008; Stephens, 2008).

Table 2. Visual Basic 2008 for retrieve data DAQ system hardware to computer

```
Private Sub tmrGetData_Tick(ByVal sender As System.Object, _
ByVal e As System.EventArgs) Handles tmrGetData.Tick

        On Error Resume Next
        data = readSerialPort(cp)
        Dim flag As Boolean
        If data = “” Then
              flag = False
        Else

              If data.Contains(“T”) And data.Contains(“E”) Then
                   flag = True
updateSpeed(flag, data)     ‘ routine for update display monitor
              Else
uData = uData + data
              If uData.Contains(“T”) And uData.Contains(“E”) Then
                   data = uData
uData = “”
                   flag = True
updateSpeed(flag, data) ‘ routine for update display monitor

                         End If
                   End If
              End If
        End Sub
```

Table 3. Visual Basic 2008 code for sending data to remote server

```
‘Routine for sending data to server
Private Sub sendData(strData as String)
        On Error Resume Next
        Using cl As New TcpClientcl. Connect (IPAddress. Parse_
        (txtIPaddress._Text),8000)
        Dim bw As NetworkStream = _
        cl.GetStream()
        Using w As New BinaryWriter(bw)_
        w.Write(strData)
        End Using
        cl.Close()
        End Using
 End Sub
```

Table 4. Visual Basic 2008 code for saving data to hard disk

```
Private Sub updateXML(ByVal d As String, ByVal t As String,_
ByVal s As String, ByValdr As String, ByVal tm As String)
       On Error GoTo MM
       Dim Wr As XElement = XElement.Load("D:\wind.xml")
       Dim newWr = <WD>
<Date><%= d %></Date>
<TIME><%= t %></TIME>
<Speed><%= s %></Speed>
<Dir><%= dr %></Dir>
<Temp><%= tm %></Temp>
<Remarks>ok</Remarks>
</WD>
Wr.AddFirst(newWr)
Wr.Save("D:\wind.xml")
Debug.WriteLine("File (D) saved on " &TimeOfDay)
File.Copy("D:\wind.xml", "D:\wind1.xml", 1)

File.Copy("D:\wind.xml", "e:\wind.xml", 1)
Debug.WriteLine("File (D)  saved on " &TimeOfDay)
GoTo NN
MM:
Debug.WriteLine("File not saved on " &TimeOfDay)
NN:
             'Debug.WriteLine(Wr.ToString)
       End Sub
```

Table 5. Visual Basic 2008 code for uploading data to web server

```
Private Sub uploadSite()
       Dim thisFile As FileInfo = New _
       FileInfo("D:\wind.xml")
       Dim ITC As Type
       Dim parameter() As Object = New _
       Object(1){}
       Dim ITCObjectAs Object
       On Error Resume Next
       ITC = Type.GetTypeFromProgID _
       ("InetCtls.Inet")
       ITCObject = Activator.CreateInstance(ITC)
       parameter(0) = CType _
       ("ftp://remotedaq.orgfree.com:_
       password@orgfree.com", String)
       parameter(1) = CType("PUT " + _
       thisFile.FullName +  " /" + _
       thisFile.Name, String)
       ITC.InvokeMember("execute", _
       BindingFlags.InvokeMethod, Nothing, _
        ITCObject, parameter)
 End Sub
```

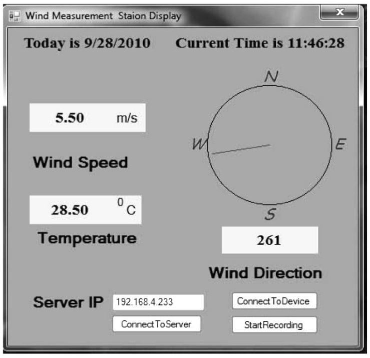

Figure 5. Client application software

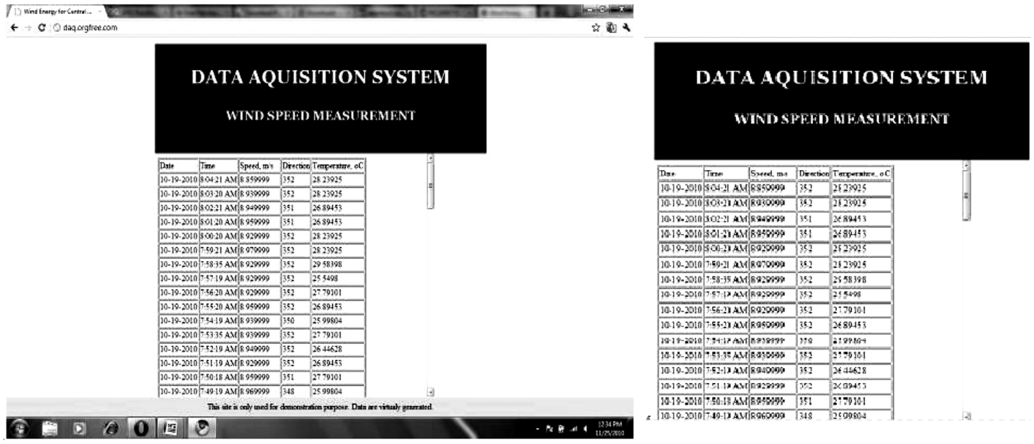

Figure 6. Website showing measured data

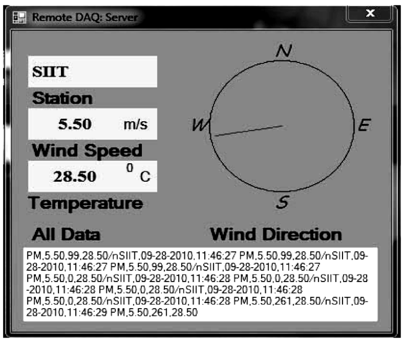

Figure 7. Server application software

Table 6. Visual Basic 2008 code for starting remote server

```
//Routine for starting server
Private Sub listenServer()
        Dim tcpListner As New TcpListener(8000)
        Dim handlerSocket As Socket
        Dim thdstHandler As ThreadStart
        Dim thdHandler As Thread
tcpListner.Start()
        Do
handlerSocket = tcpListner.AcceptSocket()
            If handlerSocket.Connected Then
SyncLock (Me)
allStaions.Add(handlerSocket)
              End SyncLock
              thdstHandler = New ThreadStart _
              (AddressOfhandlerThread)
              thdHandler = New Thread(thdstHandler)_
thdHandler.Start()
              End If
        Loop
handlerSocket = Nothing
        End Sub
```

Table 7. Visual Basic 2008 code to receive client data

```
//Routine for handling data from clie
Public Sub dataHandlerThread()
Dim handlerSocket As Socket
handlerSocket = allStaions(allStaions.Count - 1)
Dim networkStream As NetworkStream =_      New
NetworkStream(handlerSocket)
Dim bSize As Int16 = 1024
Dim thisRead As Int16
Dim dByte(bSize) As Byte
SyncLock Me
Using w As New BinaryReader(networkStream)
a = w.ReadString
parts = a.Split(delimiters)
a1 = parts(0) & "," & parts(1) & "," &_ parts(2)_
& "," & parts(3) & "," & parts(4) & "," & parts(5)
Using fs As New FileStream(filename, _
FileMode.Append)
Using wfs As New StreamWriter(fs,  _
Encoding.UTF8)
wfs.WriteLine(a1)
End Using
End Using
blnUpdate = True
End Using
End SyncLock
handlerSocket = Nothing
End Sub
```

Results and Discussion

As illustrated in Figure 8, the remote data acquisition system over the wireless Internet network is tested by different methods for different test parameters. A 3-cup anemometer and a wind vane are used to measure the wind speed and the wind direction, respectively. They are manufactured and calibrated by Theodor Friedrichs & Co, Germany. The measurement ranges are 0-60 m/s and 0-360^o for the wind speed and the wind direction, respectively. The server is assigned with a unique IP address and it receives the data from the client's applications that areinstalled with the remote data acquisition system. These server/client applications run without any interruption. The remote data acquisition system is tested to upload data to website every minute. In a 24 h testing period, it is found that there is an average of 1% data loss in the website since uploading time may exceed 1 minute. The data loss is completely rectified by setting a time interval to 5 min. Actually, a 10-min interval is recommended for wind measurement. Figure 9 shows examples of measurement data such asthe wind speed and the wind direction, which are recorded on August 10, 2011 at a laboratory building. Wind speed is found to vary between 0–8 m/s between 9:30 am to 3:30 pm on that particular day while wind is flowing in the same direction.Additionally, Figure 10 depicts the corresponding ambient temperature during the day of experiment. The proposed technique can be used for the remote data

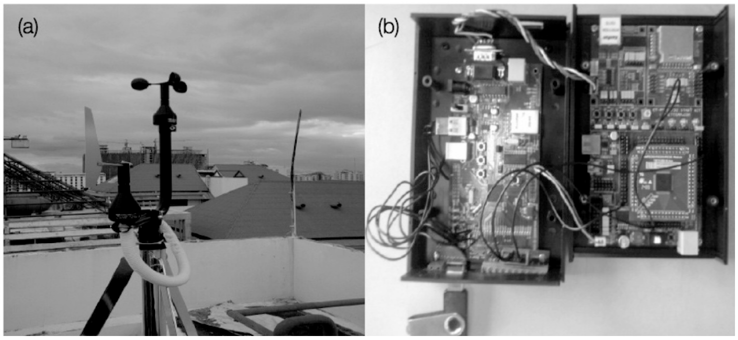

Figure 8. Installation of developed system: (a) wind measurement device and (b) remote DAQ system

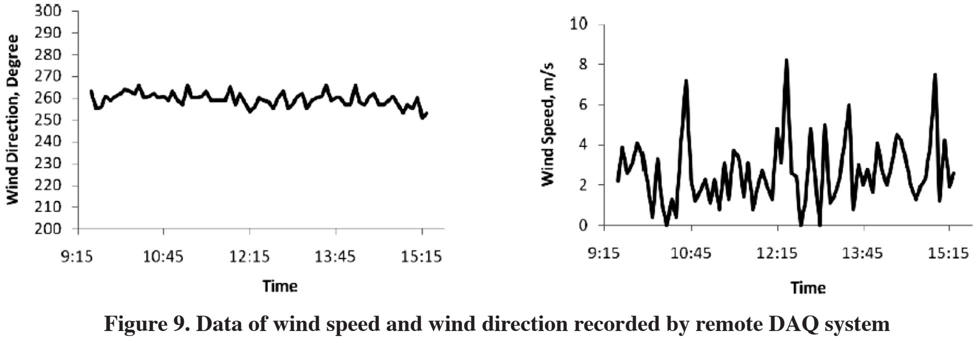

Figure 9. Data of wind speed and wind direction recorded by remote DAQ system

acquisition system for wind measurement. However, the system also runs in a slow Internet connection with exceptional data loss. The lost data can be compensated by the measurement data at the remote site later. Data acquisition system accuracy is found tobe within ± 0.1 m/s, ± 10, ± 10C for wind speed wind direction, and temperature, respectively. A variation (fluctuation) test is performed by assigning a constant known analog value to all 3 analog channels. The variation is observed against the time period and varying ambient temperature. It is found that variation is within the achieved accuracy limit. A repeatability test also reflects errors within the accuracy limit. This system is recommended for calibration every year. Measurement values should be adjusted in the case of an error larger than the accuracy limit by applying correction factors. The main advantage of this system is that it is very cheap and easy to implement. The DAQ system's components costapproximately 1000 Baht. An Ethernet controller can be used to send data from a remote client site tothe sever instead of a computer and the client application as a further development of the system which requires a high speed microcontroller. The remote data acquisition system is considered a complete package for remote wind measurement. The proposed

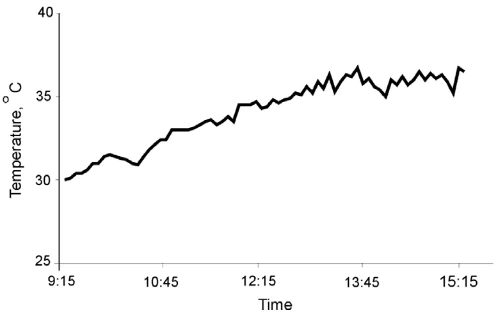

Figure 10. Data of ambient temperature

Table 8. Comparison of price and capability of RDAQ system with commercial data loggers in market

Parameters	RDAQ system	Commercial Remote Data Loggers	Commercial Standalone Data Loggers
Number of Analog Channels	9	4-8	8-12
ADC resolution	10	12-16	12-16
Maximum Sampling rate	1 sample per second	1-2 sample per second	1-10 sample per second
Anolog Inputs	0-3.3 VDC	4-20 mA, 0-5 VDC	0-20 mA, 4-20 mA, 0-30 VDC
Power Supply	9 VDC	220 V AC, 5-25 VDC	220 V AC, 5-25 VDC
Maximum Power Consumption	10 watt	10 watt	2.5 to 5 watt
Local Storage Media	USB flash drive Supported up to 16 GB	Not Available	USB flash drive Multimedia memory card
Remote Data Transmission	Ethernet (TCP/IP)	GSM/GPRS (TCP/IP and SMS)	Not Available
Flexibility	Number of channels, Higher resolution ADC, other control function can be added by slight modification	Different type and range of analog inputs can be measured	Different type and range of analog inputs can be measured
Price	**15000 Baht**	**15000-25000 Baht**	**5000-22000 Baht**

remote data acquisition system in this study is not suitable for high accuracy measurement. However, it can be used for high accuracy requirement by use of additional external high bits analog to digital converter. It is needed in slight modification in design of hardware as well as firmware of system. Table 8 summarizes the main differences between the developed system and commercial similar products.The remote data acquisition system in scenario of wind measurement has prevailing capability to capture wind data in remote locations by leaving standalone equipment and collecting and sending data via Internet network periodically. Besides wind measurement, the application of the proposed methodology can be generalized to other types of sensors according to those specific demands for remote measurement.

Conclusions

The remote data acquisition using a microcontroller via the Internet network is covenient to be implemented just like plug and play,not only because of the wide availability of Internet service nowadays but also because onlya little bit of coding effortis required for microcontrollers as developed in this work. The proposed methodology allows us to capture data in remote sites such as measuring the wind data from the masts at distant sites.

Acknowledgment

The authors sincerely thank the National Research Council of Thailand (NRCT) and the Thailand International Development Cooperation Agency (TICA) for financial support in this study.

References

Akkaya, R. and Kulaksiz, A.A. (2004). A microcontroller based standalone photovoltaic panel system for residential application. Appl. Energ., 78:419-431.

Axelson, J. (2007). Serial Port Complete: COM Ports, USB Virtual COM Ports and Ports for Embedded Systems. 2nd ed. Lakeview Research LLC, Madison, WI, USA, 379p.

Belmili, H., Cheikh, S.M.A., Haddadi, M., and Larbes C. (2010). Design and development of a data acquisition system for photovoltaic modules characterization, Renew. Energ., 35:1484-1492.

Benghanem, M. (2010). A low cost wireless data acquisition system for weather station monitoring. Renew. Energ., 35:862-872.

Benghanem, M. (2009). Measurement of metrological data based on wireless data acquisition system monitoring. Appl. Energ., 86:2651-2660.

Bishop, R.H. (2008). Mechatronic System Control, Logic, and Acquisition. 2nd ed. CRC Press, Taylor & Francis LLC, Boca Raton, FL, USA, 754p.

Evjen, B., Hollis, B., Sheldon, B., and Sharkey, K. (2008).Professional Visual Basic 2008. 1st ed. John Wiley & Sons Inc., Hoboken, NJ, USA, 1420p.

Herman, T., Jones, A., MacDonald, M., and Rajan, R. (2008). Visual Basic 2008 Recipes: A Problem-Solving Approach. 1st ed. Apress, New York, NY, USA, 680p.

Ibrahim, D. (2008). Advance PIC microcontroller Projects in C. 1st ed. Elsevier Ltd., Burlington, MA, USA, 544p.

Microchip Technology Inc. (2011). PIC32 Microcontroller Families. Microchip Technology Inc., USA Available from: http://ww1.microchip.com/downloads/en/DeviceDoc/39904k.pdf Accessed date: June 29, 2011.

Microchip Technology Inc. (2010). PIC16F87XA Data Sheet. Microchip Technology Inc., USA Available from: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en010242 Accessed date: October 29, 2010.

Reid F. (2004). Network Programming in Net with C# and Visual Basic .Net. 1st ed. Elsevier Digital Press, Burlington, MA, USA, 541p.

Risø National Laboratory for Sustainable Energy, Denmark. (2009). WAsP 10 Help Facility and On-line Documentation.Technical University of Denmark (DTU), Denmark. Available from: www.wasp.dk. Accessed date: June 29, 2011.

Stephens, R. (2008). Visual Basic 2008 Programmer's Reference. 1st ed. John Wiley & Sons Inc., Hoboken, NJ, USA, 1205p.

Verle, M. (2009).PIC Microcontrollers–Programming in C. 1st ed. MikroElectronika, Belgrade, Serbia, 336p.