

Automatic Temperature Monitoring/Logging System with Computer USB Interface

Iyapo, Kamoru Olarewaju¹; Akinbobola, James²; Odo, Ekundare Ayodele³; Fasunla, Olukayode Michael⁴, Egbuwalo, Shadrach Alaba⁵ & Raimi, Oluwole Abiodun⁶

^{1,3,4}Department of Science Laboratory Technology, Rufus Giwa Polytechnic, Owo, Nigeria

²Department of Electrical Electronic Engineering, Ahmadu Bello University, Zaria, Nigeria

³Department of Physics, Federal University, Oye-Ekiti, Nigeria

⁶Department of Mechanical Engineering, FUTA, Akure, Nigeria

EMAIL: oluwoleaimi15@gmail.com

ABSTRACT: Temperature is one of the most measure weather parameter. In most cases it is sufficient to measure and read out the instantaneous temperature of an environment. For useful analysis and weather prediction however, it is important to see the trend and variation of such measure temperature over a long period of time. This project will involve the design and fabrication of a microcontroller based temperature monitoring and logging system. The microcontroller will acquire digital temperature values from a precision temperature sensor, whose value is then processed and send to computer software through the USB port for logging and future analysis of the measure data. This system can be upgraded to form a full weather monitoring Equipment

Keywords: Temperature, Humidity, Microcontroller, USB port and Logging

Citation: Iyapo, K. O., Akinbobola, J., Odo, E. A., Fasunla, O. M., Egbuwalo, S. A. and Raimi, O. A. (2017). Automatic Temperature Monitoring/Logging System with Computer USB Interface. *Journal of Industrial Technology*, 2 (3): 62-77

1. INTRODUCTION

It is very much essential in laboratories as well as in industries to monitor temperature and humidity continuously. Earlier measurements had been done through manual measurement from analog instruments such as thermometers, manometers and hygrometers. Such type of measurements can't fulfill the current requirements in terms of the time

duration and accuracy. The efficient solution for this problem is to develop a data logger (Muhammad and Janice, 2004). Microcontroller based embedded design for monitoring temperature/humidity has brought a revolutionary change (Kalsi, 1999; Omosule, Olusegun, Abiodun and Feyisetan, 2017; Olalekan and Toluwani, 2017). The stand-alone system allows monitoring and controlling the physical parameters but, not suitable to log

data for a long duration of time. Now a days, PC or laptop becomes very common, cheap and reliable. Further development in data logging took place as people begin to create PC – based data logging system (Andrew *et al.*, 2002).

The processes to collect, analyze and store data for later use is called logging. It is a process to record events during a test or measurement with the use of a system. The human brain and its memory, the nature's creation, no doubt is the best data logging mechanism. Where there is the need to collect information faster than a human, data loggers can possibly collect the information and in cases where accuracy is essential. Data logger is a device that can be used to store or retrieve the data (Peter, 2004; Oyebola, Odueso and Olugbemi, 2017). It also implies the control of how sensor collects and analyzes the data. It is commonly used in scientific experiments and in monitoring systems.

In devising a monitoring system that tracks the entry and presence records of individuals inside a confined space, (Goswami *et al.*, 2014) used microcontrollers and laser sensors to detect the presence of a person and in a triggered event stores the details of the person in form of an image or video. The design showed that every image or video segment has its own tag uniquely identified by the new entrant and updates the pre-existing entrant records.

In a design for temperature and humidity monitoring system based on GSM module, (Wang, 2014) presented a wireless monitoring system through microcontroller coding and then sent to the user via the TC35i GSM network to enable accurate, real-time remote temperature monitoring.

In the design and development of a temperature monitoring and controlling system, (Baruah *et al.*, 2014) used microcontroller ATmega328 and LM35DZ as the temperature sensor. The main target of the system is to have it design and implementation such that it is cost effective. The design showed that the system has various advantages over other similar systems such as been cost-effectiveness, smaller size, on-device display, less complexity and greater portability.

In controlling and monitoring certain weather hazards, (Susmitha and Sowmyabala, 2014) developed an embedded system to design a weather monitoring system which enables the monitoring of weather parameters in an industry. The gas and humidity was monitored using LPC1768 microcontroller program developed in an embedded C using IDE keiluvision4. The design showed that the embedded controlled sensor networks have proven themselves to be a reliable solution in providing remote control and sensing for environmental monitoring systems.

In a design of device of real time weather monitoring, (Patil *et al.*, 2016) presented an experimental design to monitor the temperature and relative humidity of the atmosphere via GSM network, using analog and digital component. The device design showed that it has many advantages compared to other weather monitoring system in terms of its smaller size, on-device display, low cost and portable.

Temperature and humidity are the ever changing parameter because of exposition to huge array of stimuli from environment. One must be careful when measuring temperature to ensure that the measuring instrument is

really the same temperature as the material that is being measured.

2. Design Framework

In an attempt to implement the proposed system, the design process is divided into two: Hardware and Software designs.

2.1 Hardware Design Considerations

This section describes briefly the different hardware used in the design. To ensure accuracy and standard practice the project is design around standard atmospheric sensor used in most weather parameter measurement. Figure 1 shows a graphical illustration diagram of the mini weather monitoring system.

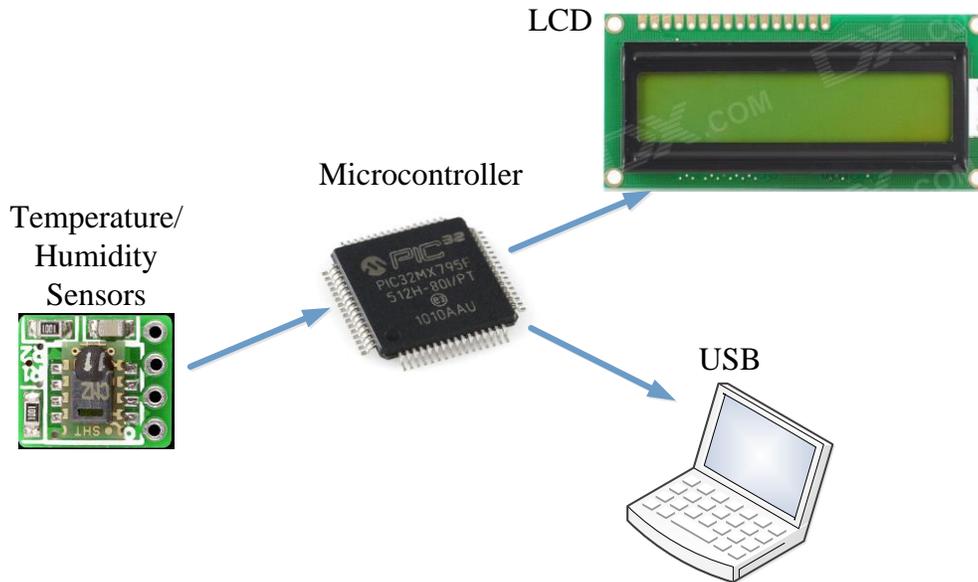


Figure 1: Showing the block diagram of the Temperature/ Humidity monitoring system

2.1.1 Microcontroller and sensor unit

The heart of the mini weather system is the PIC18F4550 microcontroller from Microchip. This microcontroller is connected to the SHT11 sensor via the Data and Clock pins. The controller which is program in C language communicates with the sensor using I2C protocol usually referred as a 2 PIN I2C communication. The received data is then processed, and components representing the atmospheric temperature and humidity are separately extracted. The system also consists of a visual display unit the LCD for immediate assessment of the measured weather parameters. For the purpose of further analysis of the measure parameters a data logger system is provided through a USB connection of the system to a computer where a C# based software is design to capture and save the measured temperature and humidity variables in a text file for future analysis. The following section gives detail description of the USB

hardware feature of the selected PIC microcontroller chosen for this work and the software implementation of the USB protocol needed for satisfying the USB 2.0 data exchange process.

The system is designed around the SHT11 temperature humidity sensors. The Sensirion SHT11 humidity and temperature sensors are single chip modules that provide calibrated digital output over a 2-wire serial interface. These modules are easily connected to a microcontroller, but to achieve full accuracy, floating calculations must be performed. These calculations compensate for the non-linearity of the humidity sensor, convert the data to standard units of temperature and relative humidity, and determine the dew point using a logarithmic equation. Figure 2. Shows the recommended connections required for the Sensirion SHT11 module as applied in this design.

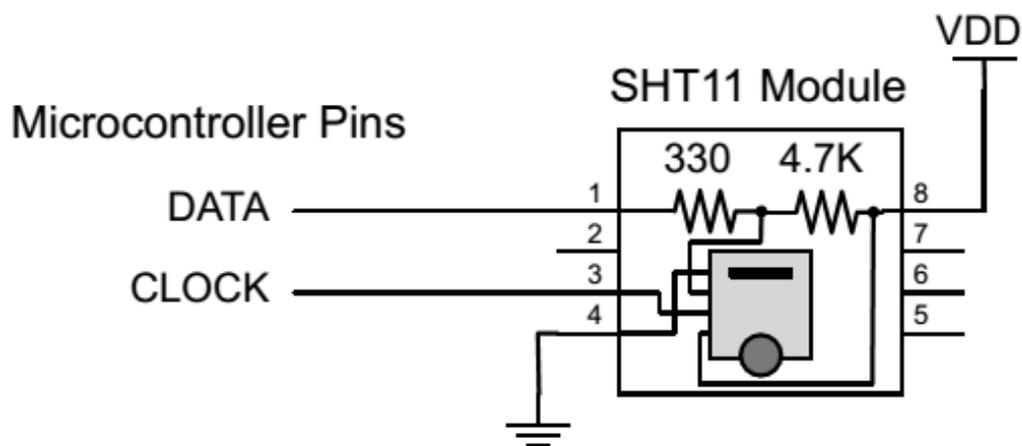


Figure 2: Showing the internal working of the SHT11 temperature/humidity sensor and its interface to a microcontroller unit.

3 System Design and Implementation

This section gives a detail description and experimental procedure used in this project. Celsius scale temperature and humidity (percentage) sensor displays the ambient temperature and humidity through a LCD display. It consists of two sections. One is that which senses the temperature/humidity which is a temperature/humidity sensor SHT11. The other section converts the temperature/humidity value into a suitable

scale which is done by computer software via a USB port. The software was implemented on a windows 7 operating system and was written using Visual studio 2013 from Microsoft. A 5V power supply is derived from the power source with further filtering provided by capacitor C11 and C12. The general circuit diagram of the project is shown in Figure 3. The arrangement is shown in the diagram which links the various sub-units which has been put together.

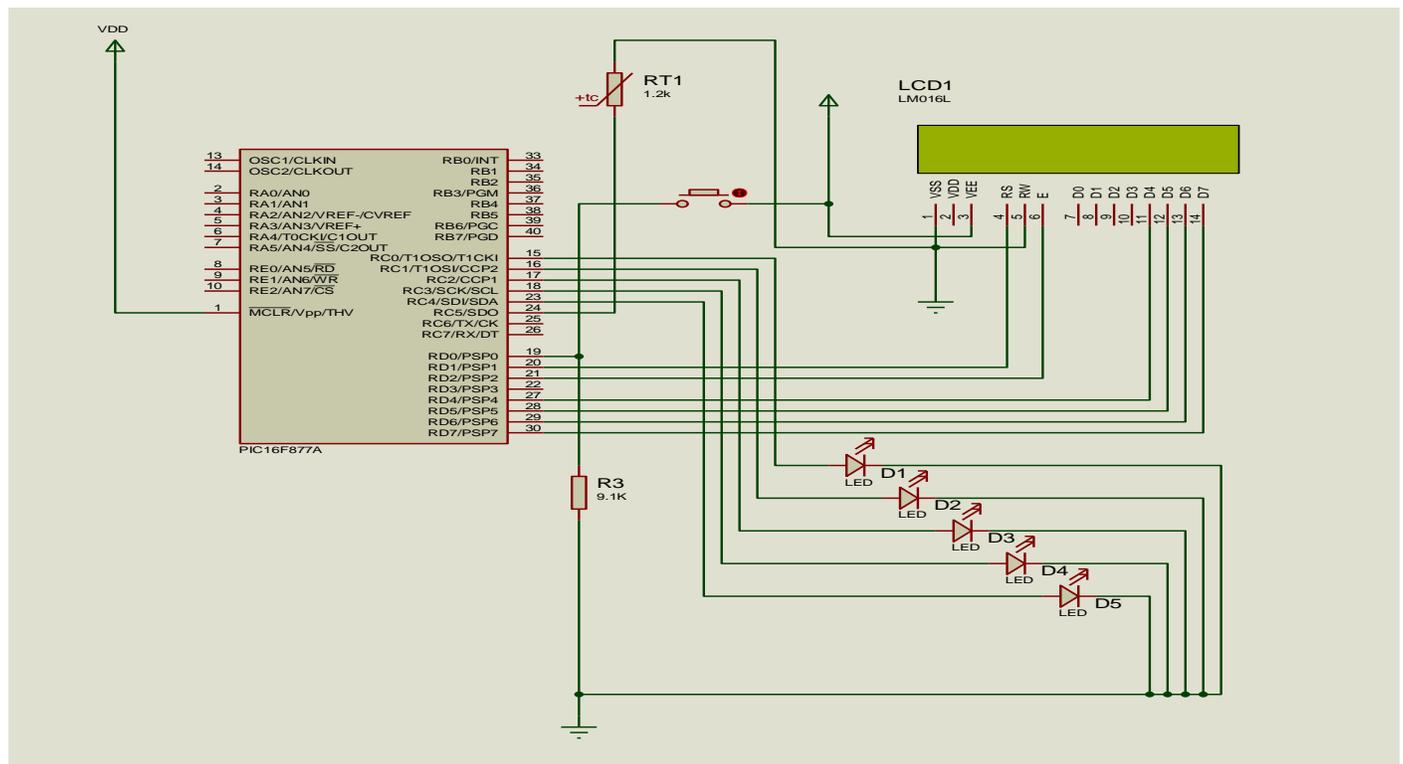


Figure 3: The general circuit diagram of the project.

3.1 Design and Construction

The actual design of the mini weather monitoring system was carried out using

Altium PCB software. It consisted of a filtered 5V power supply derived with further filtering provided by capacitor C11 and C12. A further

3.3V regulated supply derived from the 5V supply was achieved using a bipolar transistor Q1 resistors R5 and R7 and filter capacitors C9 and C10. The design also included an ICSP inputs for the programming of the microcontroller which is a connector attached to RB6, RB7 and the MCLR pin 1 of the microcontroller unit. The design of the microcontroller circuitry follows standard practice to ensure a stable operation of the

circuitry. The microcontroller is clocked by a 20MHz crystal oscillator and 22pF capacitor banks which ensure a stable 20MHz clock later scaled using a PLL inside of the microcontroller to the desired 48MHz required for the operation of a USB device. Also implemented is the LCD circuit which includes a 16 PIN header connector driven by the PORTD of the microcontroller unit.

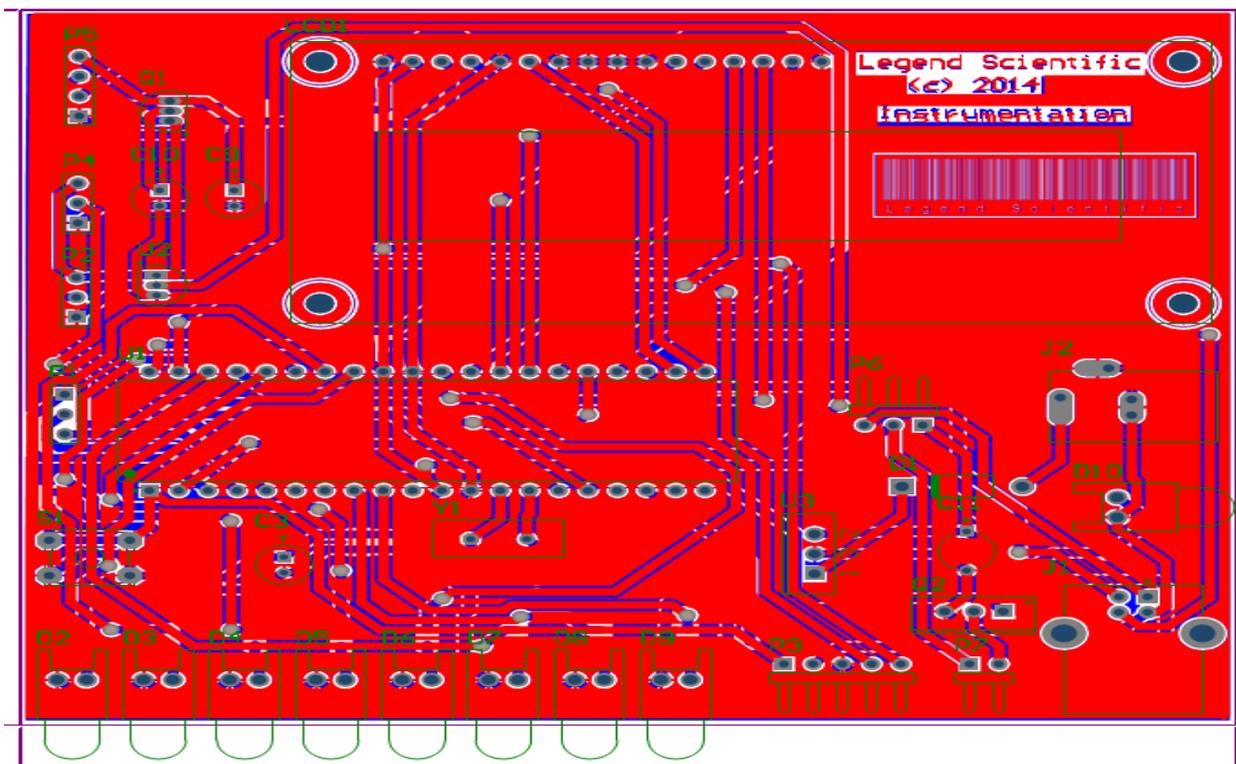


Figure 4: PCB board layer derived from the design using Altium PCB.

The circuit also includes a 5 PIN header connector for the SHT11 sensor connection. This circuit was compiled and checked for errors before being transferred to a PCB layout for onward processing as a PCB file. On the PCB layout, the different component footprints were

carefully placed on a dimension of 80 X 100 mm. The placement was done to ensure the shortest possible distance between related components. An electrical rules violation check (ERC) was performed on the design, and the placed footprints were checked before proceeding with the

routing. The entire component electrical connection was then auto routed using the routing component of Altium software. Figure 3 shows the final PCB layout drawing of the design implemented for a dual layer PCB board. Figure 3 shows the 3D image of both layer of the finished PCB design as derived from the 3D view of the Altium software. The actual implementation of the PCB design was achieved using the tonner.

3.2 Design of Power Supply Module

The power supply for this project is a single 9V supply obtained from 240v/12v step down transformer. This transformer is selected to deliver power at unity power factor. It also contained DC section connected to the AC session with automatic change over. The particulars of the ac section include:

Transformer primary	=	240v
Transformer secondary	=	12v
A.C Current Ia.c	-	300mA

With reference to transformer ratio equation, it then becomes

$$E1/E2 = N1/N2 = K$$

$$\text{Then } K = N1/N2 = 240/12 = 20$$

The DC current is given by $I_{dc} = V_{dc}/R_L$ where the $V_{dc} = 17v$ and $R_L = 1K\Omega$

$$\text{Then } I_{dc} = 17/1000 = 17mA$$

Meanwhile, while designing the power supply unit, the following characteristics were put in to consideration.

- Regulation
- Maximum efficiency
- Ripple factors
- Efficiency

3.4 Rectifier Selection

The diode peak inverse voltage (PIV) is obtained from

$P_{IV} = 2 \times V_{dc} = 2 \times 17 = 34 v$ The P_{IV} is the diode maximum reverse voltage above which breakdown may occur causing current to flow in the reverse direction thereby damaging the diode. For the each diode in full wave bridge rectifier circuit must have P_{IV} rating of twice value of the peak voltage developed at the output. For this project, a general purpose rectifier diode IN 4002 was selected.

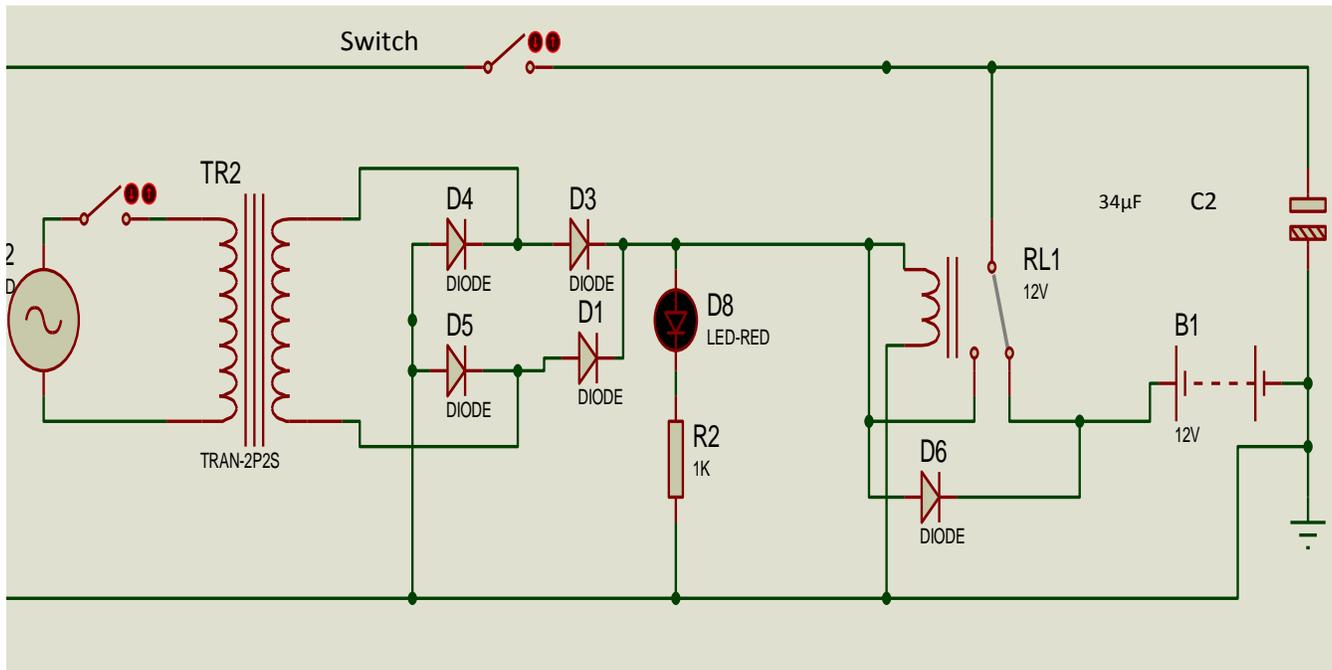


Figure 5: Shows the power Supply

3.5 Filtering /Smoothing Capacitor Selection

The choice of smoothing capacitor can be obtained from supply frequency of 50HZ, the input voltage is given by:

$$V_{\text{RIPPLE}} = V_{\text{PP}} - V_{\text{RMS}} \quad (1)$$

Where V_{pp} is the peak voltage and V_{rms} is the secondary voltage which is 12V

$$\text{Then } V_{\text{ripple}} = I_{\text{dc}} / 2fc \quad (2)$$

$$C = I_{\text{dc}} / 2fV_r = (17 \times 10^{-3}) / (2 \times 50)(17 - 12) = 34 \mu\text{F}$$

3.6 Programming the Microcontroller

This review the programming of the microcontroller used in the design of this project.

Ccs Usb Api Software Development

The CCS C Compiler provides an API for developing USB applications on the PIC. There are several files associated with the USB API

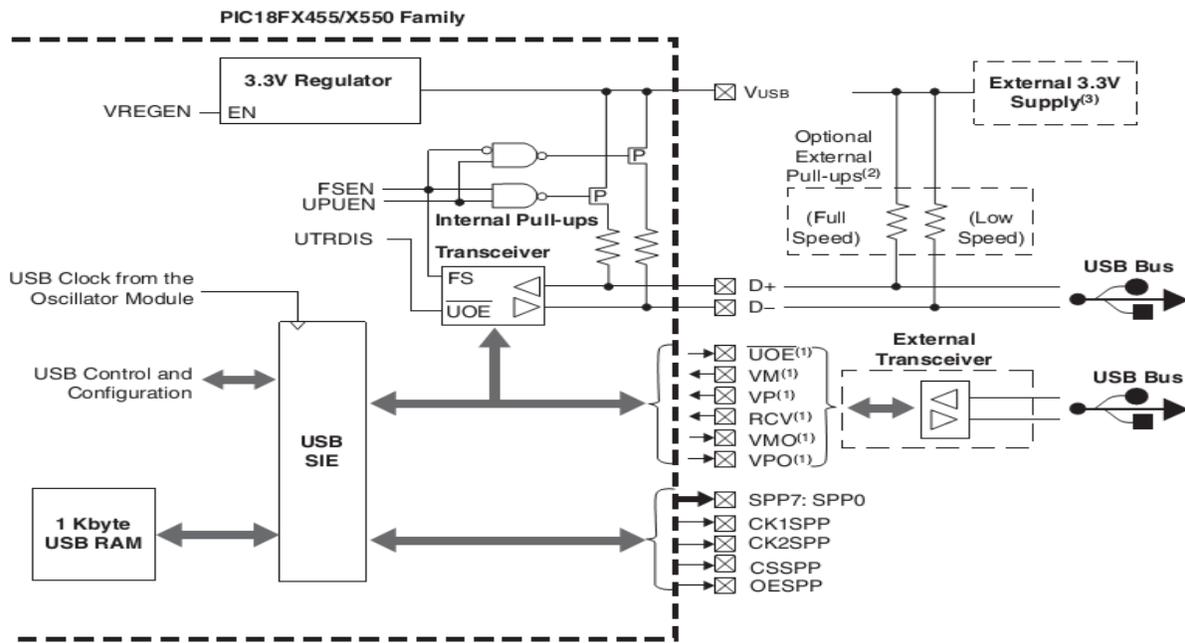
that were used in this project: `pic18_usb.h` - Provides hardware layer functions for Microchip PIC18 microcontrollers that have a built-in USB peripheral, such as the PIC18F4550. This provides functions for setting up the peripheral, sending and receiving packets, etc. `usb.c` - Using an interrupt driven by the hardware, this provides token handler support. The bulk of this file automatically handles and responds to the SETUP tokens used by the PC to configure the USB device. There are some global definitions made here that can be changed to alter the USB API to an application. `usb_desc_*.h` - Provides example device, configuration, interface, endpoint and string descriptors needed to describe the USB devices used in the examples. The SETUP handler in `usb.c` will use these descriptors to

answer Get Descriptor requests. Since USB descriptors are application dependent, there is a different descriptor for each example program.

Pic18 Microcontroller Usb Interface

Some of the PIC18 microcontrollers support USB interface directly. For example, the PIC18F4550 microcontroller contains a full-speed and low-speed compatible USB interface that allows communication between a host PC and the microcontroller. Figure 6 is an overview of the USB section of the PIC18F4550 microcontroller. PORTC pins RC4 (pin 23) and

RC5 (pin 24) are used for USB interface. RC4 is the USB data D- pin, and RC5 is the USB data D+. Internal pull-up resistors are provided which can be disabled (setting UPUEN = 0) if desired and external pull-up resistors can be used instead. For full-speed operation an internal or external resistor should be connected to data pin D+, and for low-speed operation an internal or external resistor should be connected to data pin D-. Operation of the USB module is configured using three control registers, and a total of twenty-two registers are used to manage the actual USB transactions.



- Note 1: This signal is only available if the internal transceiver is disabled (UTRDIS = 1).
 Note 2: The internal pull-up resistors should be disabled (UPUEN = 0) if external pull-up resistors are used.
 Note 3: Do not enable the internal regulator when using an external 3.3V supply.

Figure 6: The PIC18F4550 microcontroller USB built in hardware structure. (2009 microchip Tech)

3.7 Firmware Software Development

The firmware which is the instruction set that controls the entire functionality of the

PIC18F4550 and how it communicate with and transfer the measured data between the SHT11 sensor, the LCD display system and the computers via its USB port was writing using

C++ language with a CCS C compiler. Figure 7 shows a flow chart describing the basic program design for the implementation of the project

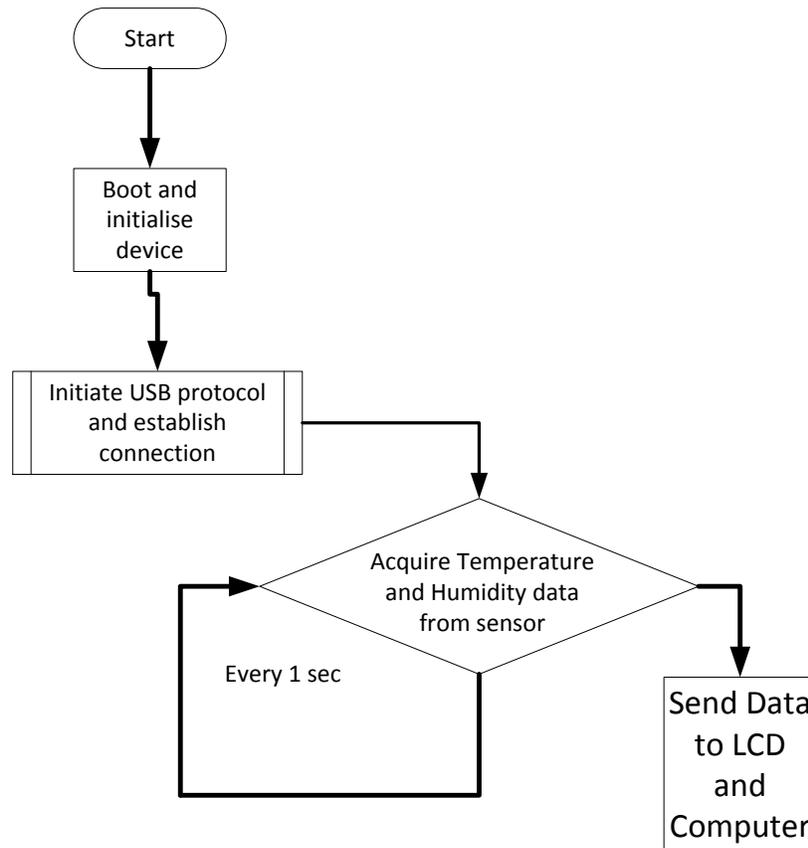


Figure 7: A flow chart describing the operation of the system

The system module receives power supplies from the USB port and as such the basic operation starts with the connection of the module to the USB port of the host computer. Upon been connected to its host computer the system immediately receives power and initializes its USB hardware. Next,

the module sends acknowledge message to the computer system for proper connection and then established an agreed speed of transfer and reception of messages from its host computers as well as handling of both hardware interrupt and software interrupt service routine embedded in the processors. At a regular interval of 1 second the processor

acquire a measured temperature and humidity data from the SHT11 sensor, this is then processed and channel via the USB for onward transmission to the computer. The system also runs at the background a graphics driver for displaying the acquired measured value on the attached LCD screen. The entire firmware code use for the implementation of the algorithm described here can be found in the appendix.

3.8 User Interface Software Development

The computer software used for the capturing of the measured data via USB port was implemented on a windows 7 operating system and was written using Visual studio 2013 from Microsoft. Figure 8 shows the Graphical User Interface (GUI) which the user uses to establish connection to the hardware and the subsequent reception of transmitted measured value from the hardware.

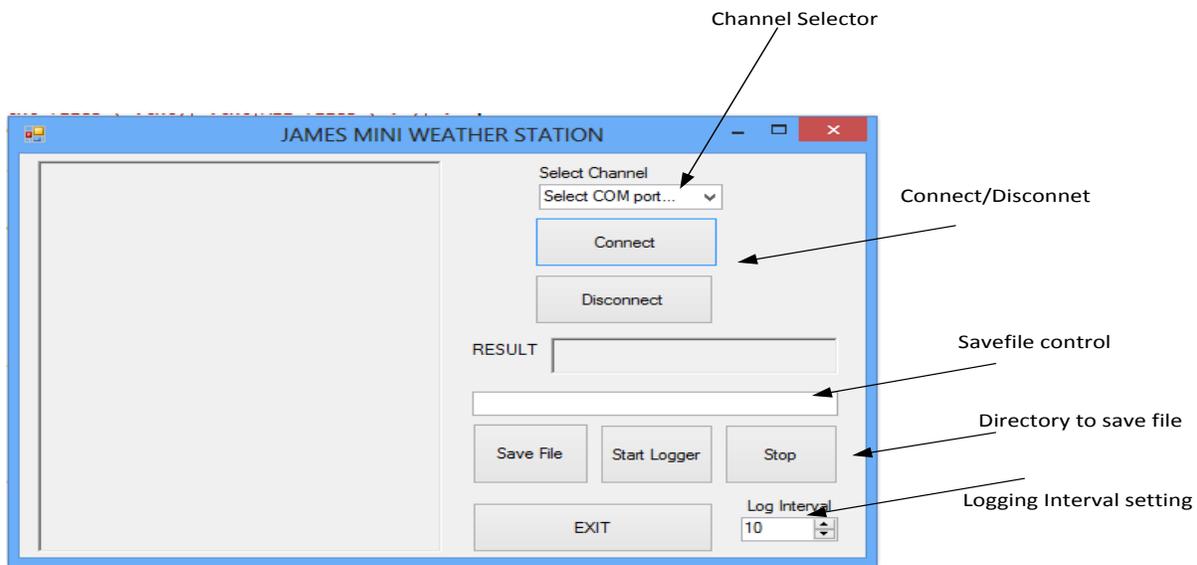


Figure 8: C++ Graphical User Interface (GUI) with all the necessary buttons and panel to monitor data log captured from the device via USB.

The GUI is divided into two main sections (a) the setting plane and (b) the result plane. In order to establish connection to the hardware module the user need to select a communication channel number corresponding to a virtual serial port created by the firmware of the PIC18F4550 whose driver have been preinstalled on the host PC.

After selecting the given channel number the user then click on the connect button to establish a USB connection to the hardware. Once a connection is established the GUI starts to display the received data every second on the result plane. To enable future analysis on the acquired data an option for data logging the acquired result is also presented as part of the

GUI. The User selects a directory to save the acquired data and then start the logging at an

interval indicated by the log interval value.

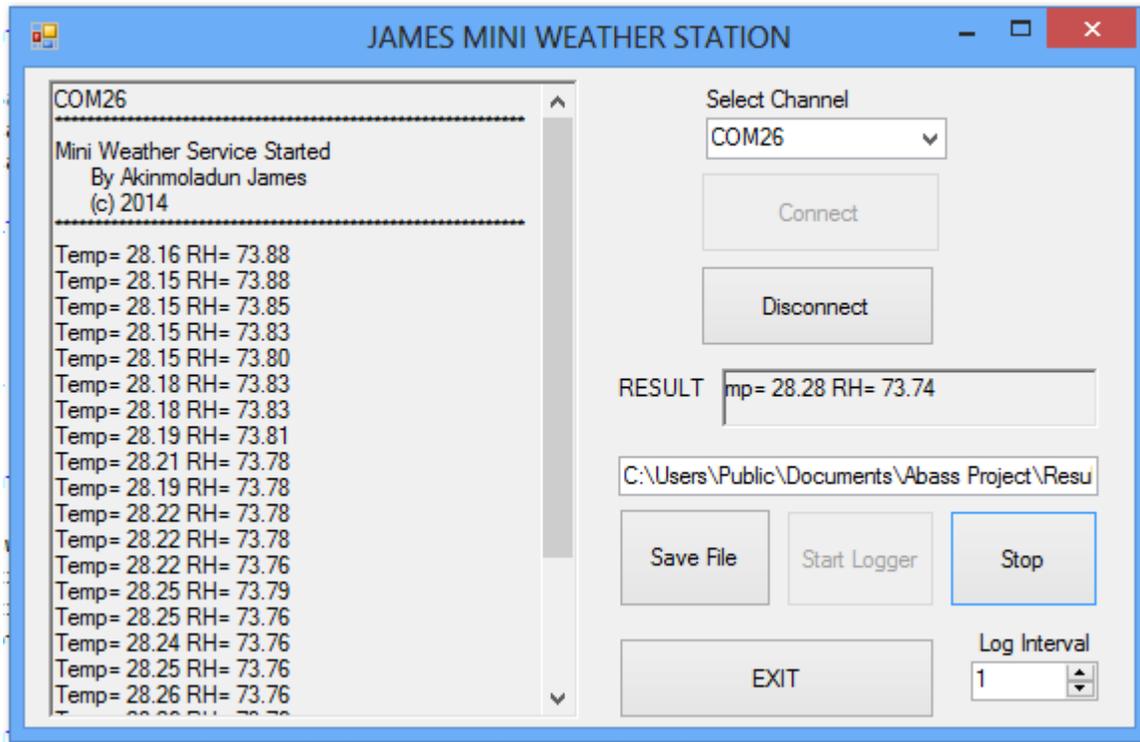


Figure 9: C++ GUI in operation with the panel displaying the monitor and data log captured from the device via USB.

The code snippet below shows part of the program code used to achieve the logging aspect of this project. For detail please refer to the entire GUI software code at the appendix.

4 Construction

The physical implementation of the design was carried out on the manufactured PCB boards using mainly surface mounted components. The PCB was achieved through the tonner transfer method, where the image of the PCB design was printed on a glossy paper and then transferred

onto the copper plated board by means of a heated surface on which pressure was applied. Etching of the board was achieved using concentrated Hydrochloric Acid and Hydrogen Peroxide. The components were then soldered into the respective component slots using a 30W soldering Iron. Initial testing of the board was carried out using continuity meter to ensure all short circuit faults are properly cleared. Figure 10 shows the image of the finished project connected and powered through a USB cable.



Figure 10: Showing the finished board with all component assembled and tested with temperature reading of 30.2°C and Humidity of 71.2%

4.1 Testing And Result

Two major tests were carried out during the project phase and they are:

- i. Component test,
- ii. System test.

Component Test

All the components incorporated in this project was tested using multimeter before construction in order to ensure good component and exact one's are used.

System Testing

The system module receives power supply from the USB port and as such the basic operation start with the connection of the module to the USB port of the host computer upon was connected to its host computer, the system immediately receives power and utilizes its USB hardware.

The implemented design was initially tested on an open area after construction and the system perform as expected. The device was further install in a household to measure the temperature and relative humidity of the house. Figure 11 shows the screen capture of the experiment and the result as saved in the file.

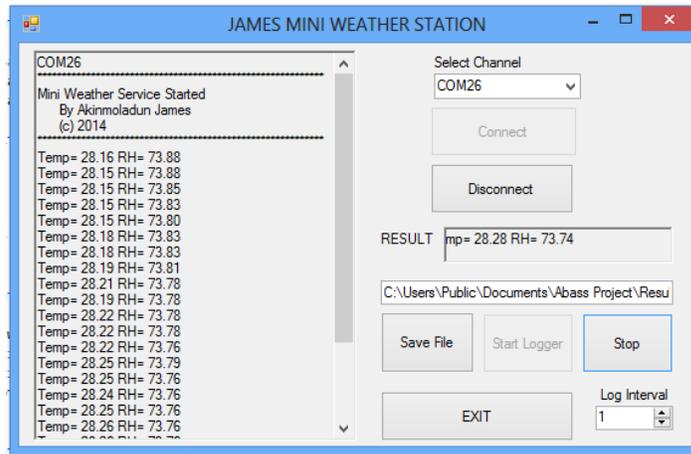


Figure 11: shows the screen capture of the experiment and the result as saved in the file.

Results

The readings are taken under different conditions for some time interval. There is no stable environment where temperature/humidity are relatively constant therefore, it is not easy to check for correctness for repeatability of the thermometry and hygrometry system but temperature was checked for correctness in the value reading with standard mercury in glass thermometer these was shown on Table 1 below. Therefore, it was found that correlation of dry

thermometer unit was 0.99 against standard mercury in glass thermometer. The correlation were much closed to unity therefore, the sensor linearity for the system were very good. Also, Table 1 shows the data collected by developed instrument and the instrument at (certified) weather station were compared. The values were in completely in agreement with each other, which have also confirmed correctness and accuracy of the instrument.

Table 1: Standardization of the developed, dry and wet thermometer (CERT weather station)

Time	Dry Bulb	Wet Bulb	Designed
8am	24.5	21.8	24.5
9am	26.5	22.5	27.5
10am	28.5	24	28.16
11am	29	24.5	28.25
12am	31.5	25	28.26
1pm	32.5	25.5	32
2pm	33.5	26	33

3pm	34.5	26.5	35.5
-----	------	------	------

5 Problem Encountered/Limitation

The problem encountered was the final packaging of the design, as this actually caused problem on the circuit board. Such problems include partial contact within the circuit board, between components and also with the wiring. However, this project cannot measure temperature below 2°C because the SH11 sensor chosen for this project can only sense and read temperature between the ranges of 2°C to 150°C from its design specification.

5.1 Conclusion

The aim of this project is to design a microcontroller based temperature and humidity measurement device which is of less cost, portable, very low power consumption, self-contained. The instrument is capable of sensing and measuring change in temperature in degrees and relative humidity in percentage. A step-by-step approach in the designing has been followed. The measurement of temperature and relative humidity of a place can be achieved when the sensor is moved closer to the vicinity of the place to be measured and the equivalent value will be displayed on the LCD. This instrument can be used as a temperature and humidity sensor in Nigerian research Reactor (NIRR -I) control systems, for monitoring temperature and humidity in order to ensure the reaction occur

within safe limits, weather stations and also for domestic appliances. The instrument cost of production is cost effective compared to present analog monitor on NIRR console which cost around 10,000.

Recommendation

Future implementation of this project should be done to support higher temperature as the maximum limit reached by this design is 150°C.

REFERENCES

- Andrew, J. T., John, L. B., and Neil, R. T., (2002): "Low power data logger", *Proceedings of Conference Department of Physics, University Of Otago*
- Baruah, J., Islam, B., and Zaman, S., (2014). Microcontroller based temperature monitoring and controlling system. *International Journal of Advanced Information Science and Technology (IJAIST)*, 26(26), 117-121
- Goswami, A., Mukhopadhyay, A., and Sengupta, S., (2014). Micro-controlled automated smart tag-based attendance register (M.A.S.T.A.R). *International Journal of Computer Applications, International Conference on Microelectronics, Circuits and Systems (MICRO-2014)*, 6-9
- Kalsi .H.S., (1999): "Electronic instrumentation", Tata McGraw-Hill Ltd., New Delhi, 1999.
- Muhammad, A. M., and Janice, G., (2004): "The 8051 microcontroller and embedded systems", Pearson education ltd., India.
- Olalekan, O. B. and Toluwani, O. V. (2017). Remote Control of Electrical Loads Using

- GSM Module. *Journal of Industrial Technology*, 2 (2): 38- 50.
- Omosule, N. I., Olusegun, A. S., Abiodun, R. O. and Feyisetan, F. O. (2017). Evaluation on the Challenges of Modern Car Maintenance in Nigeria. *Journal of Industrial Technology*, 2(3): 51- 61.
- Oyebola B. O., Odueso V. T. and Olugbemi S. A. (2017), Study on Biochips Technology. *Journal of Industrial Technology*, 2(1):29-37.
- Patil, K., Mhatre, M., Govilkar, R., Rokade, S., and Gawas, G., (2016). Weather monitoring system using microcontroller. *Journal on Recent and Innovation Trends in Computing and Communication*, 4(1), 78-80
- Peter, R., (2004): "Using data loggers", science teachers' workshop, 2004, North Sydney
- Susmitha, P., and Sowmyabala, G., (2014). Design and implementation of weather monitoring and controlling system. *International Journal of Computer Applications*, 97(3), 19-22
- Wang, X., (2014). Temperature and humidity monitoring system based on GSM module. *International Journal of Computer, Consumer and Control (IJ3C)*, 3(1), 41-49

Appendix

```
private void btnSaveFile_Click(object sender,
EventArgs e)
{
SaveFileDialog saveFileDialog1 = new
SaveFileDialog();
saveFileDialog1.Filter = "txt files
(*.txt)|*.txt|All files (*.*)|*.*";
saveFileDialog1.RestoreDirectory = true;
if (saveFileDialog1.ShowDialog() ==
DialogResult.OK)
{
txtSaveFile.Text = saveFileDialog1.FileName;
}
}
private void btnLogger_Click(object sender,
EventArgs e)
{
sw = new StreamWriter(txtSaveFile.Text);
timer1.Interval =
Convert.ToInt16(1000*cntLogger.Value);
timer1.Enabled = true;
btnLogger.Enabled = false;
}
private void btnStop_Click(object sender,
EventArgs e)
{
// close the stream
```