Advance Journal of Food Science and Technology 6(11): 1241-1248, 2014
DOI:10.19026/ajfst.6.190
ISSN: 2042-4868; e-ISSN: 2042-4876

Submitted: July 01, 2014 Accepted: September 13, 2014 Published: November 10, 2014

Research Article

Design and Implementation of Food Monitoring System Based on WSN

Kong Xiangsheng
Xinxiang University, Xinxiang, China

Abstract: This study describes a design and implementation of the Internet of things technology, aiming to design and realize an intelligent food monitoring system based on a wireless communication technology. A remote wireless monitoring system is proposed for food supply network based on Zigbee and RFID which are mainly used to detect and gather food supply information and upload information to monitoring center and download orders to coordinators. In this study, we propose a low-complexity, low-cost, low-data-rate and low-power-consumption design principles and food data clustering approach for WSN. Further, we propose several key issues that affect the practical deployment of gathering techniques in intelligent food monitoring system.

Keywords: Food monitoring system, RFID, wireless sensor network, ZigBee

INTRODUCTION

Today's agricultural information is very dynamic and turbulent. Traditional agricultural information systems have mostly been implemented upon hierarchical architectures, which are inflexible to adapt changes and uncertainties promptly. As more and more people are using the smart devices, more and more consumers can check food's status information by smart device, next-generation agricultural information systems must be agile and adaptable to accommodate changes without significant time delays. It is essential for an agricultural information system to obtain real-time status from the distributed and dynamic agricultural environment for decision making. ZigBee, Wireless Sensor Network (WSN) and Radio-Frequency Identification (RFID) technology provide an excellent infrastructure for data acquisition, distribution and processing.

The goal of the ZigBee is to provide the consumer with ultimate flexibility, mobility and ease of use by wireless intelligence and capabilities into everyday devices (Liu *et al*., 2013). Therefore, ZigBee is suitable for food monitoring system. The IEEE 802.15.4/ZigBee standard is a popular technology in the area of low data rate, low cost, low power consumption and security-oriented WSNs. Although the ZigBee standard is well designed, many features related to the WSNs are still challenging, especially for cluster-tree topologies where it is possible to achieve semi-deterministic behavior.

The RFID system consists of a reader and a tag and the UHF. RFID is classified into passive and active types according to the method used to obtain the energy source of the electric wave. The passive type obtains the transmission energy from the electric wave received from the reader. The active type obtains the transmission energy from the battery. The passive tag has a semi-permanent life, no battery, is inexpensive and is suitable for short range communication. On the other hand, the active tag can be used for long range communication because the battery is built into the tag. The active RFID can provide long range data transmission because it uses different fields. The requirement for food history management systems is growing due to the increasing importance of food safety problems. The freshness of vegetable, meat, or dairy product is very important. An active RFID tag is attached to the product that requires refrigeration and the temperature/process of distribution is monitored so that product spoilage can be prevented.

Wireless Sensor Networks (WSNs) are modern networks used in many industrial and agricultural application areas. WSN refers to a wireless network that consists of a large number of sensor nodes which is deployed in the detection region. The wireless sensor network is a self organized distributed network and composed of a large number of tiny sensor nodes with wireless communication capacity and computing power. It can monitor, track, perceive and cluster the various environment or the information parameters of the detection objects and it also has the abilities of automatic control, remote monitoring and data computation. Sensor node is equipped with integrated sensors, data processing capabilities, data gathering capabilities and short range radio communications. Sensor nodes are spread randomly over the deployment region. Sensor networks are being deployed for a wide variety of applications including sensing, tracking, monitoring. The collected data of the member nodes are processed in the head cluster before being send to the base station and after collecting the data and in the steady state phase in the shape of a packet are send to the base station in a direct way (Bal and Rath, 2013).

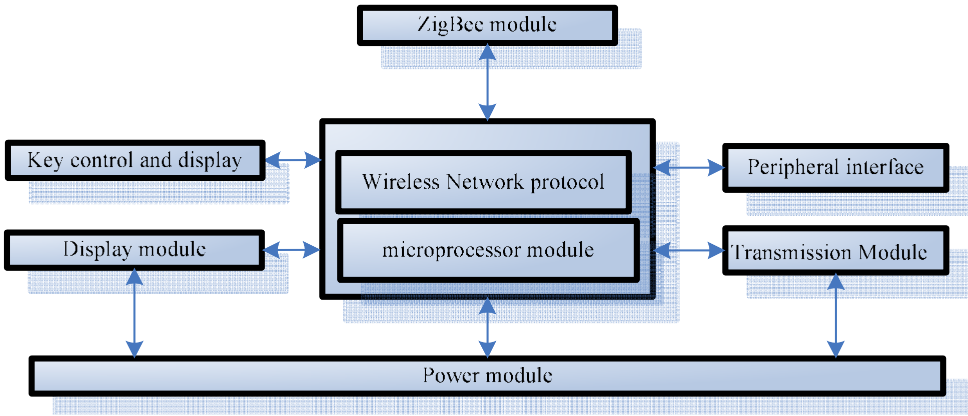

Fig. 1: The hardware structure of WSN

This study proposes a smart WSN-based food monitoring system composed of ZigBee Module, Sensor Module, Microprocessor Module, Power Module and monitoring algorithms. Milk which is a food that many people consume at all times is susceptible to storage condition. To validate the effectiveness of the proposed system, we were monitoring the freshness of milk by the environmental. By temperature sensor, humidity sensor and pH sensor, we measured temperature, humidity and the pH of milk and analyzed the factors affecting the freshness of milk.

MATERIALS AND METHODS

Hardware design of food monitoring system: As a hardware basic unit, wireless network nodes consists of the following components: the master control module, wireless transceiver module, camera module, key control and indicate module, displays module, peripheral interface, ZigBee module and power module. In practical applications, the master equipment consists of all parts except display module; a wireless route not includes cameras and display interface. User terminal not include cameras module. Hardware Design of wireless sensor networks based on ZigBee in this study consists of a sensor module, a transmission module, a microprocessor module and a display module (Fig. 1).

ZigBee module: ZigBee's general characteristics include use of dual PHY (2.4 GHz and 868/915 MHz) with Data rates of 250 Kbits/s (@ 2.4 GHz), 40 Kbits/s (@ 915 MHz) and 20 Kbits/s (@ 868 MHz). The protocol is optimized for low duty cycle applications (<0.1%). CSMA-CA channel access yields high throughput and low latency for low duty cycle devices like sensors and controls. It also provides for an optional guaranteed time slot for applications requiring low latency as well as for low power usage with battery life ranging from multi-month to years (Xu *et al*., 2013).

ZigBee allows for multiple topologies including star, peer-to-peer and mesh and is a full handshake protocol for transfer reliability with a typical range of 50 m (5-500 m based on environment). ZigBee uses spread-spectrum technologies to avoid multi-path fading and increase robustness. This approach allows for improved signal immunity in the presence of radio interference.

The IEEE 802.15.4 standard defines two PHYs representing three license-free frequency bands that include sixteen channels at 2.4 GHz, ten channels at 902 to 928 MHz and one channel at 868 to 870 MHz. The maximum data rates for each band are 250 Kbits/s, 40 Kbits/s and 20 Kbits/s, respectively. For the 2.4 GHz PHY, the standard specifies how the data coding, spreading and modulation must be performed. Starting from the raw baseband bit stream, bits are examined by groups of four bits. Each four-bit sequence is mapped to one symbol out of 16 possible symbols. Each symbol is in turn mapped to a 32-chip sequence. These sequences are pseudo-random and they are nearly orthogonal. For commercial applications, ZigBee (IEEE 802.15.4) in particular has great potential in the area of wireless sensor networks and ZigBee networks can incorporate a variety of topologies including Wireless Sensor Network (Fig. 2).

Sensor module: The farmland information acquisition and transmission system collects the humidity, temperature and light intensity data information of the farmland. Due to the monitored area in the farmland and the conditions are relatively poor, there are many uncontrollable factors. Therefore, it is needed to consider the various factors when selects suitable sensor. The sensor should have excellent anti-interference and be waterproof and heat resisting. It also should be small in size and easy to integrate and low power to extend the service life. In conclusion, after analysis and comparison of various sensors, the SHT10 temperature and humidity sensor and the TSL2561 light intensity sensor are selected to measure the temperature, humidity and light intensity of the farmland environment respectively (Gupta and Saini, 2013).

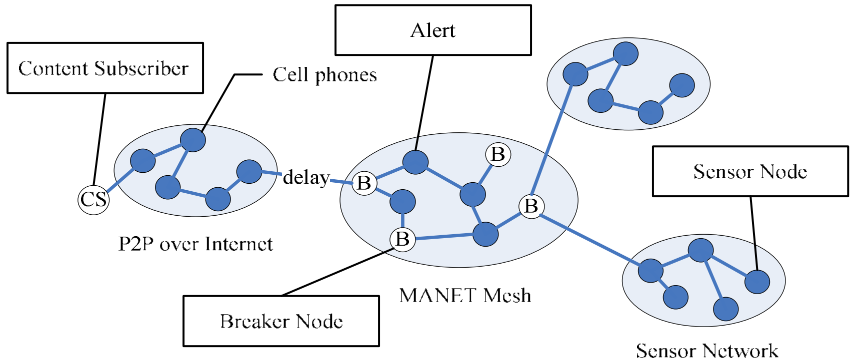

Fig. 2: WSN can incorporate a variety of technologies

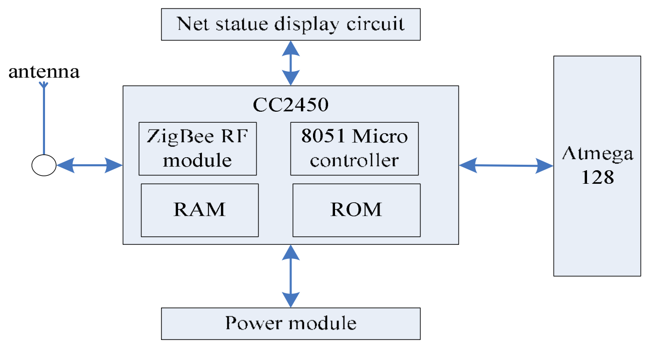

Fig. 3: Gateway node with ATmega128

Microprocessor module: In this study, MEGA128 as a microcontroller, it has a variety of enhanced power management mode, in standby mode, the current consumption is only 0.1 microampere and can work normally from 2v to 5.5v. And MEGA128 suit for battery-driven, have 2 Synchronous serial port supporting SPI and I2C, 2 USART Asynchronous serial port supporting LIN protocol and 64 KB flash. Processing capacity can meet the system requirements of the gating. The diagram of gateway node with ATmega128 is shown in Fig. 3.

Transmission module: A possible way to save resources is the improvement of data transmission. In a wireless sensor network, data is transmitted together with its meta information resulting in relatively big messages; sometimes including redundant information (e.g., data source). Those messages need more resources, especially energy, for transmissions due to long periods of full radio activity.

RF transceiver chip have amount of type and quantity, 433 MHz, 868 MHz and 2.4 GHz are its operating band, the 2.4 GHz is used by ZigBee (Hashemi *et al*., 2013). Currently, Chipcon, Freescale and other companies are focused to develop chip working at 2.4 GHz for ZigBee. Choose the ideal wireless transceiver chip can reduce development effort and shorten the development cycle, reduce costs, bring products to market faster considering the cost and other factors. In this study, we choose the wireless transceiver chip MC13192 of Freescale company. MC13192 requires minimal peripheral components, including the oscillator clock circuitry, the RF input/output matching circuit and microcontroller interface circuit. The chip local oscillator signal is provided either by external active crystal or by internal circuitry. Oscillator signal of internal circuitry need the external crystal oscillator and two load capacitors, the capacitance depends on the crystal frequency and the input capacitance and other parameters.

Power module and others: The sensor nodes can be powered from energy storage devices or by energy scavenging. The former technique employs a variety of tiny batteries made up of thin films of vanadium oxide and molybdenum oxide. The battery supplies power to the complete sensor node. It plays a vital role in determining sensor node lifetime. The amount of power drawn from a battery should be carefully monitored. Sensor nodes are generally small, light and cheap, the size of the battery is limited. AA batteries normally store 2.2 to 2.5 Ah at 1.5 V.

From an energy perspective, the most relevant kinds of memory are the on-chip memory of a microcontroller and Flash memory-off-chip RAM is rarely, if ever, used. Flash memories are used due to their cost and storage capacity (Fu *et al*., 2013). Memory requirements are very much application dependent. Two categories of memory based on the purpose of storage are: user memory used for storing application related or personal data and program memory used for programming the device. Program memory also contains identification data of the device if present.

The sensor node consumes power for sensing, communicating and data processing. More energy is required for data communication than any other process. The energy cost of transmitting 1 Kb a distance of 100 m (330 ft) is approximately the same as that used for the execution of 3 million instructions by a 100 million instructions per second/W processor. Power is stored either in batteries or capacitors. Batteries, both rechargeable and non-rechargeable, are the main source of power supply for sensor nodes. They are also classified according to electrochemical material used for the electrodes such as NiCd (nickel-cadmium), NiZn (nickel-zinc), NiMH (nickel-metal hydride) and lithium-ion.

Current sensors are able to renew their energy from solar sources, temperature differences, or vibration. Two power saving policies used are Dynamic Power Management (DPM) and Dynamic Voltage Scaling (DVS) (Ahmed *et al*., 2013). DPM conserves power by shutting down parts of the sensor node which are not currently used or active. A DVS scheme varies the power levels within the sensor node depending on the non-deterministic workload. By varying the voltage along with the frequency, it is possible to obtain quadratic reduction in power consumption.

Software design of food monitoring system: Wireless sensor networks can be divided into centralized and distributed system. In a centralized system a single element is responsible for gathering and processing data. So, all components of the system are connected to this single element. In a distributed control system the connections between nodes and the information processing is distributed among the system components.

Sensor nodes: A sensor node, also known as a mote is a node in a wireless sensor network that is capable of performing some processing, gathering sensory information and communicating with other connected nodes in the network. A mote is a node but a node is not always a mote.

Sensor nodes can collect and filtration data actively, but also can receive and execute the coordinator instruction. The sensor node was to have an x and a y coordinate the idea of using a structure. The structure members consisted of the node identification number (nodeID) as well as the X-coordinate (x_co) and Y-coordinate (y_co).

To get an ID number and set of coordinates for each of the one hundred sensor nodes, the use of a for loop was used. Also each node and its information were stored in an array of the structure:

```
for (i = 1 ; i< = no ; i++)
{
x_coordinate = Math.random() * x_co
y_coordinate = Math.random() * y_co
nodeID[i] = i
x[i] = xco
y[i] = yco
}
```

In the "for" loop, "no" used as number of nodes in the network area. Also, the coordinates are coded to exceed the network area which is 10 m×10 m. The results of this coding are displayed with two resources in simulation result. The first is the output of the program showing the node ID as well as the random generated x and y coordinates.

Software design of WSN: Software development for wireless sensor networks requires novel programming paradigms and technologies. Sensor node and network coordinator software is implemented in the TinyOS environment. TinyOS is a lightweight open source operating system for wireless embedded sensors. It is designed to use minimal resources and its configuration is defined at compile time by combining components from the TinyOS library and custom-developed components. Well-defined interfaces are used to connect and define the data flow between components. A TinyOS application is implemented as a set of component modules written in nesC. The nesC language extends the C language with new support for task synchronization and task management. This approach results in a natural modular design, minimal use of resources and short development cycles. TinyOS fully supports the Tmote sky platform and includes library components for the Chipcon CC2420 radio drivers and other on-chip peripherals. Radio configuration, MAC layer communications and generic packet handling are also natively supported.

A nesC component exposes a set of interfaces. An interface consists of a set of methods. A method is known as either a command or an event. The component implement that provides methods and expects other components to implement that uses methods (Gu *et al*., 2013). A nesC component is either a configuration that contains a wiring of other components, or a module that contains an implementation of its interface methods. A TinyOS program consists of a set of nesC components, where the top-level file that describes the application is a nesC component that exposes no interface methods.

Figure 4a shows a TinyOS program called SenseToLeds that displays the value of a photo sensor in binary on the LEDs of a mote. SenseToLeds contains a wiring of the components Main, SenseToInt (Fig. 4b),

```
configuration SenseToLeds {
} implementation {
components Main, SenseToInt,
IntToLeds, TimerC,
DemoSensorC as Sensor;
Main.StdControl -> SenseToInt;
Main.StdControl -> IntToLeds;
SenseToInt.Timer ->
TimerC.Timer[unique("Timer")];
SenseToInt.TimerControl ->
TimerC;
SenseToInt.ADC -> Sensor;
SenseToInt.ADCControl ->
Sensor;
SenseToInt.IntOutput ->
IntToLeds;
}
```

(a)

```
module SenseToInt {
provides {
interface StdControl;
}
uses {
interface Timer;
interface StdControl
as TimerControl;
interface ADC;
interface StdControl
as ADCControl;
interface IntOutput;
}
} implementation {
...
}
```

(b)

Fig. 4: nesC source code

IntToLeds, TimerC and DemoSensorC. These components are just a few of the nesC components that are available in the TinyOS library.

Data Records are constructed in the following way: Periodically the sensor node's hardware sends out a read command to its sensor board (Safaric and Malaric, 2006). This read command is addressed to all connected sensors on the previously specified sensor board. In return sensors answer with their individually measured values. Depending on the latency of the network the order of incoming values vary from the order of read commands. In order to ensure the correct value order in the resulting Data Record each sensor needs to be associated with its respective Field ID, Enterprise ID and Field Length. The implemented bidirectional interface IPFIX Data Sampler supports this design. Each sensor is linked to exactly one IPFIX Data Sampler (Fig. 5). If a node enters the established wireless sensor

```
configuration ControllerAppC{}
implementation{
components ControllerC as App;
// Component Intitialize
components new IPFIXDataSampler16C(0x80A0,0xF0AA00AA) as Temp;
components new IPFIXDataSampler16C(0x80A2,0xF0AA00AA) as Light;
components new TempHumc() as TempSens, new TaosC() as LightSens;
// Connection of hardware sensors to IPFIX wrappers
Temp.Sensor -> TempSens;
Light.Sensor -> LightSens;
App.Sampler -> Temp;
App.Sampler -> Light;
}
module ControllerC { ...
uses interface IPFIXDataSampler as Sampler;
}
implementation {...}
```

Fig. 5: IPFIX data sampler source code

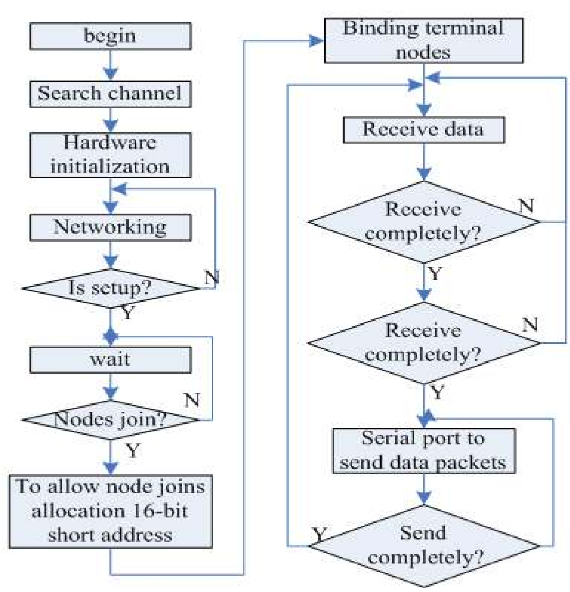

Fig. 6: The procedure flow of monitor

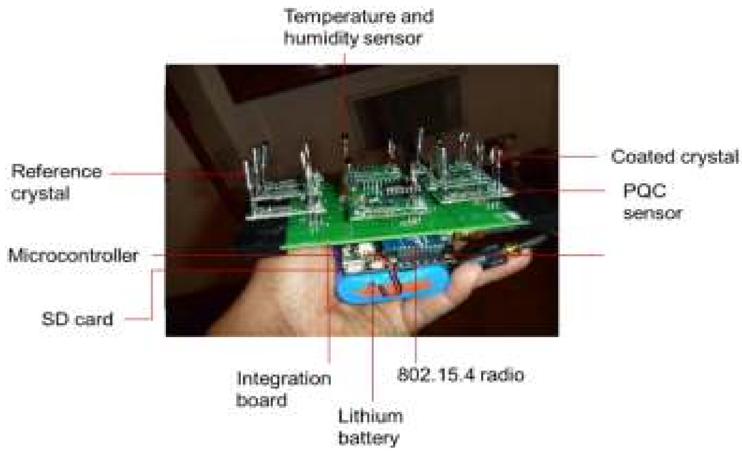

Fig. 7: Temperature sensor and relative humidity sensor

network, the rest task is the announcement of the Template Record used, followed by the Data Records after data acquisition.

Monitoring algorithms: Coordinator is the central node of the entire wireless network in ZigBee. Compared to networked macro sensors, an advantage of wireless sensor networks is the possibility to implement cooperative algorithms (Otto *et al*., 2006). A potential application for these algorithms is the reduction of network traffic by data preprocessing and aggregation. For a sensor application, it is not important whether a data aggregation is performed within the node itself or by a neighboring node. However, communication directed to data sinks has to be minimized since data sinks are usually located far away. It begins to work first. The main task of coordinator is to organize ZigBee network, allow the sensor nodes to join the network, bind the sensor nodes and send the collected data to the server. An example for a cooperative algorithm is location determination by triangulation (Cheong *et al*., 2006). This algorithm needs at least measurements from three different nodes. Computed positions then can be used for addressing or routing. The procedure flow of coordinator is shown in Fig. 6.

Implementation of food monitoring system:

Temperature sensor and relative humidity sensor: Temperature sensor and relative humidity sensor are used a commercial product called Sensirion SHT75 (Fig. 7). The characteristics of the temperature sensor and relative humidity sensor are shown in Table 1 and 2.

pH Sensor: The pH sensor is used MSFET 3330 Integrated Sensor. It has a structure similar to a MOSFET. Because operation of sensor responds to the H+ ions (solution exists) and an insulating layer (Ta2O5) surface, the electrochemical potential difference occurs. In this regard the potential difference

Table 1: The characteristics of the temperature sensor

Characteristics	Content
Supply voltage	1.5V to 5.5V
Supply current	9 µ A (type)
Temperature accuracy	20°C to 40°C±1.5°C
	-50°C to 70°C±1.8°C
	-50°C to 90°C±2.1°C
	-50°C to 150°C±2.7°C
Operating temperature	-50°C to 150°C

Table 2: The characteristics of the relative humidity sensor

Characteristics	Content
Supply voltage	1.5V to 5.5V
Sensing range	20 to 95% RH
Humidity accuracy	±9%RH
Response time	10s (average)

Table 3: The characteristics of pH sensor

Characteristics	Content
Sensitivity	55 mV/pH
Sensing range	pH 1 to pH 12
PH Accuracy	±0.05 pH
Response time	10s (average)

is a function of the ion concentration, ISFET channel conductance can change. Therefore, the ion concentration in the solution appears as a change in the current flowing in the drain. PH sensor includes a temperature sensor that is necessary for temperature compensation. Table 3 shows the characteristics of pH sensor.

Implementation of food monitoring system: Our system has a main PC master terminal which is used to monitor the status of all the slaves which covers the whole food area and slave structure which is under the PC master terminal supervision. The PC master will communicate to the slaves by Zigbee module. To avoid the communication fails, the system will place 2 slaves which will be placed in such way that they will be always in range of the PC master.

Our system makes the use of MEGA128 and 8051 micro controller which offers high performance and

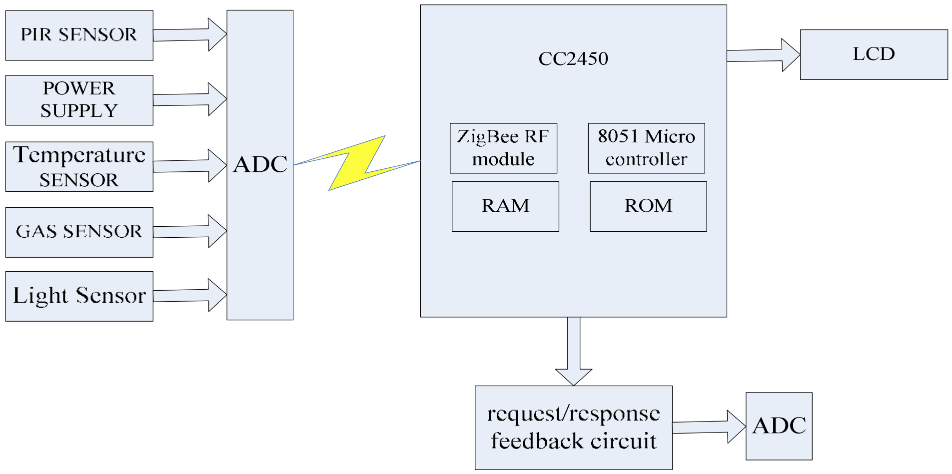

Fig. 8: Food monitoring system block diagram

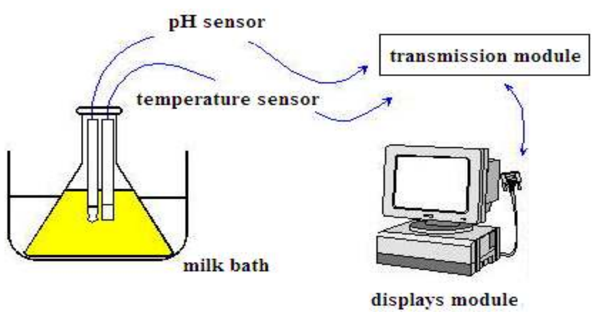

Fig. 9: Typical performance diagram of pH sensor

very low power consumption. This system has 16*2 LCD which indicates 16 columns and 2 rows to visualize the output of the application. So we can write 16 characters in each line and total 32 characters we can display on 16*2 LCD. LCD can also used in a project to check the output of different modules interfaced with the microcontroller. Thus LCD plays a vital role in the project to see the output and to debug the system module wise in case of system failure in order to rectify the problem.

Our system makes use of a master request protocol and slave response protocol (Fig. 8). In this system the Master sends the request to all the slaves. In the request frame the master mentions the slave ID. The request frame is received by all the slaves who are in range. The slaves who are in range receive the incoming frame and store it in its internal RAM memory. Then they check for the slave ID. If the incoming slave ID matches with their own slave ID then they accept the frame and send the parameter back to the master. If the ID does not match then the slave discards the frame. In this way we totally have 2 slaves.

The monitoring action of food is done through request/response feedback circuit when value exceeds above threshold value.

Table 4: Experiment result

Days	pH at 4°C	pH at 13°C	pH at 22°C
1	6.72	6.71	6.71
2	6.7	6.69	6.67
3	6.67	6.63	6.52
4	6.61	6.54	6.1
5	6.53	6.12	5.33
6	6.15	5.35	4.57

RESULTS AND DISCUSSION

In this study, the proposed system would experiment with using milk. The freshness of milk can be confirmed through pH. Experiment temperature is 4, 13 and 22°C. The experimental environment is shown in Fig. 9.

Host PC runs a monitoring program (developed in Microsoft Visual Studio) to display real-time sample data and store history data. The main display window is shown in Table 4 which is experiment result. pH of fresh milk is 6.5 to 6.9. And spoiled milk is fewer than 6.5. We obtained the model equations based on the experimental results.

Equation (1), (2) and (3) are the model equation of milk at 4, 13 and 22°C:

$$y = 6.7109e^{-0.000322t.} \quad (1)$$

$$y = 6.7109e^{-0.000399t.} \quad (2)$$

$$y = 6.7109e^{-0.000603t.} \quad (3)$$

CONCLUSION

In this study, we proposed the food monitoring system that managers and consumers can check the freshness of food. Proposed system consists of temperature sensor, humidity sensor, pH sensor, reader, server and dual frequency smart RFID tag with 1.56 MHz and 900 MHz. To validate the effectiveness of the proposed system, we were monitoring the freshness of milk by the environmental. The freshness of milk can be confirmed through pH at 4°C, 13°C and 22°C. PH of fresh milk is 6.5 to 6.9. And spoiled milk is fewer than 6.5. We obtained the model equations based on the experimental results.

REFERENCES

Ahmed, M.R., X. Huang and H. Cui, 2013. Mrakov chain Monte Carlo based internal attack evaluation for wireless sensor network. Int. J. Comput. Sci. Network Secur., 13: 23-31.

Bal, R.S. and A.K. Rath, 2013. Design of node clustering in wireless sensor networks. Int. J. Intell. Comput. Appl. Sci., 1: 3-10.

Cheong, E., E.A. Lee and Y. Zhao, 2006. Joint modeling and design of wireless networks and sensor node software. Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Technical Report No. UCB/EECS-2006-150, pp: 1-12.

Fu, G., J. Zhang, W. Chen, F. Peng, P. Yang and C. Chen, 2013. Precise localization of mobile robots via odometry and wireless sensor network. Int. J. Adv. Robot. Syst., 10: 1-13.

Gu, Q., H. Zuo, R. Su, J. Xu and H. Xu, 2013. A rank table based routing method for multi-sink zigbee wireless sensor network. J. Commun., 8: 535-539.

Gupta, S. and P. Saini, 2013. Modified pairwise key pre-distribution scheme with deployment knowledge in wireless sensor network. Int. J. Sci. Res. Network Secur. Commun., 1(2): 21-23.

Hashemi, S.E., H. Motameni, M.R. Ghaleh and S. Esmaeili, 2013. Clustering and routing wireless sensor network based on the parameters of distance, density, energy and traffic with the help of fuzzy logic. Int. J. Comput. Sci., 10: 9-15.

Liu, Z., P. Ni, X. Ma and W. Xiao, 2013. Development novel organization structure of wireless sensor network protocol based on ZigBee technology. Sensor. Transducers, 155: 107-114.

Otto, C., A. Milenkovi, C. Sanders and E. Jovanov, 2006. System architecture of a wireless body area sensor network for ubiquitous health monitoring. J. Mob. Multimedia, 1: 307-326.

Safaric, S. and K. Malaric, 2006. ZigBee wireless standard. Proceeding of 48th International Symposium ELMAR-2006 Focused on Multimedia Signal Processing and Communications, pp: 259-262.

Xu, Z., Z. Shi, Y. Zhang and Y. Zhang, 2013. Design of farmland information acquisition and transmission system based on ZigBee wireless sensor network. Proceeding of International Conference on Computer, Networks and Communication Engineering (ICCNCE, 2013), pp: 165-168.