Wireless Fire Monitoring System for Ancient Buildings

Hongli Xu[1] Liusheng Huang[2] Junmin Wu[3] Yang Wang[4] Ben Xu[5] Jichun Wang[6] Dashan Wang[7]
Depart. of Computer Science and Technology, Univ. of Science & Technology of China
Anhui Province-Most Co-Key Lab. of High Performance Computing and Its Application
Hefei 230027, P.R. China
Email:{hlxu3[1], angyan[4], ustrenyy[5],jichunw[6],dswang4[7]}@mail.ustc.edu.cn,
{lshuang[2], jmwu[3]}@ustc.edu.cn

Abstract
With the development of micro-processor and wireless communication, wireless sensor networks have becomes reality in some applications. In this paper, we mainly introduce the development of the fire monitoring system for ancient buildings based on wireless sensor networks. To satisfy the requirements of the application, lifetime and latency, etc, we design some key technologies in the different layers of the sensor nodes, such as data link layer, network layer and application layer. Finally, we realize the system based on MicaZ node and MTS300CB sensor broad, made by XBow Co..

Categories and Subject Descriptors
C.2.2 [**Computer-communication Networks**]: Network Protocols–Applications; C.3 [**Special-purpose and Application -based system**]: Real-time and embedded systems

General Terms
Design, Reliability

Keywords
Sensor Networks, Wireless, Fire Monitoring, Energy, Latency

1. INTRODUCTION
The recent advancements in micro-sensor technology and low-power analog/digital electronics have led to the development of wireless network of sensor devices [1-3]. Sensor network usually consists of hundreds or thousands of nodes that can be used to collect the useful information in an autonomous manner. Fire detection represents a class of monitoring applications with potential benefit for common security. The sensor nodes can provide the localized measurements and detailed information which is impossible through the traditional instruments. The capacity of the local processing and storage allows sensor nodes to perform the complex functions. Because of ability to communicate with each other, the data packets can be transmit across the network, and nodes can cooperate in performing a more complex task with each other, like data aggregation [4] and query[5]. Power efficiency makes applications flexible in resolving fundamental design trade-offs, e.g., between sampling rate and battery consumption. Low-power radio with efficient protocol stacks allows general communications among network nodes. The computing capability allows the sensor networks to be reprogrammed before deployed in the field. The nodes are able to adapt their operations dynamically in response to the changes of the environment.

INFOSCALE 2007, June 6-8, Suzhou, China

DOI 10.4108/infoscale.2007.890

We are cooperating with the member of the Fire Community to make the potential of this emerging technology a reality. The new applied context is helpful to differentiate problems with simple, concrete solutions architecture for this kind of application. Moreover, we shall design some key technologies in the system development. Collaboration with the scientists in other fields helps to define the boarder application spaces, so we will propose a specific requirement of application, and offer objective evaluations of the technologies. The impact of sensor networks for monitoring will be measured by their ability to enable new applications.

This paper mainly introduces the development of a fire monitoring prototype system that is greatly representative of this domain. It presents a collection of requirements, constraints and guidelines that serve as a basis for the general sensor network architecture for such an application. It also describes some key energy-conserved technologies for different layers of the sensor node, such as data link layer, network layer, applications layer and the server software.

The remainder of this paper is organized as following. Section 2 defines the requirements of the fire monitoring system and introduces the application environment. Section 3 designs a master-slave network architecture serving the functions that specified by the application. Section 4 introduces the hardware platform for this system. Section 5 discusses the implementation of the software module of this system, deployed in Chenzhi tang. Section 6 concludes the paper.

2. Background Preliminary
Researchers in the fire science become increasingly concerned about the new technology in some applications where the traditional technologies are not as suitable as before, such as ancient buildings, forest, etc. That's because the wired equipments will destroy the original structure of the ancient buildings or can not be deployed in the forests. The other reason is that the wire will result in fire in many situations. The focus of this paper is to develop a fire monitoring system for ancient buildings. In the followings, we will introduce an application environment and the application requirements.

2.1 Application Environment

The fire monitoring system is developed to be used in Chenzhi Tang, Huangshan, Anhui, China. Chenzhi Tang is an important protection site of culture relics at the level of the province. It is located in the middle of the region, initially built in the fifth year of the XianFeng of Qing dynasty (about A.D. 1835), which is the tenement of a big salt merchant, Wang Dinggui. It has large and splendid architecture, reasonable distribution in reason and framework perfection and also establishment is complete and facture is polished. There are seven floors, nine courtyards, about sixty large or small rooms and one hundred and thirty-two stakes of the overall house. It is mainly composed of adobe and timber, decorated with the exquisite carving made from the stone, adobe and timber. It is a classical work of the dwelling of the south of Anhui province, China.

2.2 System Requirements

Through the discussion with the experts of the fire science, we conclude some functional requirements and performance requirements. First, the main functional requirements are as followings:

- All the sensor nodes and base station can self-organize the network.
- The sensor node can join the network freely.
- The server can determine whether the fire happens, and send an alarm message to the manager immediately.

Also there are two critical performance requirements for fire monitoring system. One is the sensor network longevity, i.e., the time to recharge the battery of the node. It is required that the network longevity is at least one year, and the period of two years is optimal. The other is that the time of alarm is no more than 30 seconds. These requirements greatly affect the system design and implementation.

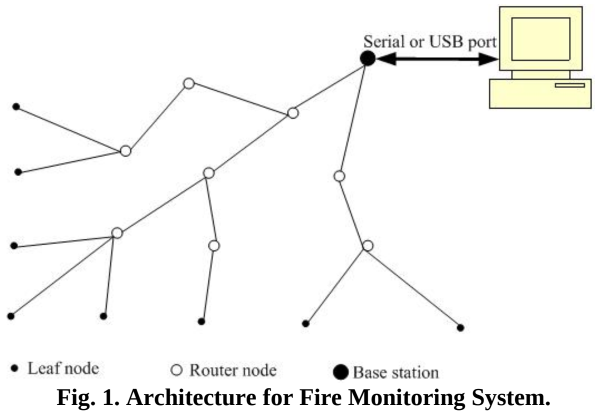

Fig. 1. Architecture for Fire Monitoring System.

3. System Architecture

This section describes the system architecture which is fit for this king of application. In this system, we adopt the master-slave architecture, as shown in Fig. 1.

The slave level consists of the sensor nodes that perform the general function of sensing and monitoring the environment. These nodes, called leaf nodes, can only communicate with router nodes or base station. The master level consists of the sensor nodes that perform the function of relaying the sensed data from the sensing nodes to the base station. We call this node as a router node. In the system, the router nodes may be deployed in the aisle to connect the leaf nodes and the base station. And the base station is responsible for transmitting the sensed data from the sensor network to the server by the serial port or USB port.

In this architecture, the leaf nodes don't need to maintain the complex routing information, but sends the sensed data to one of the router nodes. Thus, the leaf nodes can save the energy by turning the radio off in the most time.

4. Hardware Platform

4.1 Sensor Node

In our system, the network is built on Mica platform. The sensor nodes being used are MicaZ (MPR2400) [6], which contains IEEE 802.15.4/ZigBee compliant RF transceiver. MicaZ node uses 16 channels, 2.4GHz radio to provide bidirectional communication with the rate 256Kbps. A pair of conventional AA batteries and a DC boost converter provide a stable voltage source, though other renewable energy source can be easily used. The micaZ node contains 51-pin Expansion Connector to support interfaces from analog input to UART.

4.2 Sensor Board

In order to support additional sensor and data acquisition, the XBow cooperation provides a variety of sensor boards. In this system, we use the broad MTS310CA, which contains the following sensors: temperature, photo, microphone and sounder, etc. All of these boards connect to the MicaZ node via the 51-pin Expansion Connector. In the prototype system, we use the temperature as the main data source. In the future, we will add the smog sensor to the board.

5. Implementation Strategies

To develop the system for fire detection in the ancient buildings, we design some energy-conserved technologies in the different layers, such as data link layer, network layer, application layer and the server layer.

5.1 Data Link Layer Implementation

As we know, idle listening is the main energy waste in the wireless sensor network. We adopt the period of sleep and activation scheme as that in S-MAC [7] to reduce the energy consumption. For example, if a node sleeps for 900ms, then listens for 100ms, its duty cycle is reduced to about 10%. So it can achieve 90% energy saving at most.

The basic scheme is that each node goes to sleep for some time, and then wakes up and listens to see if any neighbor node wants to communicate with it. But there are some differences between the leaf nodes and router nodes. Each period T_p consists of a sleep timeslice, T_s, and an activation timeslice, T_a. That is, $T_p=T_s+T_a$. The leaf nodes only send the sensed data to one of the routers, while the routers will relay the data packet and maintain the multi-hop routing. Hence, the period for the leaf node is larger than that for router nodes. For example, the period for the leaf node is 20s, while 2s is for router node. In each period, the activation time is enough to receive a message.

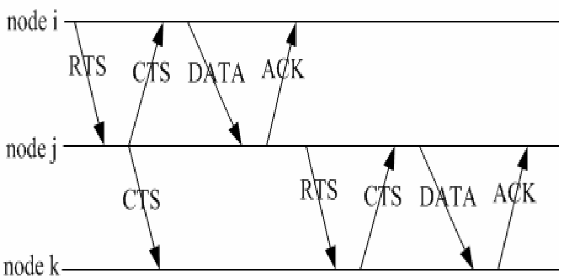

Fig. 2. Energy-Delay-Efficient MAC Mechanism.

In order to decrease the transmission's delay, we design a novel MAC mechanism. As shown in Fig. 2, when node *i* wants to send a packet with type *s* to node *j*, it first sends a RTS message to node *j*. On receiving the RTS, node *j* will send CTS to node *i*. One field of the CTS packet is NHDest, which is the identity of the possible next-hop node to base station. The NHDest is determined by the last successful transmission of type *s* from node *j*. After receiving CTS, the node checks whether its identity is the same as NHDest. In Fig. 2, NHDest is *k*. Then the node *k* will keep additional active time to support further reception from node *j*. Otherwise, the field NHDest is assigned with NULL.

Our scheme requires periodical synchronization with the neighboring nodes. Unlike the previous synchronization protocols, such as RBS [8], ATS [9], etc, we design a new technology, called on-demand synchronization. In this method, each node only synchronizes to the communicating neighbors. Instead of special packet, we add the timestamp information into the CTS packet. After every synchronization period T_{sync}, the adjacent neighbors are synchronized.

As each router node may receive more than one packet at a certain time, it needs buffer to store the packet. The memory of the sensor node is very limited, and the buffer can't be too large. In our system, the buffer can contain two units. One if for the packet being sent, the other is for the packet to be transmitted. The buffer obeys the FIFO order. If there is a new packet arrived and the buffer is full, we should use it to make place of the old one.

5.2 Network Layer Implementation

In this special application, the main issues of WSN include the network lifetime, the delay and the stability, and or so. The lifetime of the network is highly affected by the routing protocol. Appropriate routing methods can balance the energy level among all the sensors and can lengthen the lifetime of the network. According to the architecture of this system, the network topology is a tree described as follows: the base station being the root, the router nodes composing the backbone of the tree, while the leaf node being the leaf nodes and sensing the environment. The routing algorithm running on this topology should be energy-balanced to lengthen the lifetime and be stable to alarm the fire in time.

When a node wants to join the network, the node will keep active for a certain time. It broadcasts a request-join packet in the vicinity, and the neighboring router that receives this packet will reply it only if the router has joined the network. After a given time, the node determines its parent node, and informs the data link layer to enter the normal sleep-active cycle. After obtaining the neighbors' information, the node will select one from the available routers as its parent. The first criterion is the minimal hop count from the base station. If these routers are not unique, the parent selection metric is related to the energy to extend the network's lifetime. We adopt the energy allocation index (EAI) which is a ratio: the remainder energy divided by the number of its possible children. We randomly choose one whose energy allocation index is greater than the threshold, C_{eai}.

In the phase of routing maintenance, we design the different mechanism for the leaf node and router node. For the leaf node, if it detects that the current parent is disabled, it will request to re-join the network the same as before. While the maintenance mechanism for the router node is a little complex. Each router maintains a neighboring router table which records the dynamic information for each neighbor, such as node identity, hop count, remainder power, possible children and liveness, etc. when receiving a packet, it is necessary to update the neighbor table. Especially, the liveness field is related with the number of the correct transmission. To balance the energy consumption, each router will re-select the parent in a large period, T_{sp}. The selection criterion is similar to that described in the last paragraph. To keep the router stability, if the energy allocation index of the current parent is greater than the threshold, C_{eai}, the router won't change the parent node. If the router detects that the parent is disabled, it will select another candidate router as the parent. If there are no candidate parent nodes, the router will request the neighbors' information to discover the living neighbors. At the same time, each router will broadcast its existence in each period, T_{be}. Usually, $T_{be} > T_{sp}$. By this way, we establish and maintain an energy-efficient data gathering routing tree.

5.3 Application Layer Implementation

In this system, base station collects the data which contains environment information from the sensing node through the data gathering tree, then these data are sent to the server by serial port. Sensor nodes usually have limited power, therefore it is of great importance to make full use of the energy.

```
If (sampling timer fired)
  Then get data from ADC
  If (data is abnormal or sending timer fired)
     Then send data
     Else notify MAC sub-layer to turn RF off
   End If
End if
```

Fig. 3 The strategy for the application layer on leaf node

In the leaf node, one timer is arranged to sense the environment at interval of T_{sensor}, which is the sampling period. When this timer is fired, the leaf node will request to initiate an ADC conversion on a given port to generate the source data. T_{sensor} is supposed to be a right value about 20 seconds, for if it is too large, we may not capture the fire information in time; if too small, it is a waste of the energy. To conserve the energy deeply, not all the sensed data should be sent to the base station over multi-hop routing, so sending data at the interval of T_{send}, which is defined as $T_{send} = n \times T_{sensor}, n \in Z^{+}$. The strategy of sending data is described in Fig. 3.

5.4 Server Implementation

The server implementation is programmed by C#, using Microsoft Visual C# 2005 in environment of Microsoft XP sp2 and taking Microsoft SQL server 2000 sp4 as DBMS.

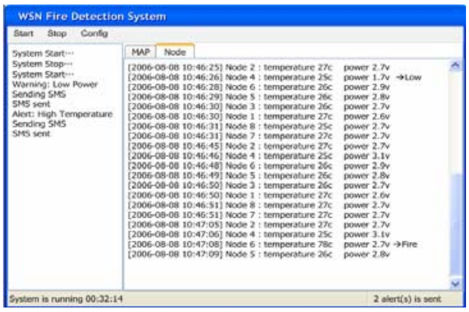

Fig. 4 Graphic Interface for Sampling Data

The server collects packets from the serial port which connects the base node and saves those data into database. And foremost it is responsible for sending the alert message to users when something abnormal (take catching fire as an example) happens. In the prototype system, we use a simple rule to detect the fire. For example, if the detected temperature is higher than a threshold, then we regard that the fire happens.

There are four threads in the server software. Thread Portread reads the packets from the port and puts them in a ingoing buffer. Thread Info-Convert takes the packets from ingoing buffer and converts them into nature data that could be easily understood. Thread MainApp shows the data collection, as shown in Fig. 4, and decides whether to send an alert, if necessary, it not only indicates the abnormal data but also the region with the problem (the read area illustrated in Fig. 5). The alert message will be put into an outgoing buffer and be delivered immediately. Thread DBop inserts the collected data into database.

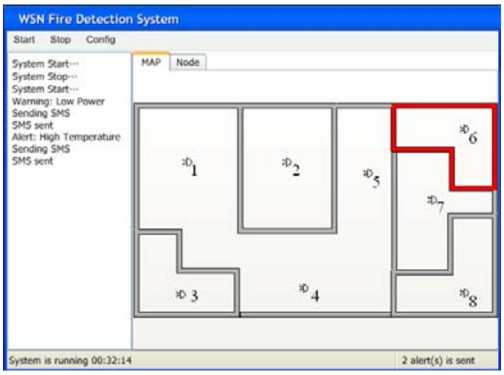

Fig. 5 Graphical Interface for Fire Monitoring

6. Conclusions

In this paper, we introduce the development of the fire monitoring system for ancient building, from the architecture to the key technologies of the different layers. Because this project is cooperated with the Fire community at the University of Science and Technology of China, the application requirements are obviously defined, which benefits our development. Now, this system including 15 leaf nodes, 12 routers and base station is in the phase of debugging in the real environment. The testing results show its efficiency of our system. Especially, the current network longevity is about two months.

In the future, we shall improve the performance of this system. The most of all, how to conserve the energy of the router is a challenging problem for us. Moreover, our team is planning to extend this system to about 150 nodes supported by National 973 program.

7. ACKNOWLEDGMENTS

We thank to Prof. Yao Bin and other researchers for giving us some constructive advices in the project development. And this paper is supported by the National Development and Reform Commission of China under Grant No. CNGI-04-10-2A, and the National Grand Fundamental Research 973 Program of China under Grant No. 2006CB303006.

References:

[1]J. M. Kahn, R. H. Katz, and K. S. J. Pister, Mobile Network-ing for Smart Dust, In proceed- ing of 5th ACM/IEEE Mobile Conference, 1999

[2]T. He, S. Krishnamurthy, J.A. Stankovic, T. Abdelzaher, L. Luo, etc, “An energy-efficient surveillance system using wireless sensor net- works”, Mobisys’04, 2004

[3]A. Naubwarubgm J. Polastre, R. Szewczyk, D. Culler and J. Anderson, “Wireless Sensor Networks for Habitat Monitoring”, WSNA’02, Sep. 28, 2002, Altanta, Georgia, USA

[4]C. Intanagonwiwat, D. Estrin, R. Govindan and J. Heidemann, “Impact of Network Density on Data Aggregation in Wireless Sensor Net- works”, ICDCS’02, 2002

[5]J. Gehrke, and S. Madden, “Query Processing in Sensor Networks”, IEEE Pervasive Computing, 1536-1268/04, 2004, pp. 46-55

[6]XBow Coop., “MPR/MIB User’s Manual”, www.xbow.com, Sep. 2005

[7]Wei Ye, John Heidemann and Deborah Estrin, Mediam Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks, ACM/IEEE Transactions on Networking, 12(3): 493-506, Jun 2004

[8]J. Elson, L. Girod and D. Estrin, “Fine-Grained Network Time Synchronization using Reference Broadcasts”, OSDI 2002, Vol 36, pp. 147-63, 2002

[9]H.L. Xu, L.S. Huang, Y.Y. Wan and B. Xu, “Accurate Time Synchronization for Wireless Sensor Networks”, MSN’05, LNCS 3704, pp. 153-63, 2005