A ZigBee* -based Sensor Node for Tracking People's Locations

Satoshi Takahashi[1,2], Jeffrey Wong[4], Masakazu Miyamae[1], Tsutomu Terada[1,3], Haruo Noma[1], Tomoji Toriyama[1], Kiyoshi Kogure[1], and Shojiro Nishio[2]

[1] ATR Knowledge Science Laboratories, 2–2–2 Hikaridai, Keihanna Science City, Kyoto, Japan
{jwong,miyamae,noma,toriyama,kogure}@atr.jp

[2] Osaka University, 1–5 Yamadaoka, Suita, Osaka, Japan
{takahashi.satoshi,nishio}@ist.osaka-u.ac.jp

[3] Kobe University, 1–1 Rokkoucho, Nadaku, Kobe, Hyogo, Japan
tsutomu@eedept.kobe-u.ac.jp

[4] Simon Fraser University, Engineering Annex, 6298 Biological Science Road, Vancouver, BC, Canada

ABSTRACT
A sensor network system has been developed for tracking people's locations in workplaces as part of a ubiquitous network system for providing context-aware services in daily activities. Since the installation of such a sensor is desired any place within its target domain with few limitations, it must operate by battery for a relatively long time, e.g., one month. To satisfy this requirement, we designed a battery-operated sensor node based on ZigBee technology and extended its operation period by developing a flexible sleep control protocol and a high-accuracy time synchronization mechanism between sensor nodes to reduce power consumption. From simulations based on actual data collected, we confirmed that a sensor node located in a hospital's medical ward can work over 21 days using four AA Ni-H batteries.

Categories and Subject Descriptors
C.2.4 [COMPUTER-COMMUNICATION NETWORKS]: Distributed Systems

General Terms
Experimentatio,Design

1. INTRODUCTION
In recent years, the evolution of radio communication technologies and the miniaturization of electrical devices have allowed the construction of wireless sensor networks. Such wireless sensor networks consist of several sensor nodes with radio communication, each of which can transfer sensor data to other nodes. Sensor network technology can provide various context-aware services in such fields as medicine, health, crime prevention, and agriculture.

We developed a sensor network system to provide context-aware services to support their daily activities. In such a system, location is not only an element of context but also a clue that can be used to restrict the type of activities taking place in it. Therefore, obtaining people's location information is important.

We previously developed a sensor network system to track people to specify the room in which a person is located [1]. This system consists of infrared ID (IrID) transmitters attached to a person and IrID receivers installed in a doorway. An IrID receiver can detect IrID signals transmitted by an IrID transmitter and send the data to a server by a wireless LAN (IEEE-802.11b).

However, such a wireless-LAN-based sensor node consumes so much power that a wall outlet is required. This restricts where the sensor nodes can be installed and complicates attaching them to such movable objects as a bed or a medical device.

To solve such problems, we designed a ZigBee-based sensor network system that includes battery-operated nodes and developed a flexible sleep control protocol to reduce electrical power consumption.

In such a wireless sensor network, synchronizing the sensor node clocks with high time accuracy is important. For example, low time accuracy may create misunderstanding of a person's tracking pass route. We therefore provide a new sensor node with a high-accuracy time synchronization mechanism between sensor nodes.

In the following, Section 2 discusses the related work, and

*ZigBee is registered trademark of Koninklijke Philips Electronics .N.V.

CASEMANS '08, May 19, 2008 Sydney, Australia

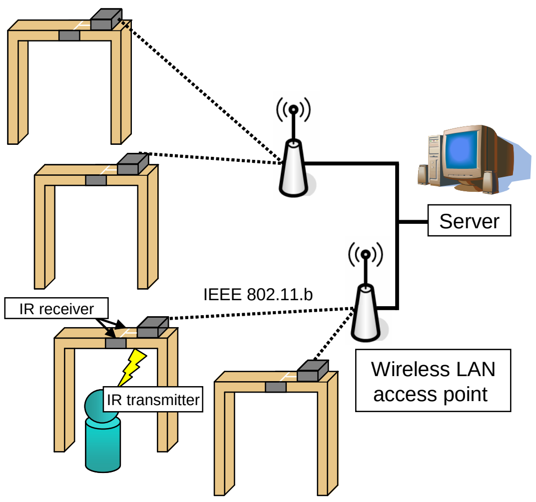

Figure 1: Sensor network system for tracking people's location based on WiF [1]

Section 3 describes our sensor node design. Section 4 presents our prototype implementation and its evaluation, and Section 5 presents our location system. Finally, Section 6 presents our conclusion and future work.

2. RELATED WORK

Several research efforts have constructed a common framework and a device for ubiquitous computing environments.

Smart-Its [2], one of the most famous sensor network nodes, was developed by the Disappearing Computer program [3] in Lancaster University, ETH Zurich, University of Karlsruhe, Interactive Institute, and VTT. It is a small computing device that consists of two independent boards: a core board that mainly consists of processing and communication hardware, and a sensor board that contains a separate processing unit and various sensors and actuators.

MICA MOTE [4], another famous sensor network node developed by the NEST project at UC Berkeley, is now produced and marketed by Crossbow Technology. It is driven by the TinyOS operating system, which was especially designed for sensor network nodes to allow operation with a small amount of memory and low CPU power, thus keeping power consumption low.

Other sensor nodes such as SensiNet [5], Ubisense [6], and U^3[7] have also been researched, developed, and commercialized for practical use.

Much sensor node research has considered low-power consumption. Hempstead developed a low-power sensor device for sensor network applications [8]. This study designs an optimum device by considering the characteristics of sensor network applications. In simulations, this device accomplishes one to two orders of magnitude reduction in low power. Lymberopoulo also developed XYZ, a low-power sensor node [9], whose wireless standard is based on IEEE 802.15.4. This device has RTC and reduces power consumption by putting the sensor node "asleep" for a long term.

As explained above, many products and research can be found for low-power sensor nodes and platform driving for long terms. In addition to considering low-power consumption, our goal is the realization of a high-accuracy time synchronization mechanism between sensor nodes.

3. SENSOR NODE DESIGN

We developed a location sensor system founded on an IEEE 802.11b-based wireless connection. Since this system's device consumes so much power that it requires a wall outlet, installing the sensor node any place is difficult. We developed a new low-power battery-operated sensor network based on ZigBee to solve the problem. The new sensor node's design is based on the following points to realize low-power communication.

- flexible sleep control protocol
- high-accuracy time synchronization

We explain them in detail.

3.1 ZigBee Network

ZigBee is a wireless communication standard based on IEEE 802.15.4 that contains an ad hoc wireless sensor network. The physical and MAC layers are standardized by IEEE 802.15.4, and the other upper layers are standardized by ZigBee Alliance. Although ZigBee specifies a lower data rate than other wireless communication standards such as IEEE 802.11b and IEEE 802.15.1, it can flexibly control electrical power consumption. ZigBee contains a three role: a coordinator, routers, and an end device.

- ZigBee Coordinator (ZC)
  - The network contains one coordinator device that controls the network.
- ZigBee Router (ZR)
  - This device has a routing data function.
- ZigBee End device (ZE)
  - This device does not have a routing data function.

Using these devices, a network is dynamically possible. Multihop message transmission can also send messages between ZE and ZC by ZR.

Another feature of the ZigBee End device is that it decreases power consumption by sleeping. However when ZE sleeps, it cannot receive messages from ZC. Therefore, when ZEs sleep in a network, ZR or ZC stacks the messages for the ZE devices, and the ZE gets them by polling ZR or ZC. However, when ZR's message buffer overflows or the message holding time is over, the message is discarded. At this time, since ZR does not inform ZC or ZE about the discarded messages, ZE sometimes cannot receive its messages. Therefore, users add sleep and message management controls to receive messages. ZigBee does not consider communication delays either; therefore a communicating message delay is generated when sending and receiving messages between ZE and ZC by a route, a radio wave condition, a ZE sleep condition, and so on.

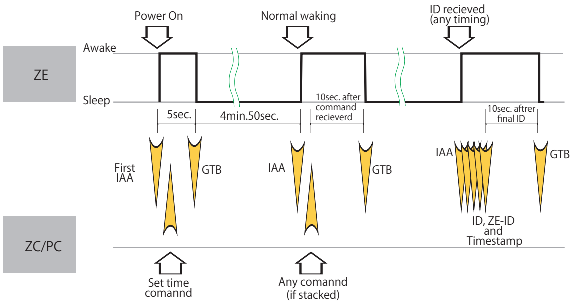

Figure 2: Sleep control protocol

In these ways, although ZigBee provides the routing function, users must implement sleep and delay control functions. Next we describe the following methods for the sleep and time synchronization controls in consideration of communication delay.

3.2 Sleep Control Protocol

To be synchronized in time among sensor nodes in a sensor network, ZC must send a time-synchronized message to the ZEs. However, when sleeping, ZE cannot receive ZC's messages, so it won't learn about messages discarded en route. To solve this problem, we propose an application that manages each sleeping state of a ZE on a server PC. The detailed protocol is shown in Figure 2. When the server application sends a message to ZE by ZC, it searches a ZE state from a list of ZE state information. If the state is "awake," the application sends a message. If the state is "sleep," the application stacks the message and sends a message to change the state to "awake." In these ways, messages are sent to ZE while it sleeps so that it can receive them. To realize these methods, we implemented state notice messages sent by ZE to a server that manages each ZE state. The following are the state notice messages.

- Go To Bed (GTB) Message
  - ZE sends this sleep message while sleeping.
- I Am Awake (IAA) Message
  - ZE sends this wake message after waking up.

The server application manages each ZE state by getting these messages. For example, when ZE wakes up, it sends an IAA message to ZC. The application receives it by ZC, searches for ZE's messages, and sends them. After ZE communicates that it can receive a message at a definite period of time, it sends a GTB message to ZC and falls asleep. By this regular action, ZE gets its messages.

When ZE detects sensor inputs, it immediately wakes up and sends an IAA message to the server and continues to send sensor data. When sensor inputs are finished, ZE remains in the wake up mode to receive messages. After a certain period of time, ZE sends a GTB message to the server and falls asleep.

In these ways, the sleep control protocol sends and receives messages in the ZigBee network that has the sleeping ZEs.

Table 1: Waking time of sensor node in each event

Power On	for 5 seconds
Normal	for 10 seconds
Receive sensor data	for 10 seconds after receiving
Receive message from PC	for 10 seconds after receiving

In addition, an immediate sleep cancel mechanism for sensor inputs covers the delay of sending sensor data from ZE's sleeping state.

3.3 Assurance of Accuracy Time Synchronization

In the location sensor system, each ZE should immediately send sensor data to the server to recognize the location of people in real time. However, a method where the server application adds a time stamp when receiving sensor data engenders pass time errors since ZigBee does not consider communication delay. To solve this problem, we adapt a method where ZE adds a time stamp to the sensor data so that the pass time becomes more precise. This method solves the communication delay; however, the accuracy time synchronization of ZE must be enhanced because the internal time error among sensor nodes restored different routes in the prototype of the location sensor system.

The analysis of accuracy time synchronization to solve this problem revealed that the necessary accuracy time synchronization is about 300[ms] a day among sensor nodes [11]. Our proposed ZE achieves high-accuracy time synchronization by enhancing the accuracy of its internal clock and synchronizing the internal time with all other ZE's in the sensor network. As in synchronization with each ZE, the server application sends a time setting command message to each ZE and calculates the time error from the difference between the time when ZE sends a command response and when the application receives it. These steps are repeated until the time error is within 300[ms].

4. IMPLEMENTATION AND EVALUATION OF PROTOTYPE

We implemented a prototype of the sensor node based on our design described in the previous section.

4.1 Prototype Implementation

We developed a prototype of the sensor node. A circuit diagram and a device prototype of the sensor node are shown in Figure 3. This device uses R8C/25 as a sensor CPU and a Renesas ZigBee module YCSCZB2A2NN as a wireless communication module. We designed a prototype of a sensor node with high-accuracy time synchronization using a Temperature Compensated Xtal Oscillator (TCXO). This mechanism not only maintains high-accuracy time synchronization but can also make a time counter circuit independent so that the sensor CPU can put its core module to sleep. Here, electric power consumption can be decreased by putting the sensor CPU to sleep and maintaining high-accuracy time synchronization. The Temperature Compensated Xtal Oscillator uses EPSON TOYOCOM TG-3530SA-3.2768 KHz.

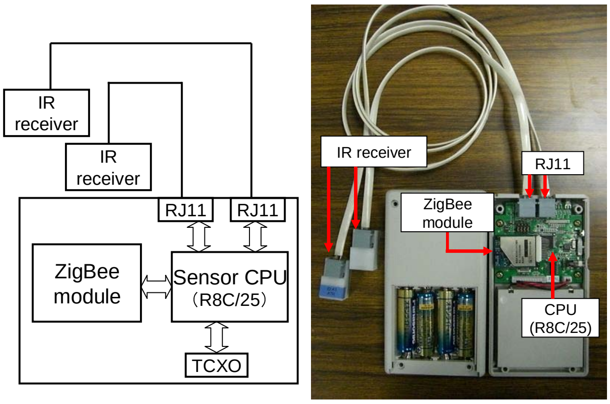

Figure 3: Block diagram (left) and device (right) of sensor node prototype

Table 2: Daily mean of received IDs at three locations measured in a hospital [1]

Sensor	Location	Average number of IDs
A	hospital room 1	36,000
B	hospital room 2	23,000
C	nurse station	140,000

Table 3: Operating time of each prototype of sensor node

Sensor	Operating time
A	24 days and 22 hours
B	25 days and 07 hours
C	09 days and 22 hours

4.2 Estimation of Electrical Power Consumption

To estimate the sensor node's prototype, we measured the consumed current at its battery port and calculated the electrical power consumption. The results show that the electrical power consumption is $4.4[mW]$ while sleeping and about $234.3[mW]$ when awake. With these results, we simulated the operating days of the sensor node using pass data logs gathered in an actual hospital during previous experiments. As shown in table2, we used three kinds of pass data logs that have different pass frequencies. The IR Transmitter sends a personal ID up to 40 Hz. The subject passed through the hospital's rooms roughly 150 times a day and visited the nurse station roughly 1000 times a day. We simulated the operating time of the sensor node using the pass data logs (table 2), as in the pre-prototype. The prototype is equipped with four AA Ni-H batteries ($5[V] \cdot 2000[mAh]$). The sensor node sleeps for 4 minutes 50 seconds and wakes up as in Table 1.

Table 3 shows that the prototype of the sensor node may usually work about 25 days in hospital rooms. In the nurse station, reducing this duration to about 10 days since the sensor node frequently wakes up. This result shows that the sensor node requires both a large battery or an AC adaptor in places where many people continually come and go.

Table 4: Time differences of each sensor node

Time	Sensor A	Sensor B	Sensor C	Sensor D
24 hours	$+22[ms]$	$-33[ms]$	$+20[ms]$	$-49[ms]$
48 hours	$+27[ms]$	$-132[ms]$	$\pm 0[ms]$	$-109[ms]$

Table 5: Operating time of each sensor node in operating experimentation

Sensor	avg number of IDs	operating time
A	3200	21 days and 10 hours
B	1500	21 days and 11 hours

4.3 Estimation of Accuracy Time Synchronization

To estimate the accuracy time synchronization of the prototype, the server PC sent a query to four prototypes of the sensor node to check its internal clock, and we calculated the time differences between the query's responses and the PC time.

Table 4 shows that the largest time difference is $49[ms]$ at 24 hours, and the largest time difference is $132[ms]$ at 48 hours. If the communication delay is about $100[ms]$, the accuracy time synchronization accomplishes its goal, which is accuracy time synchronization of $300[ms]$. Adequate accuracy time synchronization can be accomplished by daily synchronization.

In these results, the performance of the sensor node's prototype was satisfactory for operating time with a battery and accuracy time synchronization of a place where about 150 passes accrue daily.

4.4 Operating Experimentation

To evaluate the actual operating time, we located two ZE sensor nodes at the entrance of a research room at Osaka University. The sensors node tracked four people with an IR transmitter, and we monitored their operating times. They operated with four AA Ni-H batteries ($4.8[V], 2000[mAh]$).

Sensor node A lasted 21 days and 10 hours, and sensor node B lasted 21 days and 11 hours. During this operating time, sensor nodes A and B detected a daily average of about 3000 and 1500 IR data, respectively. These data are fewer than the pass data we gathered in a hospital room, even though the total operating time is almost the same. The increased electrical power consumption of excessive unscheduled wake-ups was caused by a disturbance, which may have been caused by the infrared light from the ceiling fluorescent lamps. However, it was prevented by optical film.

5. LOCATION SENSOR SYSTEM

The location sensor system using the prototype of the sensor node is shown in Figure 4. Sensor node (ZC) is connected by a serial cable to the server PC to which it sends sensor data. The messages of the sensor nodes (ZE) are sent to the

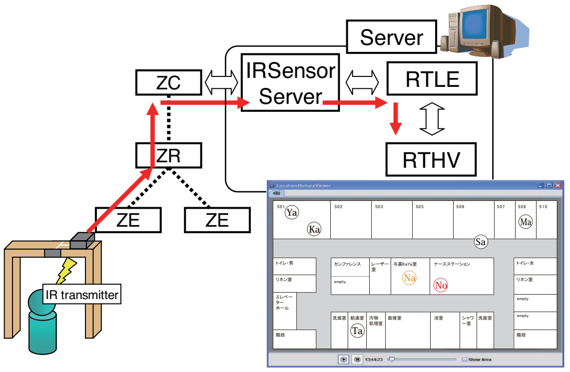

Figure 4: Sample of location sensor system using sensor node based on ZigBee

server PC through ZR and ZC. On the server PC, IRSensorServer, a sensor data management system, handles the messages sent by each sensor node. IRSensorServer sends these raw IrID data to the Real Time Location Estimator (RTLE), who extracts the gate IDs, the personal IDs, and the pass time from the raw IrID data and sends the results to the Real Time Hospital Viewer (RTHV), a person viewer that displays person icons on a map. Using the location sensor system with ZigBee sensor node, we easily recognize individuals.

6. CONCLUSION

In this paper, we described the design and implementation of a new low-power sensor node for location sensor networks. The sensor node uses ZigBee for wireless communication, and our flexible sleep control protocol realizes low electrical power consumption and high-accuracy time synchronization. In particular, ZigBee affords routing functions; however, it doesn't afford sleep and message delay controls. We implemented the prototype of the sensor node and confirmed that the system usually lasts about 25 days using four AA Ni-H batteries in hospital rooms during simulations using the pass data logs gathered at past estimation in an actual hospital. This sensor node offers satisfying operating time performance with battery and accuracy time synchronization. In the actual operating experimentation of the sensor node in our laboratory in Osaka University, we also confirmed that the sensor node can last about 20 days.

7. ACKNOWLEDGMENTS

This research was supported in part by the National Institute of Information and Communications Technology.

8. REFERENCES

[1] Fusako Takayanagi, Ren Ohmura, Futoshi Naya, Haruo Noma, and Kiyoshi Kogure: Development of IR-pass Sensor for Human Behavior Analysis: The 4th Forum on Information Technology (FIT2005), pp. 259–260 (2005) (in Japanese)

[2] Michael Beigl and H. Gellersen: Smart-Its: An Embedded Platform for Smart Objects, In Proc. of the Smart Objects Conference (sOc) (2003)

[3] The Disappearing Computer: http://www.disappearing-computer.net/

[4] Crossbow MIAC MOTE: http://www.xbow.jp/.

[5] Sensicast SensiNet: http://www.sensicast.com/.

[6] Ubisense: http://www.ubisense.net/.

[7] Kawahara, Y., Minami, M., Morikawa, H., and Aoyama, T.: Design and Implementation of a Sensor Network Node for Ubiquitous Computing Environment, In Proc. of VTC2003-Fall (2003)

[8] Mark Hempstead, Nikhil Tripathi, Patrick Mauro, Gu-Yeon Wei, and David Brooks: An Ultra Low Power System Architecture for Sensor Network Applications, In Proc. of the 32nd annual international symposium on Computer Architecture (ISCA '05), Vol. 33, Issue. 2, pp. 208–219 (2005)

[9] Dimitrios Lymberopoulos and Andreas Savvides: Networked sensor platforms: XYZ: a motion-enabled, power aware sensor node platform for distributed sensor network applications, In Proc. of the 4^{th} International Symposium on Information Processing in Sensor Networks (IPSN '05) No. 63 (2005)

[10] ZigBee Alliance: http://www.zigbee.org/en/index.asp.

[11] Tatsuya Sakamoto, Ren Ohmura, Futoshi Naya, Haruo Noma, Tomoji Toriyama, Kiyoshi Kogure, and Mutsuo Sano: Evaluation of Time Synchronization Accuracy in Sensor Networks for Tracking Locations of People, The Special Interest Group Notes of IPSJ (2007-UBI-015), Vol. 2007, No. 74, pp. 35–40 (2007).