Wireless Sensor Network Implementation for Mobile Patient

Ashraf S Mahmoud, Tarek R Sheltami and Marwan H Abu-Amara
Department of Computer Engineering
King Fahd University of Petroleum and Minerals
Dhaharn, Saudi Arabia, 31261
{ashraf, tarek, marwan}@ccse.kfupm.edu.sa

Abstract

Recent advances in embedded computing systems have led to the emergence of wireless sensor networks (SNETs), consisting of small, battery-powered "motes" with limited computation and radio communication capabilities. SNETs permit data gathering and computation to be deeply embedded in the physical environment. Large-scale ad hoc sensor networks (ASNET) can provide dynamic data query architecture to allow the medical specialists to monitor patients remotely via (PDAs) or cellular phones. A three-layered architecture is proposed where sensors, microcontrollers, and central server/handhelds occupy the lower, middle, and top layers, respectively. The implemented network distinguishes between periodic sensor readings and critical where higher priority is given for the latter. In this paper we implement 3 special cases for tracking and monitoring patients and doctors using SNETs. Finally, the performance of a large scale of our implementation has been tested by means of simulations.

Index Terms **- Sensor nodes, sensor networks, and mobile patient.**

I. INTRODUCTION

A typical sensor network consists of a large number of sensor nodes (SNODEs) that are densely deployed either inside the phenomenon or at a close proximity to it. Due to the frequent changes of the sensor network topology, the position of the SNODEs need not be engineered or pre-determined. Furthermore, in addition to being capable of sending the raw data directly to the nodes responsible for the fusion, SNODEs, with the help of the onboard processor, are also capable of locally carrying out simple computations and transmitting only the required and partially processed data. Moreover, sensor networks have self organizing and fault tolerant capabilities. Such capabilities make sensor networks suitable to many applications such as health, environmental, military, security, and home applications [1]. Note that an SNODE is equipped with a limited power [1][5][6][9] where the SNODE lifetime is highly dependent on battery lifetime. Therefore, power conservation and power management is very significant. The main tasks of an SNODE are events detection, data processing and data transmission. Power consumption in an SNODE can be divided into three categories: *sensing, communication and data processing.* While sensing power depends on the nature of the application, periodic updates consume less power than persistent monitoring.

As presented by Akyildiz et. al. [1], a typical architecture of a sensor network is shown in Figure 1 and consists of the sensor field that defines the collection of the scattered SNODEs, the sink through which the data collected from the scattered SNODEs are routed using a multihop infrastructureless architecture, and a task manager node that the sink can communicate with via Internet or satellite.

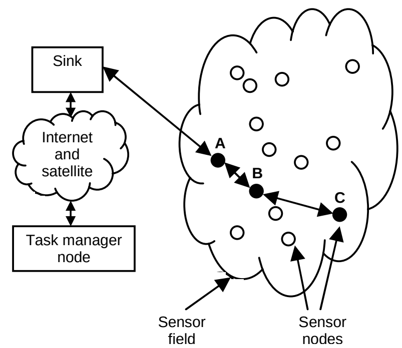

Figure 1. Sensor network architecture

In this paper we use ad hoc sensor networks (ASNET) to facilitate the monitoring of the health conditions of the patients by deploying SNODEs with each patient and possibly SNODEs with doctors and/or nurses. Each SNODE deployed with the patient has a specific task such as reading blood pressure, temperature, and so on. The resulting mobile patient ASNET allows the tracking and the monitoring of patients and doctors inside or outside the hospital. In the case of an emergency, doctors and/or nurses will be contacted automatically through their handheld personal digital assistants (PDAs) or cellular phones. More specifically, the proposed ASNET consists of sensor nodes at the first layer whose responsibility is to measure, collect and communicate, via wired or wireless interface, readings to a microcontroller at the second layer. The implemented ASNET distinguishes between periodic sensor readings and critical or event driven readings where higher priority is given for the latter. The actual setup of the mobile patient ASNET can result in one of the following three configurations:

1. Communication between SNODEs of the patient and the sensor node of the doctor/nurse without a centralized unit: We assume that each patient has small and light weight sensor nodes connected to a microcontroller attached to the patient. Doctors/nurses may also carry sensor nodes that allow other doctors to locate them within the hospital. In this case, the patient's SNODEs send

the data to the SNODE of the doctor/nurse via Wi-Fi through the patient's microcontroller without the need for a central computer system.

2. Communication between SNODEs of the patient and the sensor node of the doctor/nurse via a centralized unit: Similar to the first case, both patient and doctor/nurse have SNODEs. In this case the communication between the patient's microcontroller and the SNODE of the docor/nurse is via the centralized unit.
3. Communication between SNODEs of the patient and the non-senesor node of the doctor/nurse: The doctor's node could be a wireless mobile ad hoc node or a cellular device (hand-held or cellular phone). The doctor/nurse could be inside or outside the hospital. In this case, the communication is between two different networks. The mobile patient's microcontroller sends the data to the centralized system which takes care of sending it to the doctor/nurse.

The doctor/nurse will in turn issue a medical query to the specific mobile patient ASNET [3][4][7][8].

In section 2 we discuss the details of our implementation by describing the hardware, architecture, software, and a pseudo code of the program used. In section 3 we evaluate the performance of the proposed architecture by means of a detailed analysis/simulation. Finally, we conclude our paper.

II. CASE STUDY

As stated in the introduction section, the objectives of the case study are to facilitate the use of both medical sensors and wireless mobile network in health applications. The node of the doctor/nurse could be a wireless ad hoc node, a mobile phone device, or a PDA inside or outside the hospital. By facilitating communication between the patient's microcontroller and the doctor/nurse node the system is capable of:

1. Sending an urgent SMS message to the doctor/nurse node in case of critical sensors reading.
2. Sending an E-mail with the patient profile to the doctor.

In our implementation we used the medical blood pressure sensor MS5536 and the TINI microcontroller with Wi-Fi connectivity.

A. ASNET Architecture

The ASNET deployment scenario used in our implementation is depicted in Figure 2. As shown, SNODEs measure and communicate reading to the microcontroller in charge which in turn forwards processed reports to the central computer. The patient profile is updated with processed information in the central database. The central computer is then responsible for sending emails and/or SMS messages in cases of emergencies. At the bottom of the hierarchy the SNODEs send two types of reading: periodic messages and critical (or event driven) messages. In our implementation, critical messages are given higher priority over periodic messages. The microcontroller buffer implements this policy when reporting data to the central computer.

B. Software

The software component of our implementation can be divided into three main software modules, namely, the e-mail module, the SMS module, and serial peripheral interface module. For the e-mail and the SMS modules, off the-shelf java codes were used to implement the functionalities.

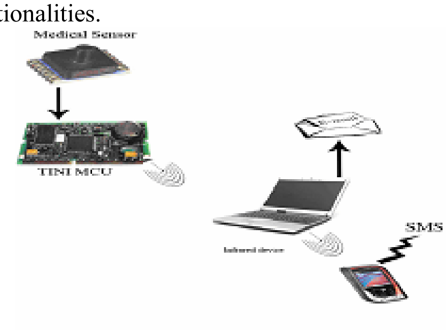

Figure 2. Case study ASNET architecture.

The following describes briefly the operation of each of these modules.

i) E-mail

In our implementation the email program sends e-mails from the central PC through port 25 to the doctor's/nurse's e-mail in case of emergency and at the same time, a pop-up will be displayed on the central computer screen. Furthermore, in case of emergency the sensor monitors the patient continuously rather than periodically.

ii) SMS

In addition to the email sent by the central computer, an SMS message is sent to a list of predetermined phone numbers. The system utilizes a GSM modem, or a cellular phone through an infra-red (IR) interface, to connect to the mobile network infrastructure.

iii) Serial Peripheral Interface (SPI) Code

The MS5536 communicates with the microcontroller digital systems via a 3-wire synchronous serial interface utilizing the SPI protocol. The SPI pseudo code listed in Table 1 communicates with the sensors using the SPI protocol. The parameters needed to construct this protocol are as follows:

- **Delay**: which specify the length of each pulse for the serial clock (SCLK). For the sensor used in our implementation the delay needed is 20 µs.

- **CPOL**: the clock polarity that indicates which state is considered idle for the SCLK (0 or 1).
- **CPHA**: clock phase which specifies when to latch the data (rising/falling edge).

Table 1. SPI pseudo code

```
Class spiExample{
   change the variables to set the delay and time of
fetching
   void reset (){
      send the sequence 10101010101010100000 to
sensor
   }
   void getW(int x){
      if(x==1)          \\WORD  1
      send the sequence 111010101000 to sensor
      if(x==2)          \\WORD  2
      send the sequence 111010110000 to sensor
      if(x==3)          \\WORD  3
      send the sequence 111011001000 to sensor
      if(x==4)          \\WORD  4
      send the sequence 111011010000 to sensor
      wait 50 micro-second
      get the result from sensor
   }
   void getD1(){
      send the sequence 1111010000 to sensor
      wait 50 micro-second
      get the result from sensor
   }
   void getD2(){
      send the sequence 1111001000 to sensor
      wait 50 micro-second
      get the result from sensor
   }
   calculate the pressure from D1 and D2
   if (pressure ≥ threshold)
      send alert message to doctor's mobile
   else
      continue measuring the pressure
```

III. PERFORMANCE EVALUATION

In this section we model a large scale network of the proposed ASNET and provide performance figures in terms of the expected cycle time, and the message waiting time for both types of messages given the priority policy assumed.

The model assumes the microcontroller operates as a queuing server for the SNODEs while the central server polls the group of M microcontrollers for traffic according to the exhaustive policy. Alternative service disciplines can be implemented on the link between the central system and the microcontrollers.

A. Assumptions:

Let the i^{th} ($1 \leq i \leq M$) microcontroller (patient) be connected to N_i sensors where output messages of the j^{th} ($1 \leq j \leq N_i$) sensor are classified into two types: Type 1 (critical messages) or Type 2 (periodic messages). For type 1 messages B_{1j} bytes generated according to a Poisson rate of λ_{sj} message per second. These messages have higher priority over type 2 messages as they represent threshold crossing messages. Type 2 messages are B_{2j} bytes that are generated every T_j seconds.

A microcontroller processes messages at the rate of C_m bytes per second while a number of M microcontrollers are connected to the central system using IEEE802.11 wireless network. The central system polls each peripheral device and processes messages at the rate of C_c bytes per second. In the lab implementation, the sensor in critical mode persists on sending reading at a rate much greater than that for the periodic mode. This is modeled by assuming B_{1j} is much greater than B_{2j}.

B. Analysis of the polling network:

To analyze the polling network, we ignore the buffering effect at the microcontrollers. This is justified since the processing power of these microcontrollers far exceeds the amount of offered work by the group of N_i sensors. Therefore, the central server when serving the i^{th} (i=1, 2, …, M) station is serving the sensors output connected to i^{th} microcontroller.

Assuming a cyclic exhaustive service over the wireless LAN interfaces, the central server polls each of the microcontroller cyclically and serves all buffer contents. The analysis below is divided into two steps:

1. Computation of the mean cycle time, T_c., and

2. Computation of the mean waiting time for messages assuming priority is given to critical messages over periodic messages.

We can show that the mean cycle time should be given by

Error! Objects cannot be created from editing field codes.

where traffic intensity due type-1 messages, ρ_1, is equal to**Error! Objects cannot be created from editing field codes.**, the traffic intensity due to type-2 messages, ρ_2, is equal to **Error! Objects cannot be created from editing field codes.**. δ_i is the time it takes the central server to switch from i^{th} station to the $(i+1)^{st}$ station.

The above relation holds for the mixed input specified earlier. However, unfortunately, the exact analysis for the message waiting time for non Poisson input is not available [10]. Therefore, message waiting times will be reported using simulations. To further simplify the results, we also assume that all mobile patients have the same number of sensors connected to them and that all sensors produce the same intensity of traffic (i.e. , $\lambda = \lambda_{sj}$ and $T = T_{2j}$ for j=1,2, …, M)

Table I. list the mean cycle time, mean message waiting time of type 1, and mean message waiting time of type

2, all in milliseconds, in the form of triplets (T_c, $E[W_1]$, $E[W_2]$) for different choices for rate of arrivals for critical messages, λ and T. The performance figures are evaluated at an offered load intensity of 70%. The figures in table assume a DSS IEEE802.11 setup where channel rate is 1 Mb/s. The polling overhead (T_poll + T_ack + 2 SIFS) is equal to 0.74 msec. Furthermore, we assume the sensor's payloads for periodic and critical messages are equal to 512 and 100K bytes. The number of sensor nodes connected to each microcontroller, N, is given by 10.

Table 2: Performance at 70% traffic intensity.

λ (arrival/min)	*T* (seconds)		
	5	20	30
1/1	(12, 926, 1083)	(12, 961, 1118)	(12, 904, 1058)
1/5	(58, 858, 899)	(131, 1932, 1993)	(111, 1442, 1552)
1/15	(137, 646, 692)	(151, 797, 818)	(169, 919, 952)
1/30	(157, 443, 467)	(313, 973, 974)	(303, 872, 892)
1/60	(175, 294, 299)	(409, 712, 721)	(949, 1526, 1563)

It can been noted from table that mean cycle time as well as mean message waiting times increase as the traffic load increases. However, the difference between message waiting times for type 1 and type 2 are more recognizable at scenarios where load offered from critical message is relatively high compared to load offered from periodic messages. For example Figure 3 shows the mean waiting times for both types of traffic versus the overall traffic intensity for the network for such scenario where λ is equal to one arrival per minute while T is equal to 20 seconds. For scenarios where the arrival rate of critical messages is considerable lower than that for periodic messages, then the waiting times for the two types of traffic are almost equal. This is shown in the curves depicted in Figure 4 for the scenario of λ is equal to one arrival per 60 minutes while T is equal to 20 seconds.

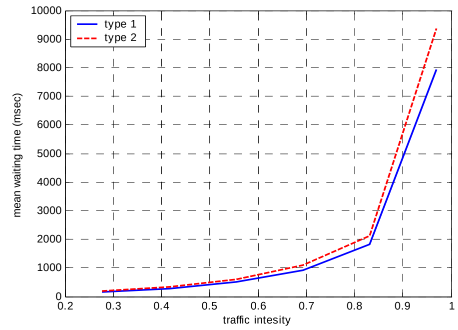

Figure 3. Mean message waiting time for λ = one arrival per minute and T = 20 seconds.

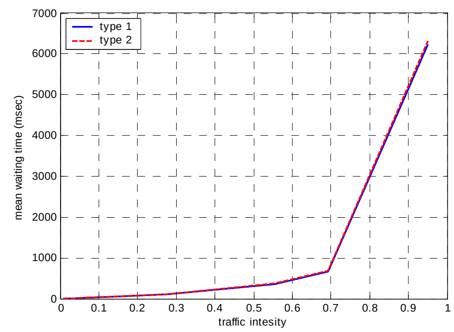

Figure 4. Mean message waiting time for λ = one arrival per 60 minutes and T = 20 seconds.

IV. Conclusions

In this paper we have explored three telemedicine scenarios where ASNET is utilized to communicate medical reading of patients to a central system through the intervention of middle hierarchy devices such as microcontrollers. The proposed architecture that consists of three layers was implemented using off-shelf components and packages at the wireless labs of the computer engineering department of KFUPM. An SPI protocol was developed for communication between sensor nodes and a microcontroller. A wireless LAN network has been created to connect microcontrollers to the central system and PDAs of doctors/nurses. The paper also provides a model that quantifies the expected performance of the network. The priority-based polling scheme between the central server the group of mobile patients. The preliminary results indicate that the priority scheme manages to keep the mean waiting time for critical message low compared to periodic messages sent by the SNODEs. However, the mean waiting time, for the selected implementation, is dominated by the polling intervist time and not greatly affected by the queuing time experienced in microcontroller buffer.

V. ACKNOWLEDGEMENTS

The authors acknowledge KFUPM and the department of computer engineering for support in this research.

References:

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cyirci. “Wireless Sensor Networks: A Survey,” Computer Networks, 38(4):393-422, 2002.

[2] Fei Hu, Jason Tillett, Jim Ziobro, and Neeraj K. Sharma “Secure Tree-zone-based Wireless Sensor Networks for Telemedicine Applications,” Proceedings of Globecom 2003.

[3] Yih-Chun Hu, Adrian Perrig, David B. Johnson. Ariadne: “A secure On-Demand Routing Protocol for Ad hoc Networks”, MobiCom 2002, September 23-28, 2002, Atlanta, Georgia, USA.

[4] Panagiotis Papadimitratos and Zygmunt J. Haas, “Secure Routing for Mobile Ad hoc Networks SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002),” San Antonio, TX, January 27-31, 2002.

[5] A. Porret, T. Melly, C. C. Enz, E. A. Vittoz, “A Low-power Low-voltage transceiver Architecture Suitable for Wireless Distributed Sensors Network,” IEEE Interational Symposium on Circuits and Systems’00, Geneva, Vol. 1, 2000 pp 56-59.

[6] L. Li, J. Y. Halpern, “Minimum-energy Mobile Wireless Networks revisited,” IEEE International Conference on Communications ICC’01, Helsinki, Finland, June 2001.

[7] N. Noury, T. Herve, V. Rialle, G. Virone, E. Mereier, G. Morey, A. Moro, T. Porcheron, “Monitoring Behavior in Home Using A Smart Fall Sensor, “ IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology, October 2000, p. 607-610.

[8] M. Ogawa et al., “Fully Automated Biosignal Acquisition in Daily Routine through 1 Month,” International Conference on IEEE-EMBS, Hone Kong, 1998, pp. 1947-1950.

[9] J. M. Rabaey, M. J. Mmmer, J. L. da Silva Jr., D. Patel, S. Roundy, “Pico Radio Supports Ad Hoc Ultra-low power Wireless Networking,” IEEE Computer Magazine (2000), pp. 42-48.

[10] H. Takagi, “Queueing Analysis of Polling Models,” ACM Computing Surveys, Vol. 20, Issue 1 (March 1988), pp. 5-28.