Chapter 2
Requirements, Challenges, and Summary of Hardware and Software Design for a WSN-Based SHM System

Compared to the short-term systems, a typical long-term WSN-based SHM system poses many challenges. In this book, without specifying otherwise, we focus on the design of long-term WSN-based SHM systems. In this chapter, we first describe system requirements. Then we raise a few challenges when designing such a SHM system using WSNs. Finally, we summarize the hardware and software design of our SenetSHM platform.

2.1 Requirements and Challenges

Our collaborators from civil engineering specified the following system requirements for a typical long-term SHM system.

For long-term WSN-based SHM systems, continuous collecting data are neither required nor feasible. It is highly preferable for the system to work only during the occurrence of some certain kinds of events such as earthquake and large wind, etc. In other conditions, sensor nodes are put into sleep to save energy. Thus *event-triggered wakeup* is necessary. Moreover, the wakeup of sensor nodes should be *fast, network-wide, and reliable*. Particularly, the network-wide wakeup means that during the events, all the deployed sensor nodes, even far from the location of event sources, should be awake and start sampling. Second, the system should be able to provide real-time or near-real-time *healthy information* of the structure under monitoring. Last but not the least, a WSN-based SHM system which solely relies on battery should be able to work for weeks or even months.

However, to meet the requirements above entails many challenges. The first one is associated with the network-wide and reliable wakeup in the presence of critical events. In comparison, sensor nodes in [1] work at fixed duty cycle (about 6 min every day), which limits the ability of the application users to initiate network operations at random or can miss the event of interest. The 'SnoozeAlarm' with 'sentry-based' approach in [3] requires constant power supply for sentry nodes. Moreover, the sleep

J. Cao and X. Liu, *Wireless Sensor Networks for Structural Health Monitoring*,
SpringerBriefs in Electrical and Computer Engineering,
DOI 10.1007/978-3-319-29034-8_2

sensor nodes are awakened one by one by a central gateway, which leads to a long wakeup time of the network (1–5 min) after the event is first detected. As a result, the system does not support capturing critical data in short-term, transient events such as an earthquake. The system in Brimon [2] is only limited to railway bridge monitoring. And to wake up sensor nodes deployed on the bridge, oncoming trains passing by need to have a node installed to broadcast beacons constantly.

Another issue in long-term SHM is embedding effective SHM algorithms within WSNs. To provide real-time or near-real-time healthy information as well as to save energy, we generally need to implement SHM algorithms within a WSN. However, SHM algorithms used in traditional wire-based SHM systems generally have two properties:

- The SHM algorithms are centralized and require the raw data from deployed sensor nodes.
- The SHM algorithms involve complicated signal processing techniques and require powerful computation units.

The two associated requirements pose significant challenges for a WSN-based SHM system, considering the limited power computational capability (i.e., power CPU and a large memory) of wireless sensor nodes. Some SHM algorithms simply cannot be implemented, or it takes significant amount of time to finish the computation, even longer than transmission the raw data. In addition, we also to need to consider the necessary wireless communication needed for a SHM algorithm.

The last challenge is related to fault tolerance. As was in other WSN applications, wireless sensor nodes deployed in a SHM system can have various types of faults. Moreover, we are interested in detecting faulty sensor readings. Despite of many existing fault-tolerance schemes to address faulty readings in WSNs, they are not able to work well in SHM because it leverages a different model to detect event (i.e., structural damage). Some assumptions in the existing fault-tolerant event detection are not valid in SHM.

2.2 Hardware Design

The hardware architecture of SenetSHM is shown in Fig. 2.1. The SenetSHM platform includes an Imote2 and a specially designed sensor board. Imote2 is chosen as the central unit because it has a good balance between low power consumption and rich resources. Using Imote2, implementing complicated SHM algorithms becomes possible. Imote2 also integrates a radio transceiver CC2420 which will take the main responsibility of transmitting raw data and control commands in SenetSHM.

However, Imote2 alone misses some key components which are necessary for the SenetSHM. Therefore, we design a sensor board that can be attached to the Imote2 to fulfill the following functions:

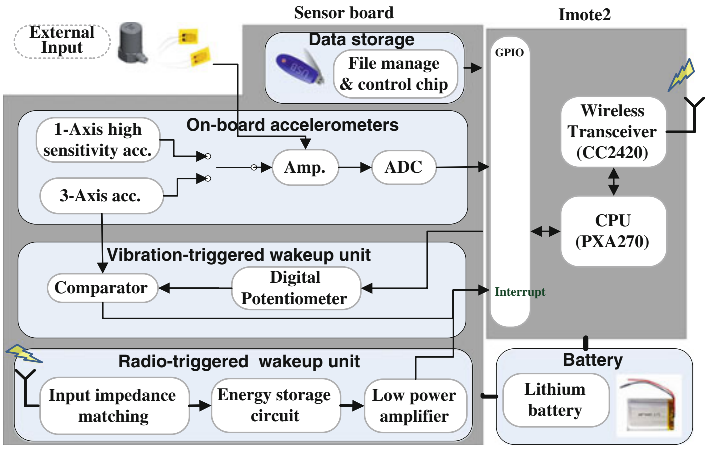

Fig. 2.1 The hardware architecture of the SenetSHM

- **Support for on-board accelerometers and external input**: The sensor board provides abundant support for measuring accelerations. The on-board sensors contain a general-purpose three-axis accelerometer, LIS344ALH, and another high sensitivity one-axis on-board accelerometer SD1221. According to different application scenarios, users can specify which accelerometer will be used through a switch or use them simultaneously. Signals generated from these two sensors are amplified and are transformed to the digital format through a programmable ADC QF4A512. Besides on-board sensors, other types of external sensors such as piezoelectric accelerometers, strain gauges, can also be directly attached with the SenetSHM nodes.
- **On-board data storage**: We also design interfaces on each SenetSHM node for μSD and USB. For each node, the measured data with the associated time stamps are stored into the μSD or the USB in a real-time manner. Both the μSD and the USB used have 2G Byte space, allowing the storage of raw data continuously sampled at 1 KHz for more than two days.
- **Vibration-triggered wakeup and radio-triggered wakeup units**: For long-term SHM, we designed two units, vibration- and a radio-triggered wakeup units and they work together to realize fast and unified wakeup of sensor nodes in the presence of some events. The vibration-triggered wakeup unit will wake up the attached sensor node from deep sleep mode when the vibration of the structure exceeds a pre-defined threshold. On the other hand, the radio-triggered wakeup unit will wake up the sensor node when it receives wakeup messages from others. Different from the 'SnoozeAlarm' mode in [3], SenetSHM nodes with radio-triggered wakeup unit do not need to wake up periodically to listen to the wireless channel and therefore can be more energy efficient and fast. How these two wakeup units collaborate will be described in Chap. 3.

2.3 Software Design

Figure 2.2 illustrates the software architecture of the SenetSHM. We design a middleware for SenetSHM which adopts service-oriented architecture (SOA). Using SOA, the complicated software system is divided into smaller, more manageable services. Particularly, the middleware provides an application programming interface for application users. For different applications of SHM such as short-term and long-term SHM, application users can simply choose from these services and compose them together to constitute the service that is needed.

The basic services contained in the middleware include *the sampling service*, which mainly deals with techniques to realize synchronized sensing; *the wireless communication service*, which supports one-to-one (e.g., for threshold setting), one-to-all (e.g., for time synchronization), all-to-one (e.g., for data collection), and all-to-all (e.g., for wakeup mechanism) wireless communications; *the data storage service*, which allows on-board data storage for short-term SHM; *the wakeup service*, which is used to provide different methods for fast, unified, and synchronized wakeup for long-term SHM applications; *the structural status service*, which provides healthy status of the structure in long-term SHM applications; and the other services such as *the fault-tolerant service* and *other services for maintenance and debug*.

In the following three chapters, we will describe how the SenetSHM addresses the following three important issues: (1) network-wide and event-triggered wakeup, (2) distributed processing of SHM algorithms, and (3) realizing fault-tolerant SHM in WSNs.

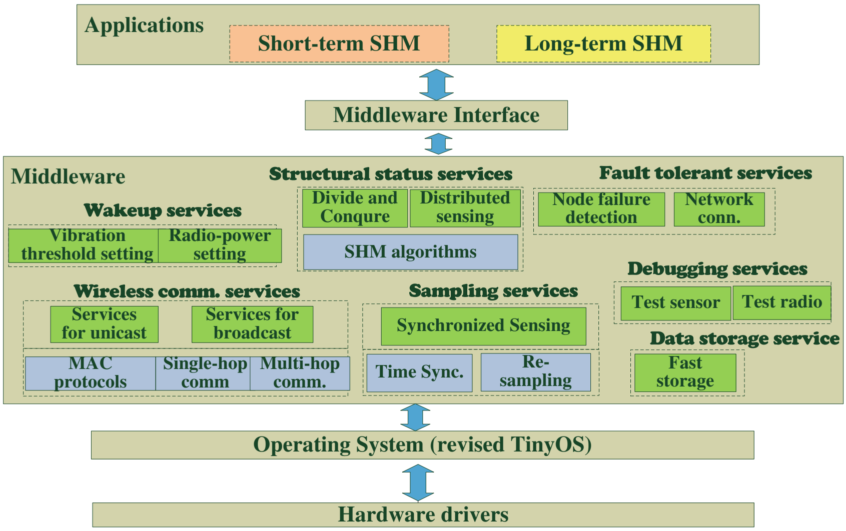

Fig. 2.2 The software architecture of the SenetSHM

References

1. M. Ceriotti, L. Mottola, G.P. Picco, A.L. Murphy, S. Guna, M. Corra, M. Pozzi, D. Zonta, P. Zanon, Monitoring heritage buildings with wireless sensor networks: the torre aquila deployment, in *Proceedings of the 2009 International Conference on Information Processing in Sensor Networks* (IEEE Computer Society, 2009), pp. 277–288
2. K. Chebrolu, B. Raman, N. Mishra, P.K. Valiveti, R. Kumar, Brimon: a sensor network system for railway bridge monitoring, in *Proceedings of the 6th international conference on mobile systems, applications, and services* (ACM, 2008), pp. 2–14
3. J.A. Rice, K. Mechitov, S.H. Sim, T. Nagayama, S. Jang, R. Kim, B.F. Spencer Jr, G. Agha, Y. Fujino, Flexible smart sensor framework for autonomous structural health monitoring. Smart Struct. Syst. **6**(5–6), 423–438 (2010)

http://www.springer.com/978-3-319-29032-4

Wireless Sensor Networks for Structural Health
Monitoring
Cao, J.; Liu, X.
2016, X, 99 p. 55 illus. in color., Softcover
ISBN: 978-3-319-29032-4