Wireless Structural Sensors using Reliable Communication Protocols for Data Acquisition and Interrogation

Yang Wang
Department of Civil and Environmental Engineering
Stanford University
Stanford, CA 94305-4020

Jerome P. Lynch
Department of Civil and Environmental Engineering
University of Michigan
Ann Arbor, MI 48109

Kincho H. Law
Department of Civil and Environmental Engineering
Stanford University
Stanford, CA 94305-4020

ABSTRACT

A new design of a wireless sensing system intended for structural monitoring is proposed. Key performance attributes of the wireless sensing unit include simultaneous accommodation of multiple sensors, high-resolution analog-to-digital data conversion, overall low-power demand and a computational core. The computational core is a powerful component of the design that allows real-time execution of embedded engineering procedures. Capabilities of the wireless sensing units for peer-to-peer communication enables collaboration between wireless sensing units with little to no penalty on the power consumption of the global system, broadening the role wireless sensing units can play in screening structures for indication of damage. To ensure the robustness of the data acquisition, state machine concepts are explored in the communication protocol design. Various validation tests are performed on a laboratory model structure to highlight the performance of the hardware and software components of the prototype wireless monitoring system.

INTRODUCTION

Civil infrastructure systems are one of the most expensive assets that countries invest (e.g. estimated at $20 trillion in the U.S. [1]). The safety and reliability of these infrastructure systems are essential for supporting the commerce, economy and social security of a nation. However, because of improper usage, material decaying, or structural damages resulting from different types of hazards, conditions of most current civil infrastructures are declining quickly. For instance, in the United States, more than half of all bridges were built before the 1940's, and approximately 42 percent of them are reported to be structurally deficient [2]. To precisely evaluate safety conditions of existing infrastructure systems, and to accurately identify structural vulnerabilities to extreme events, strong interests in various structural health monitoring technologies have been growing rapidly in recent years [3]. Structural health monitoring can provide insights into the real performance of a structure, and also offer empirical data that is helpful for refining structural models and existing building codes. In California, over 900 permanent structural sensors have been installed on 60 long-span bridges by the California Department of Transportation since 1977 [4]. Furthermore, installation of structural monitoring systems for buildings in highly active seismic zones is mandated by current California Building Code.

Traditional structure health monitoring technology has employed wire-based systems to collect structural data. However, the installation of these wire-based systems can be expensive in labor, time and price. For example, a twelve-channel wire-based system may cost about $50,000, with half of the expense associated with its installation, including labor, cabling, etc. [5]. Moreover, the installation of the wired systems can consume about 75% of the total testing time for large structures [6]. In order to reduce these monetary and time expenses for the installation of wire-based systems, new technologies in embedded systems and wireless communication have been adopted in academic and industrial research for wireless sensing and monitoring. The use of wireless communication for SHM data acquisition was illustrated by Straser and Kiremidjian [6]. Their work demonstrated the potential and cost-effectiveness of wireless monitoring systems. More recently, Lynch *et al*. extended the work by embedding damage identification algorithms into a wireless sensing unit; their work harnessed the unit's computational power for decentralized data interrogation [7]. Meanwhile, several other research groups have been developing various types of wireless sensing networks [8-10], many of which are generic systems that do not yet fit the unique demands of a structural health monitoring system.

For applications in civil infrastructures, a wireless monitoring system is expected to provide the capability for relatively long-distance communication within the span of the structure (usually from tens of meters to hundreds of meters), as well as sufficient capability for local data analysis and processing. On the other hand, since wireless sensing units will most likely operate on portable batteries with finite energy, their power consumption must be recognized by potential end users. Because long-distance communication and local data interrogation capability usually demand more power, balancing the conflict between the requirements for higher data processing and communication capacity and lower power consumption becomes one of the major challenges in designing a wireless structural sensing and monitoring system. Another major challenge for a wireless structural monitoring system is the reliability and accuracy of data acquisition. A data acquisition process includes analog-to-digital conversion of sensor signals, the temporary storage of digital data, and the transfer of digitized signals by wireless communication. Problems such as circuit noise and occasional wireless communication failure should be handled properly for reliable and accurate data acquisition.

To better address the above challenges, a novel design of a wireless structural monitoring system is proposed in this paper. The wireless structural monitoring system contains multiple wireless sensing units, which can simultaneously collect and analyze data from multiple heterogeneous analog sensors. High-precision analog-to-digital conversion of multi-channel sensor signals is implemented in the wireless sensing unit. For wireless communication, each unit employs a specially selected wireless modem that consumes relatively low power and supports long-distance peer-to-peer communication. A microcontroller, coupled with a considerable amount of external memory, is used to manage the collection of sensor data, local data storage, and wireless data streaming. Its computational power and the associated external memory are sufficient to support local data analysis. A specially designed data acquisition protocol is embedded in the microcontroller, enabling reliable real-time and near-synchronized data acquisition from multiple sensing units associated with multiple sensors. Shake-table tests have been conducted to validate the performance of this newly designed wireless structural monitoring system.

HARDWARE DESIGN OF THE WIRELESS SENSING UNIT

The hardware design of the newly devised wireless sensing unit reflects the demands of structural monitoring applications. In the design, the power consumption constraints are balanced by the requirements for long-distance communications and computational capabilities. The hardware design has also been focused on improving the unit's performance for high-precision real-time data acquisition. A functional diagram of the proposed wireless sensing unit is illustrated in Figure 1. The unit includes three subsystems: the sensing interface, the computational core, and the wireless communication system. The sensing interface is responsible for converting the analog sensor signals into digital forms. The digital data is then transferred to the computational core by the Serial Peripheral Interface (SPI). External memory is associated with the computational core for local data storage or analysis. Through the Universal Asynchronous Receiver and Transmitter (UART) interface, the computational core communicates with the wireless communication module that sets up wireless connection between the unit and other devices within a network.

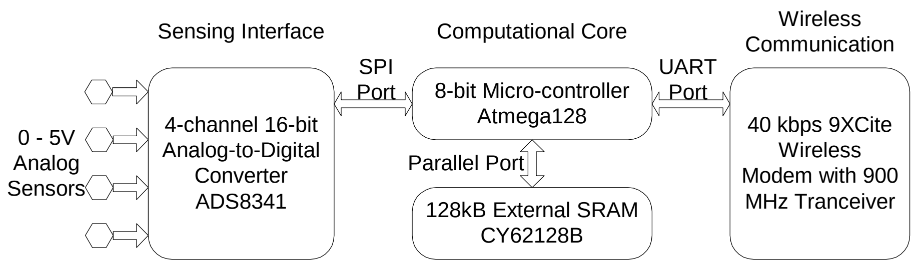


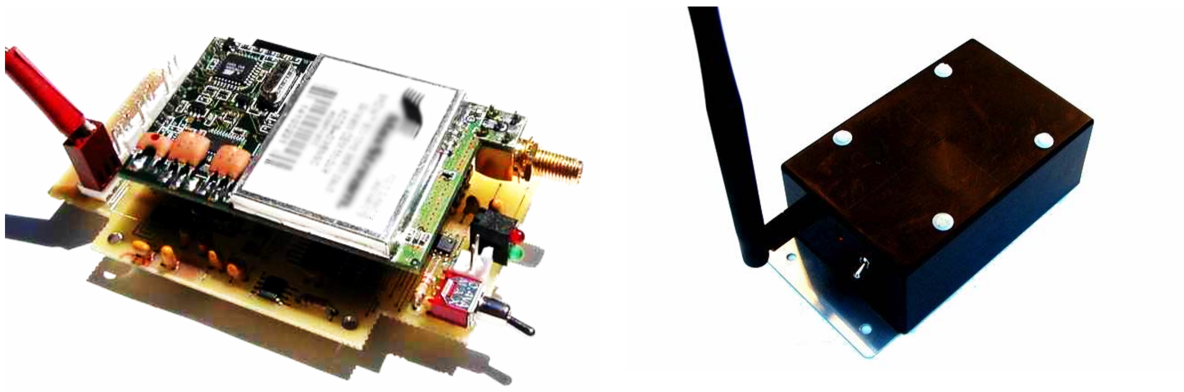
Figure 1 – Hardware functional diagram of the wireless sensing unit prototype

In the sensing interface module, a four-channel, 16-bit, and 100 kHz analog-to-digital (A/D) converter, Texas Instrument ADS8341, is selected for converting analog sensor signals into digital data that can be recognized by the microcontroller. Any analog sensor signal between 0 and 5V can be accepted by the A/D converter, so that the sensing unit is sufficiently generic for accommodating heterogeneous analog sensors. The A/D converter can be interfaced with up to four sensors at the same time, and its 16-bit resolution provides adequate accuracy for most applications in structural health monitoring. Although a sampling rate of 100 kHz is much higher than what is usually needed for monitoring civil infrastructures, this high sampling rate makes it possible for real-time data acquisition with the unit. The reason is that because each A/D conversion is brief and fast, even when the unit is conducting wireless communication with other devices, the A/D conversion can still run intermittently in the background, without interrupting the wireless communication.

For the computational core of the wireless sensing unit, a low-power microcontroller is employed to coordinate all of the different parts of the sensing unit hardware, and to provide a capability for local data interrogation. A low-cost eight-bit Atmel AVR microcontroller, Atmega128, is selected in this design. The Atmega128 microcontroller provides a flash memory (ROM) of 128kB, which is usually enough for storing embedded software. When the microcontroller is running at a system clock of 8MHz, it consumes less than 20mA of current at a power supply voltage of 5V. The 64-pin Atmega 128 provides UART/SPI communication interfaces, timer modules, interrupt modules and multiple input/output ports. Its timer and interrupt modules are used to set up the background A/D conversion at specified sampling rate. Because the 4kB SRAM integrated in the microcontroller is not large enough for sensor data storage and analysis, external memory of 128kB (Cypress CY62128B) is interfaced with the microcontroller. Although there is a limitation of the Atmega 128 microcontroller to only allow accessing 64kB of external memory at one time, it is still possible to make full use of external memory by controlling a separate line that selects the lower half 64kB or upper half 64kB of external memory. The external memory is sufficient for executing many sophisticated damage identification algorithms.

Maxstream 9XCite wireless modem is selected in this study for the wireless communication subsystem. This wireless modem provides the trade-off and balance between low power consumption and long communication distance for applications in structural health monitoring. Its outdoor line-of-sight communication range is up to 300m, which is reduced to about 100m when it is used indoors. Meanwhile, the 9XCite modem consumes a current of only about 50mA when transmitting data, or a current of about 30mA when receiving data. A much lower current is consumed when the 9XCite modem is set in sleep mode. The 9XCite modem communicates with the microcontroller though UART communication at a baud rate of 38400 bps, which is also the maximum wireless transfer rate the modem can provide. Peer-to-peer communication is supported by 9XCite wireless modems, which means that instead of being able to communicate with a central server, each wireless sensing unit can also communicate with other units. This property makes the wireless modem ideal for implementing decentralized structural monitoring algorithms, because it enables direct communication between wireless sensing units, without relaying the data through a master station.

In the current preliminary design, the circuit schematics are printed on a two-layer circuit board, which has a dimension of 3.82” by 2.36”. Surface mounting IC components are adopted to reduce the size of the circuit board. Sockets for connections with power supply, sensors, wireless modem, and microcontroller programming wire are embedded on the circuit board. Figure 2(a) shows the picture of the prototype circuit board with the wireless modem mounted on the top layer. Figure 2(b) shows the complete package, with the circuit board, the wireless modem, and the AA batteries inside a 4.02” x 2.56” x 1.57” weatherproof plastic container.



(a) (b)

Figure 2 – Pictures of the wireless sensing unit prototype and the complete package

SOFTWARE DESIGN OF THE WIRELESS MONITORING SYSTEM

With the hardware of the wireless sensing units designed and fabricated, a scalable software platform is developed to achieve the requested functionality of the wireless monitoring system. First, software that can be embedded into the microcontroller, also called firmware, is implemented to organize the operations of the wireless sensing unit's different hardware modules. Second, if the collected data is to be streamed into a central server, server-side software is also necessary to manage the entire wireless monitoring system that may include multiple wireless sensing units, and communicate with each unit for data acquisition. This paper describes a data acquisition software framework that includes both the software for the wireless sensing unit, and the software for the central server.

A robust data acquisition software system should be sufficiently reliable and able to detect the wireless communication failures and successfully recover the system whenever a communication failure happens. Because of the system complexity needed to ensure the reliability of the wireless communication channel, state machine concepts are explored for the design of software architecture [11]. A state machine consists of a set of states and the definition of a set of transitions among these states. At any point in time, the state machine can only be in one of the possible states. In response to different events, the machine transits between its discrete

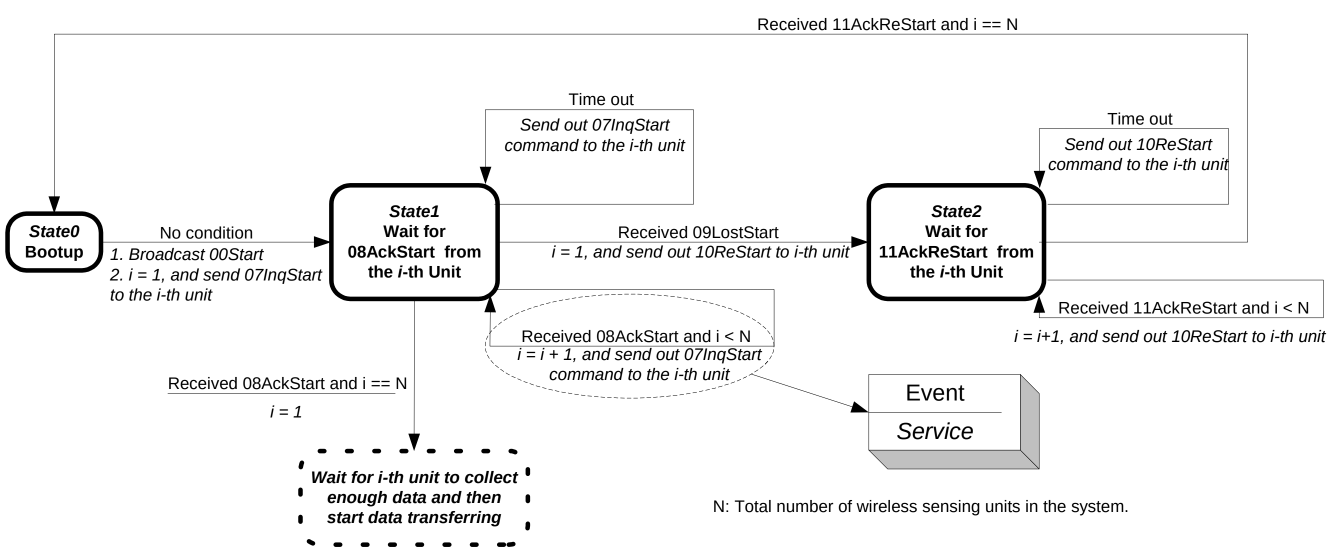


Figure 3 – Abridged communication state diagram for the central server

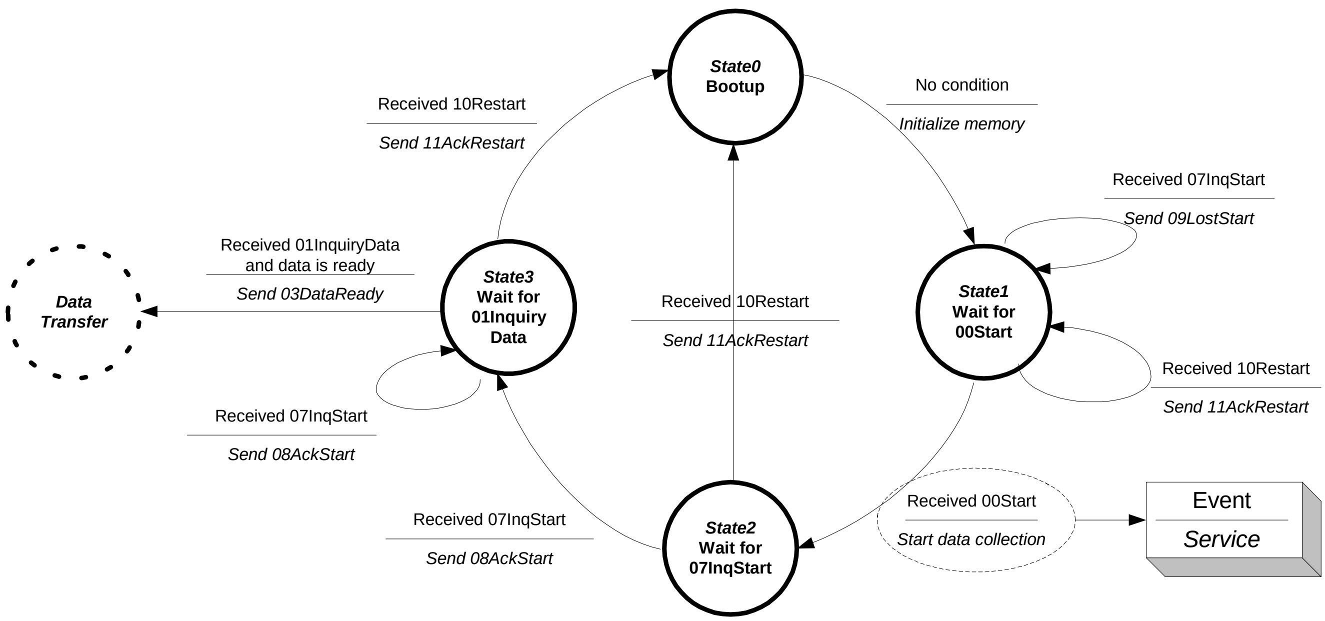


Figure 4 – Abridged communication state diagram for the wireless sensing unit

states. Figure 3 shows the state machine for the central server software, and Figure 4 shows the state machine for the wireless sensing unit software. For simplicity, only part of the communication diagram is presented. In the state diagram, each rectangle or circle with bold boundary lines stands for one possible state; lines with arrows represent state transitions. As shown in the legend, for each transition, the normal text above the horizontal line specifies the event/condition after which the transition should happen, and the italic text below the horizontal line specifies the service/action that should be completed during this transition.

When a wireless sensing unit is powered on, the unit starts from "*State0* Bootup" in Figure 4. Under "no condition", the unit automatically initializes the memory space, and transits into "*State1* Wait for 00Start". '00Start' is the command that is broadcasted from the central server to all the wireless sensing units, requesting all units to start data collection simultaneously. When the data collection program at the central server starts running, as illustrated in Figure 3, the server will automatically broadcast the '00Start' command to all the wireless sensing units. As soon as the wireless sensing unit receives and recognizes the '00Start' command, the sensing unit starts collecting data from sensors at a specified sampling rate, and saves the data temporarily into its external SRAM for later acquisition by the central server. If all the wireless sensing units are assumed to take the same amount of time to receive and recognize this '00Start' signal, then all the units start recording data at the same time, i.e. the data acquisition is synchronized.

As shown in Figure 4, after the wireless sensing unit receives the '00Start' command, the unit transits into "*State2* Wait for 07InqStart". Accordingly, the central server broadcasts the '00Start' and checks all the units in turn to confirm that all the units have received the '00Start' command. The central server sends '07InqStart' to each unit, and waits until the server receives '08AckStart' from the sensing unit, confirming that the unit has received the '00Start' signal. If any one of the wireless sensing units misses the '00Start' command, this unit will receive '07InqStart' command in "*State1*" (in stead of in "*State2*"), because the unit is still waiting for the '00Start' command. In this case, the wireless sensing unit will send '09LostStart' in response to the central server's '07InqStart' inquiry. Knowing that the sensing unit has not properly received the '00Start' command, the central server will ask all the units to restart and try the whole procedure again from the beginning, until the central server confirms that all the units have received the broadcasted '00Start' signal correctly.

The communication protocol is designed to address the reliability problem of wireless communication. To reduce the computational task of the wireless sensing units that have a limited power source from the battery pack, the central server is assigned the responsibility to ensure reliable wireless communication: the central server plays an “active” role in the communication, while the wireless sensing unit plays a “passive” role. After the central server sends a command to the wireless sensing unit, if the server cannot receive an expected response from the unit in certain time, the server will resend the last command again until the expected response is received. However, on the other side, after a wireless sensing unit sends a message to the central server, the unit doesn’t concern if the message has arrived at the central server correctly or not. The reason is that even if this message is lost in the air, the central server will be able to realize this and resend the last command, notifying the sensing unit to resend the lost message. To illustrate this, assume that when the wireless sensing unit transits from “*State2*” to “*State3*”, the ‘08AckStart’ message sent from the unit to the central server is lost. The central server is now at its “*State1*” waiting for the ‘08AckStart’, but the server cannot receive this message after expected time, so the server will resend ‘07InqStart’ to the unit. Therefore in the “*State3*” of the wireless sensing unit, although the unit is waiting for data acquisition from the central server, the unit may still receive ‘07InqStart’ command because the loss of the last ‘08AckStart’, and the unit has to handle this situation by resending ‘08AckStart’ to the central server. As shown in this example, state machine concepts are important for designing the complicated communication protocols.

After confirming that all wireless sensing units have started data collection, the central server collects data from each unit one by one. Retry and acknowledgement protocol has also been implemented to ensure the fidelity of data transfer. To achieve real-time data collection, two memory stacks are designed in each unit. After one stack is full of data to be transferred to the central server, the unit writes in the other stack, while waiting for the central server to acquire data from the stack that is already full. Because each A/D conversion is brief and fast, the wireless data transfer does not interfere with the A/D conversion. Even though the unit is sending data to the central server, the A/D conversion and data storage keep running in the background.

LABORATORY VALIDATION TESTS

A simple test is devised to check the performance of the wireless sensing units in collecting synchronized dynamic data. Two wireless sensing units are powered up and wait for the ‘00Start’ command from the central server at the same time. Then the server-side program is started, broadcasting the ‘00Start’ command to the two units. A special test program is embedded in each wireless sensing unit, so that once the sensing unit receives and recognizes the ‘00Start’ command, the microcontroller changes the status of one of its digital output pins. An oscilloscope is used to monitor the status changes of these two output pins from the two wireless sensing units, and the time difference between the status changes can be measured by the oscilloscope directly. In this test, the measured time difference between the ‘00Start’ signal receptacle and recognition by the two units is only about 10μS, which means that one wireless sensing unit starts collecting sensor data 10μS earlier than the other. This time difference is the synchronization error that is to be found in this test, and a synchronization error at this level is usually negligible for most structural dynamic analysis. When the two units are not so close to each other, this time difference is difficult to be measured accurately, and the data synchronization could be worse.

To validate the performance of the entire wireless monitoring system, validation tests on a three-story laboratory structure are devised. Figure 5 shows the aluminum structure used for the laboratory test, and a picture of the shake-table test. Each floor of the structure weighs about 7.26kg. The lateral stiffness of the structure is provided by four aluminum columns, each of which has a cross section of 0.64cm by 1.27cm. For theoretical computation, the three-story structure is simulated as a lumped-mass shear frame model. In another room about 15 meters away, a 9XCite wireless modem is connected with a computer as the central server for the data acquisition.

Two types of accelerometers are used for the validation test, exemplifying the ability of the system to accommodate heterogeneous analog sensors. A Crossbow CXL02LF1 accelerometer, which has a RMS (Root-Mean-Square) noise floor of $9.8 mm/s^2$, is placed on the ground, the first, and the third floor, respectively. A Bosch SMB110 accelerometer, which has a RMS noise floor of $66.5 mm/s^2$, is placed on the second floor. As shown by their RMS resolution, the CXL02LF1 accelerometer provides better performance than the SMB110 in noise reduction.

In the first test, the structure is provided a random initial velocity and displacement by the shake-table, then the shake-table stops moving and the structure vibrates freely. A wireless sensing unit associated with an accelerometer is mounted on the third floor, to record the acceleration time history and transfer the data to the central server. Figure 6 shows the Discrete Fourier Transform to the measured acceleration time-history of the third-floor when the structure is under free vibration. The three natural frequencies extracted from the three peaks of the DFT plot are 2.07Hz, 5.73Hz, and 8.27Hz, while the three theoretical natural frequencies computed from the structural simulation model are 2.08Hz, 5.71Hz, and 8.18Hz, respectively.

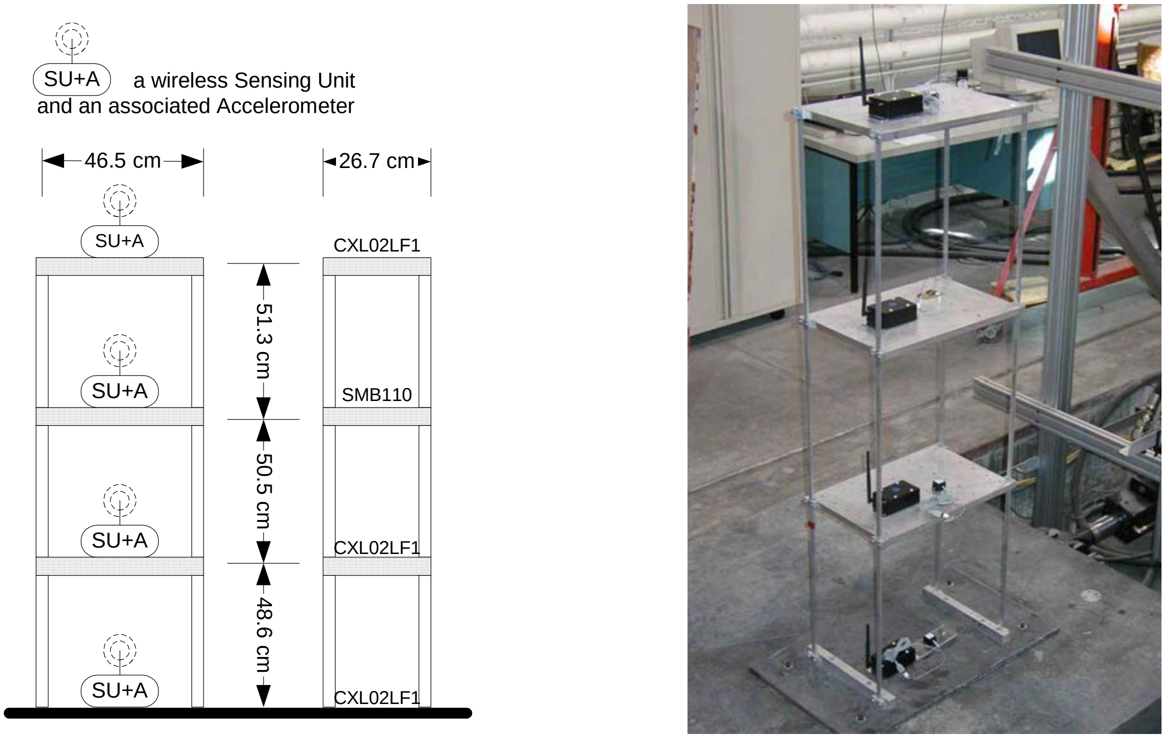


Figure 5 – Test structure for laboratory validation of the wireless monitoring system

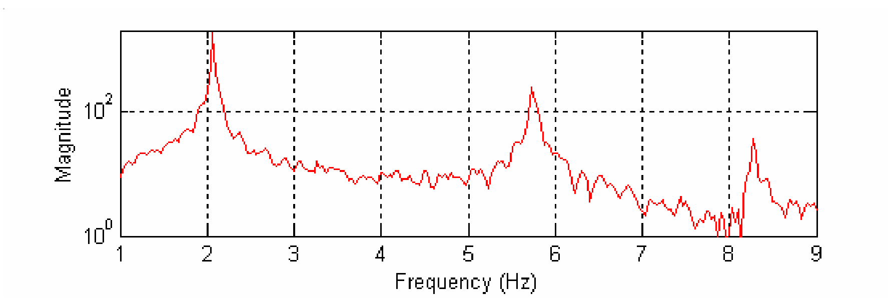


Figure 6 – Discrete Fourier Transform of the third-floor acceleration during free-vibration

In the second test, acceleration time history of each floor is measured when the structure is excited by a one-directional ground motion that is applied along the longitude of the structure. As shown in Figure 5, accelerometers are placed on each floor and also the ground level. Each accelerometer is associated with a wireless sensing unit that can collect data from the accelerometer and transfer the data to the central server. Using the communication state diagram illustrated before, the central server acquires near-synchronized dynamic data from all the wireless sensing units at real-time. Although there are occasional wireless communication failures during the test, they are all handled and fully recovered by the designed communication protocol using the state machine concepts. The sampling frequency used in this test is 200Hz, which should be high enough to capture the major characteristics of the structural vibration.

The designed ground excitation is a chirping signal that has constant displacement amplitude with a linearly varying frequency. The actual ground acceleration time history is measured by the accelerometer and collected by the central server, as plotted in Figure 7. The measurement shows that because of the limited performance of the shake-table, the actual ground acceleration is noisy compared with an ideal chirping signal.

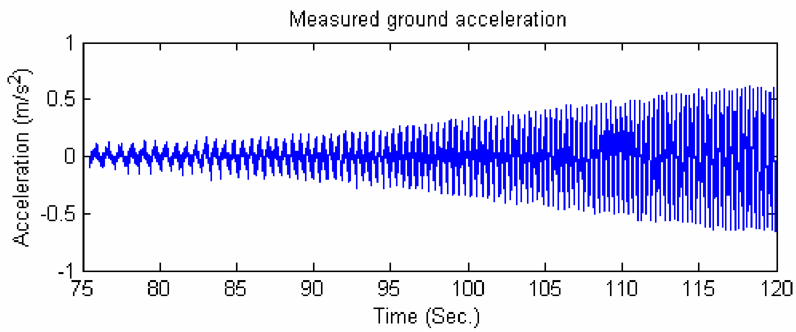


Figure 7 – Ground acceleration time history

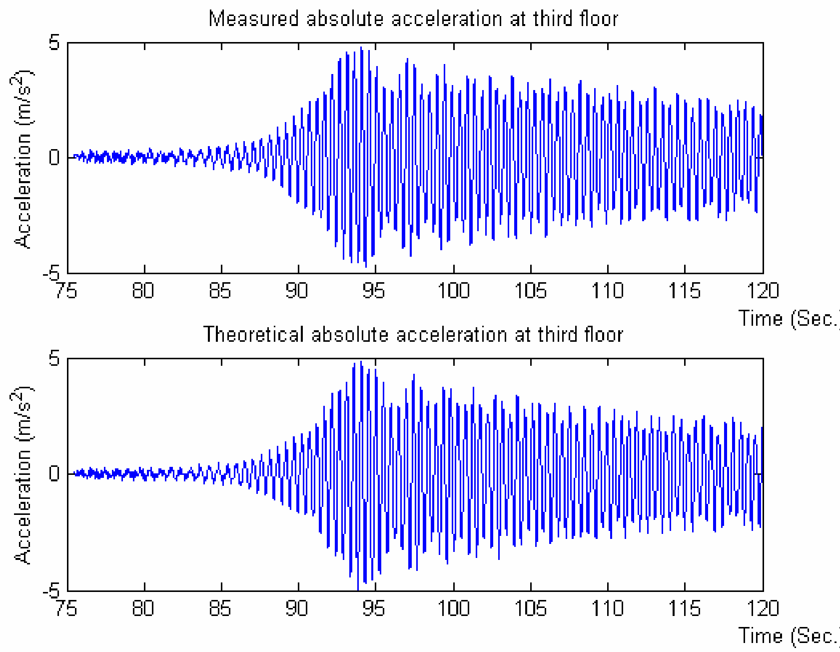


Figure 8 – Comparison of measured and theoretical absolute acceleration at third floor

Using the measured ground acceleration as input, numerical simulation is executed to compute the theoretical movement of all the three floors using average acceleration method. Comparison between the measured and theoretical absolute acceleration time history at the third floor is shown in Figure 8. The measured and the theoretical time history plots are similar in both shape and magnitude, which demonstrates the reliability of the wireless monitoring system.

	1st floor	2nd floor	3rd floor
Measured (m/s^2)	2.14	3.35	4.74
Theoretical (m/s^2)	2.07	3.76	4.93
Relative Difference	3.32%	11.5%	3.93%

Table 1 – Comparison of measured and theoretical maximum absolute acceleration

For simplicity, comparisons for the time history data at the other two floors are not plotted, but the measured and theoretical maximum absolute acceleration at each floor is presented in Table 1. The difference at the second floor is slightly larger than the difference at the first and third floor, and this is probably resulted from the difference in the performance of the two types of accelerometers. Nevertheless, the overall comparison appears to be reasonable.

CONCLUSIONS

This paper presents the integrated hardware and software design of a wireless structural monitoring system. The hardware design reflects the special demands for applying wireless sensing units and systems in structural engineering, which is to balance the conflict between the requirement for low power consumption and the requirement for long-distance communication and sufficient computational power. The combination of the selected wireless modem and microcontroller, two of the major parts of a wireless sensing unit, serves these special requirements. To ensure high fidelity and flexibility in data collection, a multi-channel high-resolution analog-to-digital converter is included within each sensing unit.

Software design of the system explores the state machine concepts to implement a robust communication network between a central data server and multiple wireless sensing units. As illustrated, state machine concepts can be used to clarify the complicated software design resulted from the inevitable unreliability of wireless communication. The software design is oriented to handle wireless communication failures at various communication stages, and fully recover the system whenever any communication failure happens. To satisfy the data synchronization requirement for dynamic data analysis, a communication protocol is designed and implemented for near-synchronized real-time data acquisition from multiple wireless sensing units associated with multiple heterogeneous sensors.

The first validation test shows that the theoretical structural natural frequencies are very close to those values extracted from the data collected by the wireless monitoring system. The second validation test shows that the measured acceleration time history at each floor of the test structure also matches with its theoretical result relatively well. The reliability of the wireless structural monitoring system is exemplified by these validation tests.

Further research is need on distributing and incorporating the advanced data analysis or damage identification algorithm into the grid of the wireless sensor network, to organize the distributed computational sources for high-level engineering analyses. The other area of interest is to improve the scalability of the wireless monitoring network, making the system capable of large-scale data collection from densely allocated sensors. On the hardware side, the highly dynamic embedded system market is offering new options for new component adoption. In particular, greater communication range, higher data rate and low power demand should be sought in future wireless modems. In addition, better solutions could be explored in improving the precision of analog-to-digital data conversion against the circuit noise.

ACKNOWLEDGEMENTS

This research is partially funded by the National Science Foundation under grants CMS-9988909 and CMS-0121842. The first author is supported by a Stanford Graduate Fellowship.

REFERENCES

[1] Liu, S. C., and Tomizuka, M., “Vision and Strategy for Sensors and Smart Structures Technology Research,” *Proceedings of the 4th International Workshop on Structural Health Monitoring*, Stanford, CA, pp. 42-52 (2003).

[2] Stallings, J. M., Tedesco, J. W., El-Mihilmy, M., McCauley, M., “Field Performance of FRP Bridge Repairs,” *Journal of Bridge Engineering*, Vol. 5, No. 2, pp. 107-113. (2000).

[3] Sohn, H., Farrar, C. R., Hemez, F. M., Shunk, D. D., Stinemates, D. W., and Nadler, B. R., *A Review of Structural Health Monitoring Literature: 1996-2001*, Los Alamos National Laboratory Report, LA-13976-MS (2003).

[4] Lynch, J. P., Sundararajan, A., Law, K. H., Kiremidjian, A. S., Kenny, T., and Carryer, E., “Embedment of Structural Monitoring Algorithms in a Wireless Sensing Unit,” *Structural Engineering and Mechanics*, Vol. 15, No. 3, pp. 285-297 (2003).

[5] Celebi, M., *Seismic Instrumentation of Buildings (with Emphasis on Federal Buildings)*, Report No. 0-7460-68170 United States Geological Survey (USGS), Menlo Park, CA (2002).

[6] Straser, E. G., and Kiremidjian, A. S., *A Modular, Wireless Damage Monitoring System for Structures*, Report No. 128, John A. Blume Earthquake Engineering Center, Department of Civil and Environmental Engineering, Stanford University, Stanford, CA (1998).

[7] Lynch, J. P., Sundararajan, A., Law, K. H., Kiremidjian, A. S., and Carryer, E., "Embedding Damage Detection Algorithms in a Wireless Sensing Unit for Attainment of Operational Power Efficiency," *Smart Materials and Structures*, IOP, Vol. 13, No. 4, pp. 800-810 (2004).

[8] Hill, J. L., *System Architecture for Wireless Sensor Networks*, Ph.D. Thesis, Department of Computer Science, University of California, Berkeley, CA (2003).

[9] Culler, D. E., and Hong, W., “Wireless Sensor Networks: Introduction,” *Communications of the ACM*, Vol. 47, No. 6, pp. 30-33 (2004).

[10] Callaway, E. H. Jr., *Wireless Sensor Networks: Architectures and Protocols*, Auerbach publications (2004).

[11] Tweed, D., “Designing Real-time Embedded Software Using State-machine Concepts,” *Circuit Cellar Ink*, Issue #53, pp. 12-19 (1994).