

Development of an Extensible Dual-Core Wireless Sensing Node for Cyber-Physical Systems

Michael Kane^a, Dapeng Zhu^b, Mitsuhiro Hirose^a, Xinjun Dong^b, Benjamin Winter^c,
Mortiz Häckel^d, Jerome P. Lynch^{*a}, Yang Wang^b, Andrew Swartz^c

^aDept. of Civil and Environ. Eng., University of Michigan, Ann Arbor, MI, USA 48109

^bSchool of Civil and Environ. Eng., Georgia Inst. of Technology, Atlanta, GA, USA 30332

^cDept. of Civil and Env. Eng., Michigan Technological University, Houghton, MI, USA 49931

^dInstitute of Structural Analysis, Leibniz Universität Hannover, Hannover, Germany

ABSTRACT

The introduction of wireless telemetry into the design of monitoring and control systems has been shown to reduce system costs while simplifying installations. To date, wireless nodes proposed for sensing and actuation in cyber-physical systems have been designed using microcontrollers with one computational pipeline (*i.e.*, single-core microcontrollers). While concurrent code execution can be implemented on single-core microcontrollers, concurrency is emulated by splitting the pipeline's resources to support multiple threads of code execution. For many applications, this approach to multi-threading is acceptable in terms of speed and function. However, some applications such as feedback controls demand deterministic timing of code execution and maximum computational throughput. For these applications, the adoption of multi-core processor architectures represents one effective solution. Multi-core microcontrollers have multiple computational pipelines that can execute embedded code in parallel and can be interrupted independent of one another. In this study, a new wireless platform named *Martlet* is introduced with a dual-core microcontroller adopted in its design. The dual-core microcontroller design allows *Martlet* to dedicate one core to standard wireless sensor operations while the other core is reserved for embedded data processing and real-time feedback control law execution. Another distinct feature of *Martlet* is a standardized hardware interface that allows specialized daughter boards (termed *wing boards*) to be interfaced to the *Martlet* baseboard. This extensibility opens opportunity to encapsulate specialized sensing and actuation functions in a *wing board* without altering the design of *Martlet*. In addition to describing the design of *Martlet*, a few example *wings* are detailed, along with experiments showing the *Martlet*'s ability to monitor and control physical systems such as wind turbines and buildings.

Keywords: wireless monitoring, wireless control, dual-core microprocessor, cyber-physical systems

1. INTRODUCTION

Civil infrastructure systems are vital assets that fundamentally ensure a high quality of life for a society. Unfortunately, the owners and managers of infrastructure systems in many developed nations are confronting a number of challenges including the combination of growing inventories of aging assets and increasing user demand originating from growing populations. These challenges are as complex as they are significant and require innovative solutions. Fortunately, the emergence of low-cost sensors, mobile computing, wireless networking, and the Internet are creating new technologies that are profoundly changing how engineers approach the management of critical infrastructure systems. There is a long tradition of civil engineers embedding sensors, actuators and computing elements into their infrastructure for monitoring and controlling their performance. In fact, these intelligent infrastructure systems are a prototype example of the recently defined *cyber-physical systems* (CPS) field [1]. Cyber-physical systems are systems that synergistically combine sensing, actuation, and computing with a physical system. Low-power wireless networking has been one of the driving technologies that have accelerated the spread of CPS examples in various engineering domains including in civil engineering.

In this study, the authors propose a new wireless sensor node for adoption in CPS frameworks. While many wireless sensors have been proposed for monitoring and controlling civil infrastructure systems, the node proposed

* e-mail: jerlynch@umich.edu phone: 1-734-615-5290 web: www.umich.edu/~jerlynch



Figure 1. The *Martlet* wireless node baseboard.

in this paper is specifically designed to emphasize two critical features for a CPS framework: high-speed computing and real-time deterministic functionality. The new device is named *Martlet*¹ and is shown in Figure 1.

The *Narada* wireless sensing unit [5] was a precursor to the *Martlet* and is a very successful platform for monitoring operational civil [6,7] and non-civil engineering structures [8]. The *Narada* platform has also been shown to be capable of distributed data processing across the wireless sensor network for system identification [9,10], controlling the response of buildings subject to earthquakes [11–15], and controlling industrial plants [16]. *Narada*'s strengths include a relatively simple set of source-code, ad-hoc communication capabilities, high-resolution data acquisition, long communication ranges, and a simple interface for sensors and actuators to freely attach to. *Narada* also has a number of drawbacks. Similar to other wireless sensors designed around an 8-bit fixed point microcontroller, *Narada* suffers from inflexible power management schemes and slow floating-point calculations emulated its fixed-point microcontroller. The design of the *Martlet* was undertaken with the strengths and weaknesses of the *Narada* node kept in mind. The development of *Martlet* has been based on an active collaboration between the Laboratory for Intelligent Systems and Technologies (LIST) at the University of Michigan (Ann Arbor, MI), the Laboratory for Smart Structural Systems at Georgia Institute of Technology (Atlanta, GA), and the Department of Civil and Environmental Engineering at Michigan Technological University (Houghton, MI). The primary goal of the *Martlet* development effort was to create a new wireless device that featured:

- wireless communications framework backwards compatible with the *Narada* architecture;
- a faster processor capable of true floating point calculations executed in hardware;
- a processor that could execute multiple threads of embedded code with true parallelism;
- a modular design with simple connections to common sensors, actuators, and peripherals;
- JTAG debugging capability that would reduce development time for new applications;
- and suitable memory for data storage as well as for the execution of complex algorithms.

The design of such a device is documented herein.

This paper begins with a description of the *Martlet* hardware and software design. A few custom designed peripherals are then presented to show the variety of applications that the *Martlet* can be used for and the ease with which the *Martlet* can be adopted in these applications. Finally, field trials and laboratory experiments are used to validate the performance of the *Martlet* when applied to structural monitoring and feedback control applications.

¹ The wireless device was named after the *Martlet* heraldic charge, depicted as a small fast bird which symbolized the fourth son, virtue, and adventure [2]. The wireless device contains a fast 80 MHz processor, is capable of learning and actuation, and sits as the 4th prototype in line starting with Straser and Kiremidjian's WiMMS device in 1998 [3], the 2nd generation WiMMS devices in 2001 [4], and the *Narada* in 2005 [5].

Table 1. Comparison of Wireless Communication Standards

	IEEE 802.11	IEEE 802.15.1	IEEE 802.15.4	WiMax	Cellular/3G
Data-rate	Video	Audio	Text / Images	Video	Video
Range	Room / Building	Body / Room	Body / Building	Campus	City
Battery Life	Minutes/Hours	Hours/Days	Days/Years	Minutes/Hours	Minutes/Hours
Cost and complexity	High	Medium	Low	High	High

Adapted from [19] and [18]

2. WIRELESS COMMUNICATION FOR MARTLET

Wireless sensors include digital radios in their design because digital radios are low-cost and power efficient while providing a scalable approach to the shared use of a common radio frequency (RF) band. Digital radios require a microcontroller to packetized data prior to transmission (it should be noted that the microcontroller is often used to perform other functions for the wireless sensor node including computing [17]). In the design of the *Martlet* node, a digital radio with a suitable wireless communication standard must be selected. In the selection process, emphasis will be placed on balancing network communication speed, message latency, and power efficiency. Selection of an appropriate radio is greatly simplified through the use of standard communication specifications such as IEEE 802.11 (*e.g.* WiFi), IEEE 802.15.4 (*e.g.* ZigBee and WirelessHART), IEEE 802.15.1 (*e.g.* Bluetooth), and cellular data systems (*e.g.* WiMax, 3G) [18]. Devices using the IEEE 802.x standards operate on the industrial, scientific, and medical (ISM) radio bands (900 MHz, 2.4 GHz, and 5.8 GHz). In most nations, federal authorities allow low-power wireless communication on the ISM bands without requiring a license for RF spectrum use. The ISM bands are most commonly used for low-power wireless devices that need to transmit data on the order of 100 m or so. Table 1 compares the wireless telemetry standards on the basis of data-rate, range, power consumption, and complexity. Based on the requirements established during the design of the *Martlet*, the IEEE 802.15.4 standard was selected. This choice gives the *Martlet* conservative power requirements and the ability to stream sensor data at rates on the order of 1 kHz over distances expected between nodes in civil infrastructure. Another motivator for selecting IEEE 802.15.4 is that many other wireless sensor platforms operate using IEEE 802.15.4, thereby giving *Martlet* the ability to communicate freely with other wireless sensors using the same standard.

The software on each node that enables the wireless network is called the *network stack* and consists of the following seven layers: physical, data-link, network, transport, session, presentation, and application. The network stack used by *Martlet* is described in brief. The first two network layers (physical and data-link) are defined in the IEEE 802.15.4 standard [20]. This incorporates a 2.4 GHz radio that digitally transmits packetized data. The network layer is dedicated to facilitating ad-hoc, node-to-node and broadcast communication schemes. If a destination node is out of RF range from the source node, then a strategy of hopping messages over intermediate nodes, called multi-hopping, would be included in the application layer. The transport layer employs: 1) a clear channel assessment (CCA) when sending packets in order to prevent packet collisions, and 2) acknowledgements to help the sender ensure that the destination received the packet. The session layer defines that transmissions must be sent in a single session. The presentation layer collects the raw data from the radio and presents a byte-stream of the packet's payload data and header to the application layer which is custom designed for each application. A customized network stack will be implemented on top of the IEEE 802.15.4 physical and data link layers for *Martlet*. This approach is in lieu of selecting more standardized IEEE 802.15.4-based network stacks like ZigBee [21], WirelessHART [22] or those associated with TinyOS [23] which are either too complex or inflexible for many of the envisioned applications for *Martlet*. The *Martlet's* network stack is intentionally designed to be backwards compatible with the *Narada* network stack.

The *Martlet's* radio module is formed by pairing a TI CC2520 2.4 GHz IEEE 802.15.4 transceiver with a TI CC2591 RF front-end; the CC2591 is a programmable low noise amplifier (LNA) for improved receiver sensitivity and power amplifier (PA) for amplification of the radio transmission power. In particular, the PA allows the communication range of the IEEE 802.15.4 transceiver to extend beyond 500 m line-of-sight. The radio module's power consumption can be varied by adjusting the output power of the module. For example, transmission consumes 55mA (references at 3.3V) when the transmission power is set to -8 dBm and 136mA when set to +17 dBm. In a similar fashion, the reception consumes 23 mA when the receiver sensitivity is set to -50 dBm and 26mA

when set at -90 dBm. When placed in sleep mode, the radio draws as little as 1 μ A. While these numbers seem larger than radios that are not power amplified, the communication range of the radio is greatly enhanced. Consuming more power at the radio to achieve greater communication range can actually result in lower overall power consumption of the network due to enhanced communication reliability.

3. CORE HARDWARE DESIGN

The CPS paradigm is based around a sense-compute-actuate framework in which actuation is expensive, sensing is cheap, and computing lies in-between [24]. In the civil engineering domain, successful field deployments have validated the effectiveness of wireless sensing [25,26]. In addition, laboratory tests have shown the possibility of wireless feedback control to improve structural performance under extreme load events like earthquakes [13,27]. However, full system autonomy and interaction with a physical system through actuators has been limited because the real-time requirements for feedback control are currently too stringent for most low-cost wireless sensor platforms to undertake. The *Martlet* aims to overcome this limitation.

For the *Martlet* to excel in monitoring and controlling infrastructure systems, it must have low power consumption, low latency, and the ability to quickly process data and execute control algorithms. Striking a balance between the low-power, yet insufficient speed, of 8-bit microcontroller units (MCUs) and the high power consumption of 32-bit MCUs, the *Martlet* was designed around a 16-bit Texas Instruments (TI) TMS320F28069 modified Harvard Bus Architecture MCU with a dual-core design and an on-the-fly programmable clock that can run up to 80 MHz. Real-time digital signal processing is possible with native single-precision floating-point calculations. The Viterbi complex math unit (VCU) extends the instruction set to support complex multiplication, Viterbi operations, and cyclic redundancy checks (CRCs). The computationally intensive tasks of control processing are offloaded from the main CPU and onto a programmable control law accelerator (CLA) core with a dedicated pipeline whose 32-bit floating-point math accelerator offers precise timing. A 9-channel dual sample-and-hold 12-bit ADC is capable of sampling analog signals at rates up to 3 MHz. Programs residing in the 256 kB x 16-bit flash memory can process this data, or the 6-channel direct memory access (DMA) controller can directly feed the ADC data into the 100 kB x 16-bit random access memory (RAM) for further processing by the CLA. If additional data storage capacity is required, a microSDHC card can be inserted into the on-board card reader giving the *Martlet* up to 32 GB of additional flash memory. A variety of general purpose input/output (GPIO) pins are available to communicate with peripherals using protocols such as UART, SPI, I²C, pulse width modulation (PWM), and timed event capture. The fast interrupt response inherent in the architecture of the MCU, together with three 32-bit timers, makes the TMS320F28069 an excellent choice for control applications with strict timing requirements. Inspiration for the new capabilities featured on the *Martlet* come from years of experience with the design and use of wireless devices for civil infrastructure applications. The *Martlet* printed circuit board (PCB) features 16 different test points that give developers access to all intra-board signals to simplify debugging of applications and peripherals. The mechanical stresses and vibrations experienced in the field adversely affect the reliability of some wireless devices currently available on the market. To this end, the *Martlet* securely joins peripheral layers, called *Martlet wings*, with threaded plastic standoffs. The use of edge mounted SMA antenna connectors soldered to both sides of the PCB and the use of through-hole posts on inter-board connectors mitigates the possibility of premature connector failures. Sleep or power enable pins have become a standard feature of many integrated circuits (ICs) likely to be used on *wings*. In order to take advantage of this feature for reducing power consumption, the *Martlet* exposes 12 of the GPIO pins through the top and bottom *wing* connectors.

4. SOFTWARE DEVELOPMENT

The development of the *Martlet* required significant collaborations between all parties involved. The team took advantage of a Git version control system [28] to manage software versioning, leading to over 600 contributions by over 10 authors. Additionally, a wiki was used to track hardware development and for project management. *Martlet* was designed with a number of features that are intended to lessen the burden on the application developer include the use of industry standard programming languages, a simple state machine, and JTAG debugging with hardware breakpoints and memory inspection. Usability features currently in development include automatic detection of peripheral boards, automatic calibration, and foolproof transducer connectors.

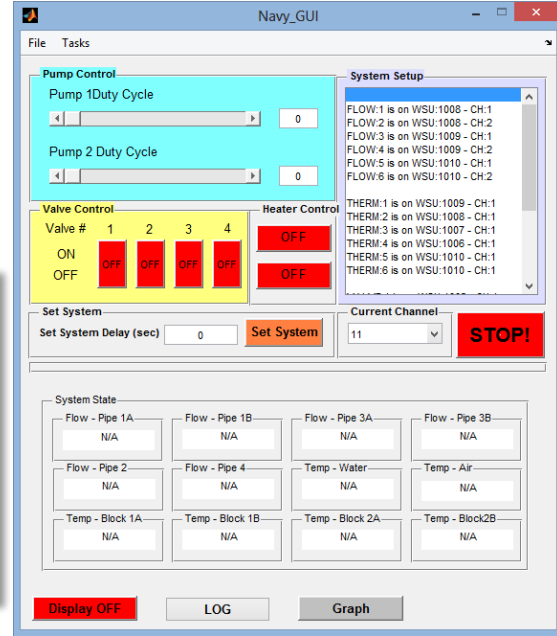
A state machine serves as the *Martlet*'s operating system (OS). After initialization, the *Martlet* sits idle until an interrupt occurs internally (*e.g.*, from a timer, analog-to-digital converter (ADC) channel event) or externally (*e.g.*, from a GPIO pin or the microcontroller UART). Over 70 different interrupts sources are mapped using the peripheral interrupt expansion (PIE) vector pointing to interrupt service routines (ISR) that execute the desired code

```

*****
* Welcome to the Martlet network command-line user-interface *
*                               *
*      Civionics, LLC          *
*    University of Michigan    *
*                               *
*      Version 3.1             *
*    Copyright 2013            *
*                               *
*****
State: SERVER_READY
Please choose an option from the list:
00: Exit Program
01: Check for current status of one wireless node
02: Check for current status of entire wireless network
03: Change radio settings on one unit
07: Change RF channel
08: Reboot a unit remotely
09: Change server-side receiver settings
10: Collect sensor data from network
14: Recollect DAQ data from network
>>

```

(a)



(b)

Figure 2. Martlet user interfaces: (a) command line interface; (b) hydronic system graphical user interface.

after an interrupt event. A lightweight state machine-based OS was chosen for two reasons: to enable rapid ($\sim 0.1 \mu\text{s}$) switching from main program code into an ISR and because simple multi-threaded applications (< 5 threads) can be quickly developed and understood by novice developers. Time-critical code threads can be implemented on the CLA core. This allows true real-time threads on the CLA to run in parallel to non-real-time threads on the other core.

Lying between the OS and the application software are middleware services. These provide the application software with standard node services such as fixed-rate data collection, network wide clock synchronization, data-request methods, and network discovery and routing, among other services. Slight variations in the crystal on each *Martlet* necessitate a method of synchronizing the network on a regular basis to ensure accurate timing between nodes. The current method uses a gateway node to send out a synchronization beacon to which all *Martlets* synchronize; this results in time synchronization of less than $30 \mu\text{s}$. In ad-hoc control networks, nodes often desire data from peers when local events occur. The data-request middleware offers the application layer a method for reliably collecting large data sets from multiple peers without blocking program execution while waiting for responses. Often, when large amounts of data are processed on the nodes, a need arises to store the data for archival or debugging purposes. Toward this end, a lightweight FAT file system (Petit FAT File System [29]) was implemented. The FAT file system enables data to be stored into text or binary files on a microSD card that can then be removed from the *Martlet* and inserted into a personal computer (PC) for processing or archiving. Middleware for multi-hop message routing and clock synchronization for non-star network configurations are currently in development.

Application software can be written to execute algorithms for control, data compression, or system identification. For *Martlet* networks not running autonomously, server-side applications have also been developed for commanding the network. A general command line interface (CLI), shown in Figure 2a, has been written for users. Also, an array of application specific graphical user interfaces (GUI) have been written such as the one shown in Figure 2b for controlling hydronic pipe networks.

5. MARTLET COMPARED TO OTHER WIRELESS SENSORS

Many options are available to designers who wish to implement a wireless CPS. Commercial off-the-shelf (COTS) solutions are available for sensing from companies such as Microstrain [30], Bridge Diagnostics Inc. [31], and National Instruments [32]. Only a few products are available for control applications, mainly WirelessHART

Table 2. Comparison of popular wireless modules for research and development in civil engineering

	<i>Martlet</i>	<i>Narada</i> [38]	<i>iMote2</i> + SHMA [39]
Processor bus width	16-bit	8-bit	32-bit
Processor speed	10-80 MHz + CLA	8 MHz	13-416 MHz
Non-volatile memory	256k x 16-bit +32GB SDcard	128k x 8-bit	32M x 8-bit
Volatile memory	100k x 16-bit	128k x 8-bit	32M x 8-bit
Active sensing power	~450 mW	~300 mW	~400 mW
Sleep power	<1 mW	~60 mW	<1 mW
Size (mm) approx.	60 x 60	70 x 70	40 x 50

products from companies like Siemens [33]. These commercial solutions are easy to implement but are often inflexible for emerging applications or research purposes. More flexible development modules which can be customized and developed into an application specific product are available from such companies as Memsic Inc. with their ‘mote’ line of wireless modules. Furthermore, academic prototype modules have been developed for research applications outside the purview of commercial modules [5,34–37].

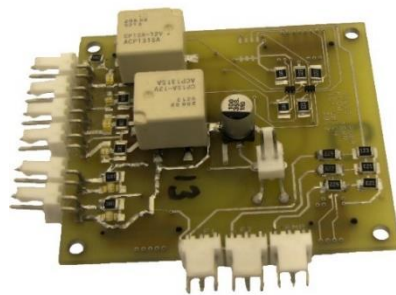
The *Martlet* fits into that last category of modules designed for research in emerging applications. It distinguishes itself from similar platforms by its dual core architecture, native floating-point calculations, sturdy extensible design, and easy-to-use development environment for engineers familiar with embedded code development. Table 2 shows a comparison of the *Martlet* with two other popular nodes used in civil engineering research: the *Narada* and iMote2 with the SHMA acceleration sensing board. The chart shows that the *Martlet* falls in the middle of the pack with respect to processing capability and active power, but the *Martlet* has more capable sleep modes than then *Narada* and is easier to program than the iMote2 which uses a non-real-time OS (TinyOS) and obscure programming language (nesC).

6. DEVELOPMENT OF PERIPHERAL WING BOARDS

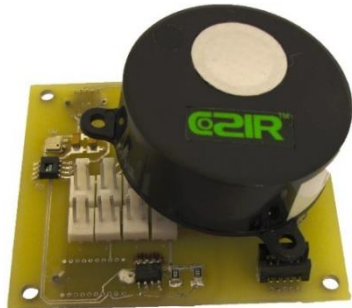
The key to the *Martlet*’s success in enabling CPS research will be its ability to sense and control a multitude of different systems and physical properties. Initial applications for the *Martlet* include vibration analysis of wind turbine structures, semi-active structural control, hydronic system monitoring and control, and HVAC monitoring and control. The *Martlet* is able to sense and control these systems because of its extensible design. *Martlet wings* have already been created for a wide variety of purposes: data collection from strain transducers and accelerometers (Figure 3a); ultrasonic sensing; fluid flow measurement (Figure 3b); pump and valve control (Figure 3b); CO₂ and temperature monitoring (Figure 3c); motor control (Figure 3d); programming and debugging (Figure 3e); prototyping (Figure 3f); band-pass filtering of signals with high dynamic range by means of programmable filter gains and cut-off frequencies (Figure 3g); actuating magneto-rheological (MR) dampers (Figure 3g); accurate time keeping with a real-time clock (RTC) (Figure 3h). These *wings* have been designed relatively quickly using standard signal conditioning and interface circuitry found in texts such as [40] and laid out on standard *wing* templates. The potential to utilize the *Martlet* for different CPS applications is nearly limitless due to the flexibility of the *wings* to communicate with the *Martlet* core via analog signals that map directly to the *Martlet*’s ADC or via digital protocols including SCI, I²C, UART, or GPIO. Two of these *wings* will be described in detail in the following sections.



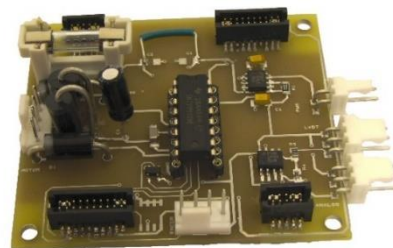
(a)



(b)



(c)



(d)



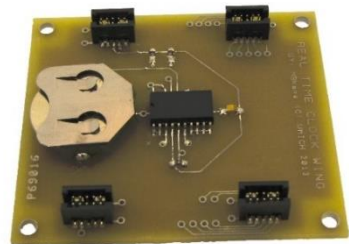
(e)



(f)



(g)



(h)

Figure 3. *Martlet* peripheral wings: (a) strain and acceleration *wing*, (b) hydronics *wing*, (c) environmental sensing *wing*, (d) motor control *wing*, (e) programming and debugging *wing*, (f) breadboard *wing*, (g) smart filter and DAC *wing*, (h) real-time clock *wing*.

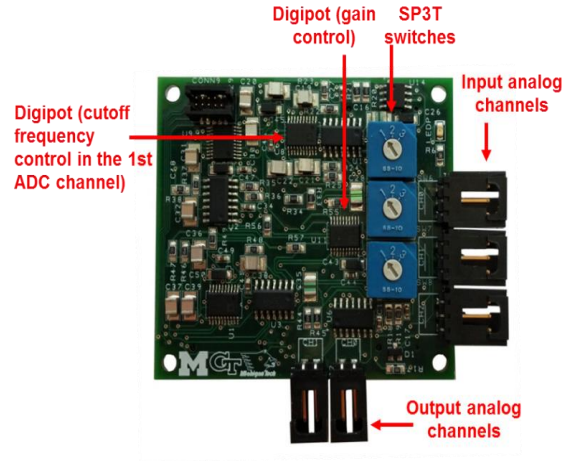


Figure 4. Smart ADC/DAC wing with components highlighted (area is 36 cm²).

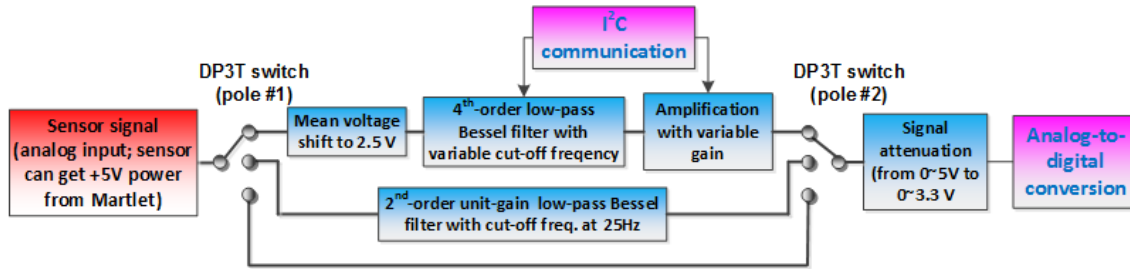


Figure 5. Functional diagram for an analog input channel.

6.1 Smart ADC/DAC Wing

The sense-compute-actuate framework of CPS requires the ability to convert real-world measurands into digital signals followed by the conversion of the results into real-world actions (*i.e.*, actuation). This is typically accomplished with sensing transducers that output analog electrical signals correlated to the physical stimulus being measured and actuators that are command by analog voltage signals. Sensors and actuators are typically connected to analog-to-digital converters (ADC) and digital-to-analog converters (DAC), respectively. To this end, the *Martlet Smart ADC/DAC wing* shown in Figure 4 was developed as a general purpose interface for sensing and actuation. The three sensing channels accept any commonly used transducer that outputs a 0 to 5V signal and requires a 5V power source. Each channel contains three possible signal conditioning paths as schematically shown in Figure 5; the paths are selectable with a double-pole three-throw (DP3T) mechanical switch. The first path offers the most sophisticated signal conditioning including mean shift, on-the-fly programmable low-pass filtering, and on-the-fly programmable amplification. The mean-shift ensures that the maximum dynamic range is possible by centering the dynamics of the signal on the center of the ADC range (*i.e.*, 2.5 V). The programmable low-pass filter (with cut-off frequencies from 15 Hz to a few hundred Hz) maximizes the spectrum of the captured signal while ensuring proper anti-aliasing. The programmable gain (with settings from 1.9x to 190x) can be used to amplify the signal so that the dynamic range of the amplified signal crosses the greatest number of bits in the ADC, thus increasing the resolution of the captured signal. An I²C signal from the *Martlet* is used to adjust factory calibrated digitally controllable potentiometers to accurately adjust the gain and cutoff frequencies. The second path on the DP3T switch has a fixed gain of 1x and an anti-aliasing filter with a cutoff frequency of 25 Hz. This path is most useful when simplicity is desired and the fixed-value gain and cutoff are appropriate of the application at hand. The third path passes the signal directly through to the *Martlet* ADC without amplification or filtering. This setting is used in conjunction with external signal conditioning. After the signal passes through one of the three paths it is scaled down from a maximum range of 5V to a maximum range of 3.3V. This is necessary so that the signal matches the input range of the *Martlet*'s 12-bit internal ADC.

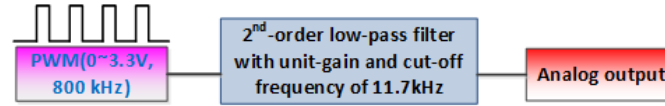


Figure 6. Functional diagram for an analog output channel.

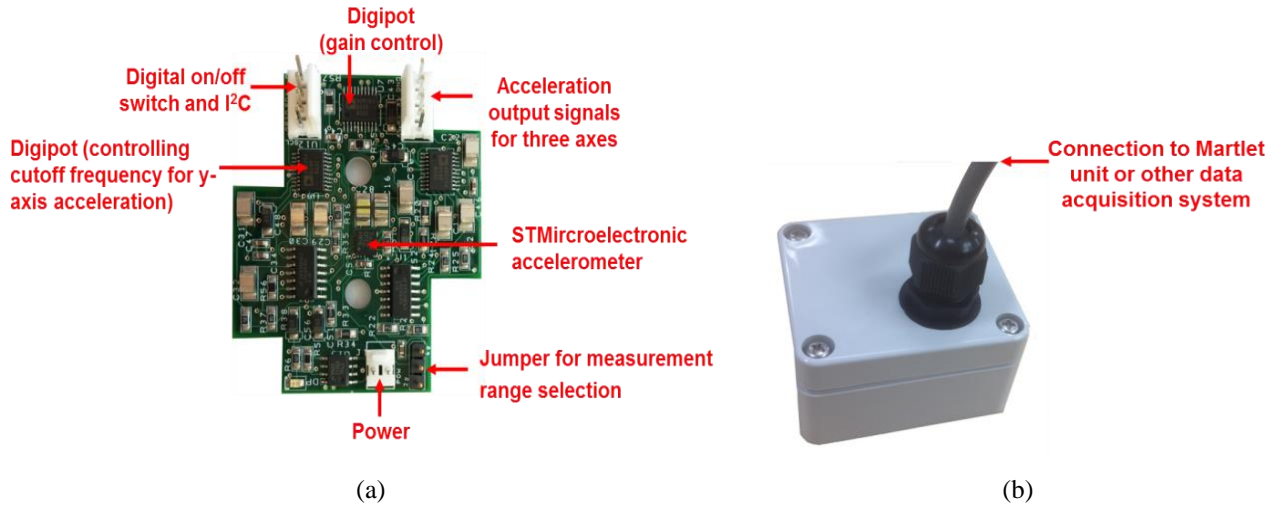


Figure 7. Integrated accelerometer board: (a) main PCB; (b) weather resistant package.

To generate analog output signals, the ADC/DAC wing board contains two independent analog output channels. The functional diagram of each channel is shown in Figure 6. The DAC process begins with the *Martlet*'s internal pulse width modulator (PWM) that is used to output a square wave to a GPIO. The output of the GPIO is then filtered to generate an analog signal. A 2nd-order low-pass filter with unity gain and a cutoff frequency of 11.7kHz is used to remove the high-frequency components of the PWM signal, leaving only the low-frequency content as the analog output. As a result, the output voltage level is determined by the duty cycle of the PWM signal. For example, a 100% duty cycle PWM signal generates a 3.3V voltage while a 50% duty cycle PWM signal generates 1.65V. Therefore, the voltage generation capability of the DAC is achieved by simply adjusting the duty cycle of the PWM.

6.2 Integrated Accelerometer Wing

To create a *Martlet*-based sensing node for acceleration measurement, a *wing* board was created by integrating signal conditioning and an accelerometer into a single PCB as shown in Figure 7a. The integrated accelerometer board uses a tri-axial microelectromechanical systems (MEMS) accelerometer (ST Microelectronics LIS344ALH). A jumper on the PCB selects either a $\pm 2g$ or $\pm 6g$ measurement scale. The noise density of the measurement is $25 \mu g/\sqrt{Hz}$ along the x-axis and y-axis, and $50 \mu g/\sqrt{Hz}$ along the z-axis. The voltage signals from the accelerometer are conditioned using the same circuitry used in the first path of the *Martlet Smart ADC/DAC wing* from Figure 5. The mean-shifted adjustable filtering and gain enable the integrated accelerometer board to be used in any orientation and signal amplitude while allowing it to maintain an ideal dynamic range and spectrum of the sampled signal. The filter and gain settings are stored in nonvolatile memory on the PCB.

The integrated accelerometer board is placed in a compact weatherproof enclosure with a dimension of 60 mm x 64 mm x 35 mm as shown in Figure 7b. In this case, the wing is not actually attached to the *Martlet* base. Rather, the integrated accelerometer board is connected to the *Martlet* with an eight-wire cable. Three wires in the cable are allocated for the acceleration output signals (X, Y and Z channels), two are allocated for I²C communications, one is allocated for power, one is allocated for ground, and the last one is allocated to a digital signal that allows the

Martlet base board to power the *wing* on and off. The current draw of the integrated accelerometer board is ~ 12 mA (referenced at 3.3 V) under normal working conditions and $\sim 1\mu\text{A}$ when switched off.

7. MARTLET VALIDATION: STRUCTURAL VIBRATION MONITORING

The development process of any new wireless node should include phases of thorough testing. The *Martlet*'s testing began before the first PCB was ever made. The radio was tested for reliability, strength, and sensitivity; the MCU was tested for computational speed and dual-core parallel operations; and the ADC was tested for accuracy, speed, and resolution. However, no amount of laboratory testing can guarantee the device will work in real-world conditions. As a result, an experimental program was created to field test the monitoring capabilities of *Martlet* using a wind turbine at Los Alamos National Laboratory in New Mexico, USA.

7.1 Goals and motivation

Researchers at the University of Michigan, Leibniz Universität Hannover, and Los Alamos National Laboratory (LANL) were interested in investigating the long-term and transient effects of environmental and operational conditions (EOC) on the structural response of wind turbines towers. A Whisper 500 wind turbine tower [41] depicted in Figure 8 and sited on the campus of LANL (located at $35^{\circ}48'5''\text{N}$ $106^{\circ}17'47.3''\text{W}$) was instrumented. The aim of the instrumentation was to identify key environmental and structural conditions that would affect the structural response of the tower. The turbine tower is 12.3 m tall and constructed of steel. Atop the tower is a 4.5 m diameter Whisper 500 wind turbine. The structure was specially designed as a test bed for wind turbine research (such as that conducted by [42]) and features a pivot point in the middle of the structure that allows the turbine nacelle to be lowered to the ground for maintenance purposes and sensor installation. The following three goals were established for the field experiment:

- 1) demonstrate the multi-day reliability of the *Martlet* DAQ application;
- 2) identify key environmental conditions that affect structural behavior;
- 3) determine other measurands besides vibrations and EOC that affect structural response.

7.2 Instrumentation Strategy

The investigation on the EOC impact on wind turbine structural behavior required weather data (*e.g.*, temperature, humidity, wind speed, and wind direction) to classify EOCs and tower vibration data to classify the structural response. Local weather information was provided in 15 minute intervals by the LANL 'Weather Machine' [43] measured at a metrological tower located 267 m from the wind turbine at a bearing of -73° . This campaign measured the structural response of the turbine tower with an array of Crossbow CXL02TG3 tactile-grade tri-axial accelerometers with a range of ± 2 g, low noise floor, and integral temperature sensing. A newly designed *Martlet wing*, previously shown in Figure 3a, was used to interface the accelerometer to the *Martlet*'s ADC using a 250 Hz low pass filter. Although the modal frequencies of the tower are expected to be well below 10 Hz, this high cut-off frequency was chosen in order to ensure that the high-frequency harmonics (the blades spin at a nominal rate of 500 rotations per minute in wind speeds from 3.4 m/s (7.5 mph) to 55 m/s (120 mph)) of the rotating blade would be captured. The *Martlet*, tri-axial accelerometer, signal conditioning *wing*, and power regulation *wing* were all mounted within a $21\text{ cm} \times 12.7\text{ cm} \times 11\text{ cm}$ ($8.25 \times 5.00 \times 4.33''$) enclosure. The pivoting mechanism of the tower allowed three wireless accelerometers to be placed on the top half of the tower, and three on the bottom half of the tower as shown in Figure 8.

The WSN installation was constrained by the security and safety requirements of the LANL site. These requirements prevented 24 hour access to the site and prohibited wireless communications to off-site. Typically, long-term WSN deployments use a cellular modem to relay data collected from the WSN to off-site servers [7]. Instead, this installation used a PC/104 single board computer (SBC) with a 16 GB flash drive running a Linux operating system to serve as a local base station. The large hard drive would offer plenty of room for data storage.

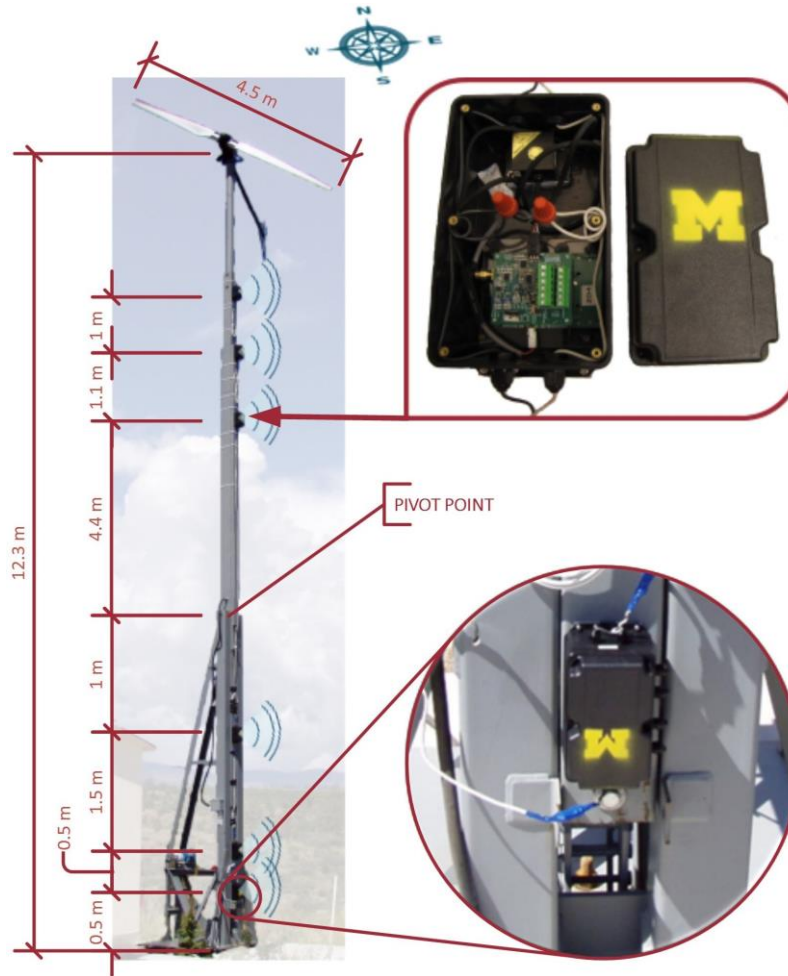


Figure 8. LANL Whisper 500 wind turbine and *Martlet* installation.

Connected to the SBC via USB was an IEEE 802.15.4 compatible radio with an external high-gain antenna. The base station was placed in an enclosure along with a backup power supply unit (PSU) connected to the 120 AC voltage generated by the wind turbine. The PSU supplied 5 V to the SBC and externally supplied power to all the *Martlets* on the tower. The server enclosure was placed within a small shed located on site with the turbine. This shed contained all of the wind turbine's power circuitry and energy storage. The external antenna and power line that provided power to the *Martlets* exited the shed through an electrical conduit.

Once the sensors were deployed, the server was programmed to collect acceleration data from the *X* and *Z* axes on all units every 10 minutes at 1 kHz. Only 12 seconds of data could be collected during each 10 minute interval because of the limited RAM available on the *Martlet*. The intervals were spaced 10 minutes apart to ensure that all the data could be wirelessly transmitted back to the server even in the worst scenario of packet drops. While personnel were at the site, the server would be manually triggered to capture interesting environmental events (*e.g.*, sudden changes in wind speed or direction) that would likely not be captured randomly within the 12 sec window every 10 minutes.

7.3 Results

Over three days of testing (September 16 to 18, 2013) a total of 561 datasets were collected. The campaign only collected structural response data and did not collect loading data (*e.g.*, turbine torque and RPM were not measured while wind mean and variance were only collected on 15 minute intervals). As a result, the output-only data

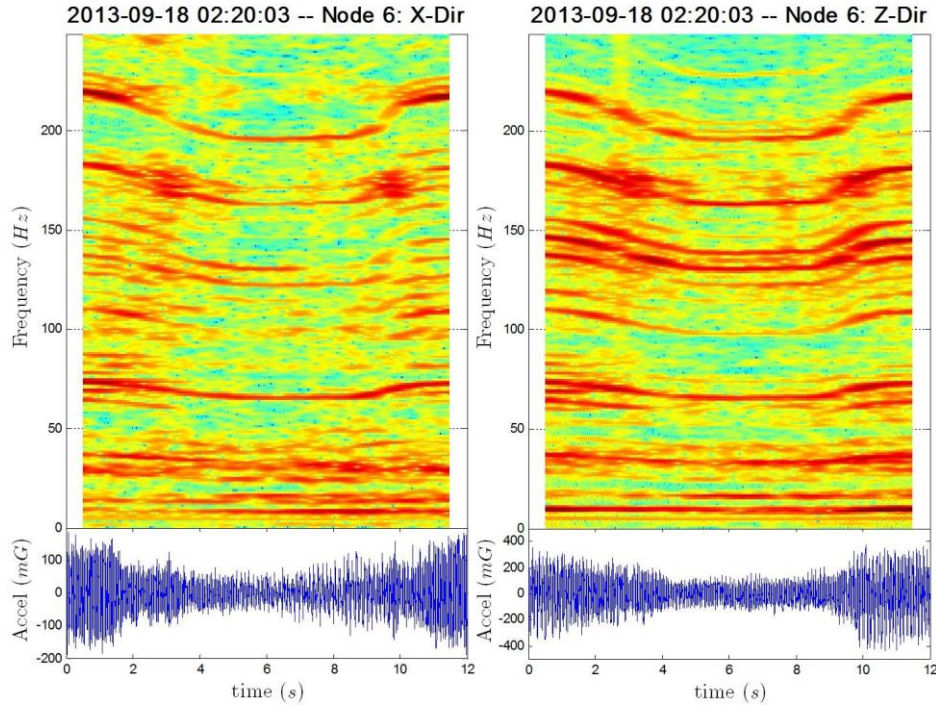


Figure 9. Orthogonal (X and Z directions) lateral tower responses measured from the top-most accelerometer (September 18, 2013 at 2:20am): (top) spectrograms; (bottom) time histories

precludes the use of input-output system identification. Fortunately, the structural response was still able to be extracted using output-only system identification techniques and engineering intuition.

Visual inspection of the acceleration time histories, like those displayed in the bottom of Figure 9 (which correspond to the tower's top-most accelerometer) shows the time varying magnitude of the tower vibrations. However, these complex signals are difficult to analyze in the time domain. The spectrogram plots (Figure 9) computed using a 1024 point FFT with 500 points of overlap are more enlightening. Many strong harmonics are shown all the way up to the cut-off frequency of the anti-aliasing filter. Furthermore, many of these harmonics slowly vary with time. Since blade speed was not measured, these changes in frequency cannot be directly accounted for, but visual observations made during testing confirm that as the blade speed decreased the frequency of the harmonics also decreased. It is also apparent that as the blade speed and possibly direction change, the response amplitude of the structure changes as the excitation harmonics excite different natural frequencies of the structure. The two short signals in Figure 9 make up only a minuscule portion of all the data collected, yet provide a surprising amount of insight. When inspecting other signals, similar behaviors emerge but also reveals some other key observations. For example, high frequency rattling (seen as a very broadband response) is apparent at nodes near the pivot point under some conditions.

In order to gain insight into the response of the entire structure, the signals from the X- and Z-direction of all the accelerometers were analyzed using the output-only frequency domain decomposition modal analysis technique. Since the excitation of the tower from the turbine rotation is narrow band and full of harmonics, excitation frequencies can be easily confused with normal modal frequencies. Also, the eigenvectors extracted from the data will correspond to operational deflection shapes. The tower's original design documents called for a normal cantilever mode shape in the E-W direction at 1.335 Hz and the other normal cantilever mode shape in the N-S direction at 1.133 Hz. Figure 10 shows the operational deflection shapes (ODS) extracted from the acceleration data collected on September 17, 2013 at 6:45 pm. It appears as if the order of the first two cantilever ODS has switched and increased to 1.58 Hz for the N-S cantilever and 2.00 Hz for the E-W cantilever. Confidence that these ODS correspond to mode shapes was reached after viewing the singular values of the FDD analysis done on all of the data sets collected as shown in Figure 11. The frequency of the peaks near 1.58 and 2.00 Hz do not change significantly over time even as the excitation changes. On the other hand, the ODS observed at the 3.92 Hz peak in the September

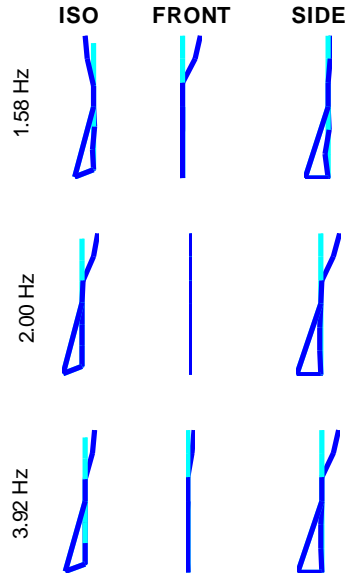


Figure 10. Operational deflection shapes (ODS) corresponding to the response time history collected on September 17, 2013 at 6:45pm. Dark shape is the ODS extracted and light shape is the undeformed tower.

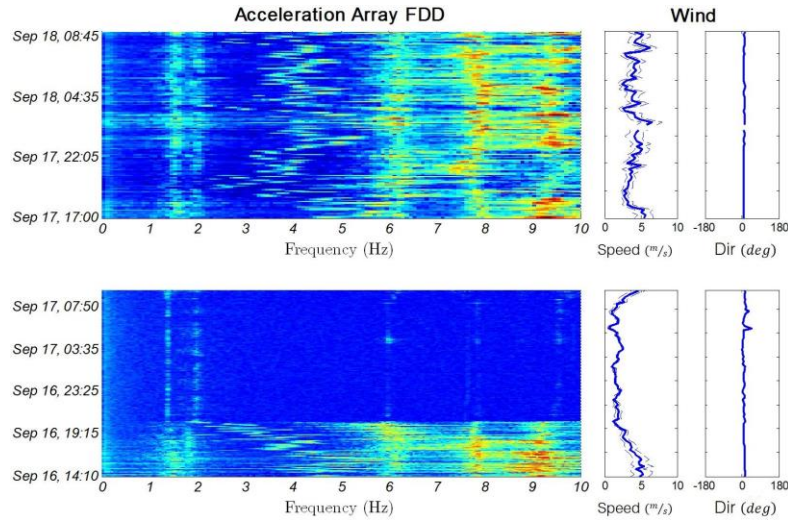


Figure 11. Spectral power of acceleration versus wind speed and direction as a function of time. Bottom corresponds to September 16 at 2:10pm to September 17 at 7:50am; top corresponds to September 17 at 5:00pm to September 18 at 8:45am.

17, 2013 6:45 pm dataset is very likely an operational deflection shape influenced by the load. The frequency of this peak slowly drifts around 4 Hz throughout the data, and the deflected shape corresponds to what would be expected from a slight imbalance in the turbine when rotating due to a wind from the south.

Observations of the spectrogram and mode shapes have been consistent with the expectation that blade rotation is the primary driver of the structural response. Comparing the spectral power of the acceleration against the wind speed and direction, as shown in Figure 11, is also logically consistent with the expectation that blade speed is highly correlated to wind speed. This is most obvious in comparing the movement in the peaks of spectral power between 2.5 Hz and 5.5 Hz with the variation in wind speed. At wind speeds of 5 m/s the structure responds around 5 Hz, but as the wind speed drops to 3 m/s the peak in response shifts closer to 3 Hz. Once the wind speed drops below 2.5 m/s, the blade rotation appears to stop causing a very significant drop in structural response amplitude. This observation is consistent with the Whisper 500 specification nominal blade speed of 500 RPM (8.3 Hz), cut-in

wind speed of 3.4 m/s and cut-out wind speed of 23 m/s. Although the excitation spectrum is not white at any given instant, considered over the entire 3 days of data, the modes of the structure should be able to be separated from the operational deflection shapes influenced by the load. The only caveat is the wind direction. Figure 11 shows that the wind is almost always coming from the south with little variation (depicted as the one-standard-deviation dashed lines). This means that while the rotational vibrations in the nacelle consistently excite the E-W direction of the tower, the N-S directions might not be sufficiently excited leading to ambiguity in the modal analysis associated with the N-S modes.

This preliminary investigation undertaken on the LANL wind turbine was able to achieve the original three main goals stated at the outset. The array of *Martlets* operated without incident for the duration of the study. During this time, data was successfully collected both automatically and manually for three days and two nights. By analyzing the spectral power and modal response of the structural acceleration using output-only system identification techniques, it was concluded that wind speed is the primary driver of structural response (as expected). The two primary cantilever mode shapes of the tower could be extracted along with an operational deflection shape. Without additional information provided regarding the precise loading on the turbine, a more quantitative analysis of the structural response and performance is difficult. Therefore, it should be proposed that further experimentation beyond this preliminary investigation should include additional sensors to try to identify the system input. A higher temporal fidelity measurement of wind speed would provide more information on the slow variations in frequency observed in the spectrogram of Figure 9. High data-rate measurements of the instantaneous voltage and current (and corresponding phase difference) produced by the turbine would provide correlation between the instantaneous wind speed and instantaneous power generation which could in turn be correlated to the torque applied by the nacelle.

8. MARTLET VALIDATION: STRUCTURAL VIBRATION CONTROL

Wireless sensing systems have been shown [7,31,44,45] to be capable of autonomously providing useful data regarding the performance of a structure. However, these systems have not yet autonomously ‘closed the loop’ by actuating the structure to improve its performance. Laboratory tests and simulations have shown proof-of-concept for the design and implementation of wireless structural control systems for earthquake hazard mitigation [27,46,47]. However, the control algorithms often neglect the effect of time-varying earthquake disturbances and state-dependent actuator constraints because algorithms that incorporate these nonlinear effects are unable to run in real-time on current wireless devices. The *Martlet* could potentially fill this gap with its dual-core real-time architecture. The following laboratory bench-scale experiment shows that the *Martlet* is capable of meeting the current state-of-the-art and potentially exceeding it.

In the experiment, a network of six *Martlets* was used to sense the inter-story drift of a 4-story shear structure model and to actuate two active mass dampers (AMDs). A photograph of the model structure with AMDs is shown in Figure 12. The structure is a single-bay, four-story building made of aluminum columns and acrylic floors with aluminum weights added to each floor. The floor height is 25 cm per story and the bays are 30.5 cm x 10.8 cm. The columns are 3.8 cm x 0.3 cm rectangular aluminum sections oriented in their flexurally weak direction. Each floor is instrumented with MTS magnetostrictive position sensors [48] connected to *Martlets* through a 25 Hz anti-aliasing filter. Additionally, the position signal is split and passed through an analog circuit that approximately differentiates the signal to produce an approximation of the inter-story velocity. On the second and third floor are active mass dampers from Quanser Inc. [49] consisting of a cart on a linear bearing that is driven by a motorized rack and pinion mechanism. A *Martlet* motor controller *wing* (Figure 3d) was developed to generate a 6V PWM signal to control the motor speed and in turn the horizontal cart motion. Cart acceleration equates to a force applied to the corresponding floor of the structure.

8.1 Linear Quadratic Regulation of AMD

The AMDs are controlled by a linear quadratic regulator (LQR) with full state feedback in which the state vector $x(t) \in \mathbb{R}^8$ is defined by the four inter-story displacements stacked on top of the four inter-story velocities. The controller’s aim is to reduce the inter-story displacement, $x_{1..4}(t)$, and forces applied by the AMDs, $u(t) \in \mathbb{R}^2$, according to (1). This control strategy is widely used in many of the structural control tests previously mentioned. Assuming a linear system and unconstrained control, the optimal feedback control law at each control step is defined by (2) where $G \in \mathbb{R}^{2 \times 8}$ is calculated using theory from [50] and MATLAB routines [51]. The weighting matrices Q and R were selected using the method in [50] with desired displacements of 2 cm and desired control voltages under 6V, resulting in (3).

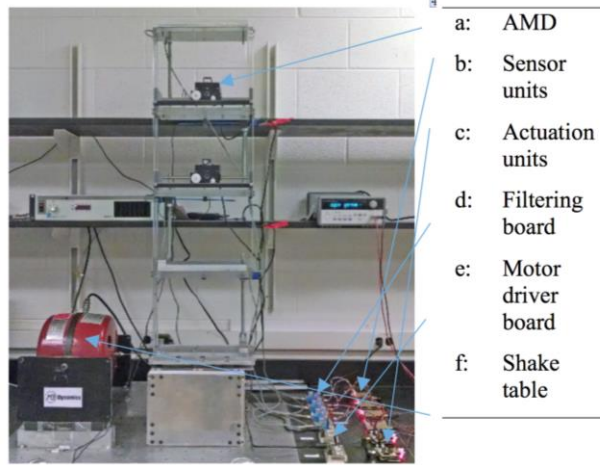


Figure 12. 4-story Structure with Two AMD Devices for Feedback Control.

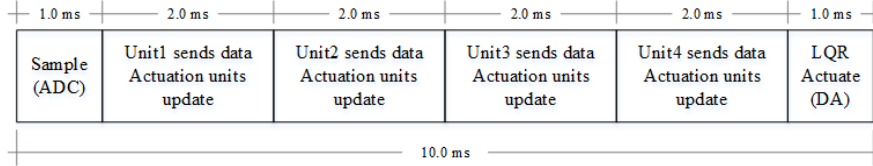


Figure 13. Timing diagram of the control actions taken by Martlet-based wireless feedback control system.

$$J = \int_0^{t_f} \left((x(t))^T \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} x(t) + (u(t))^T R u(t) \right) dt \quad (1)$$

$$u(k) = G x(k) \quad (2)$$

$$Q = \left(\frac{1}{0.02} \right)^2 \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -1 & 1 & & \\ \vdots & -1 & \ddots & \vdots \\ 0 & \cdots & 1 & 1 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -1 & 1 & & \\ \vdots & -1 & \ddots & \vdots \\ 0 & \cdots & 1 & 1 \end{bmatrix}, R = \frac{1}{6^2} \cdot I_2 \quad (3)$$

The controller was implemented in a distributed manner by programming each node with application specific code. The four sensing nodes broadcast their displacement and velocity readings at the beginning of each control step. The two actuating nodes concatenate the received state vector and multiply it by their corresponding row in the G matrix (e.g., the AMD on the second floor is the first row, and the third floor AMD is the second row). The solution to the calculation is then output through the motor controller *wing*. Figure 13 shows the timing of each node's operation in the distributed controller.

8.2 Experimental results and conclusions

The experiment was conducted by mounting the structure on a uniaxial shake table as shown in Figure 12. The base was laterally accelerated with a 50% scaled El Centro ground motion. The *Martlet* sensors operated autonomously, executing the control algorithm at a fixed rate of 50 Hz and 100 Hz in two different experiments. Sensor, actuator, and control calculation data was saved on each node and returned wirelessly to the server after the experiment concluded. The results of three different experiments are shown in Figure 14. The controlled response, both at 50 Hz and 100 Hz feedback rates, greatly outperform the uncontrolled case in which the AMDs were fixed. The *Martlet*'s computing power is leveraged to improve the performance when the feedback rate is increased to 100 Hz. This is an order of magnitude increase in rate over previous wireless structural control tests of similarly scaled networks [52]. This simple experiment shows the potential of *Martlet* networks to executed advanced control laws.

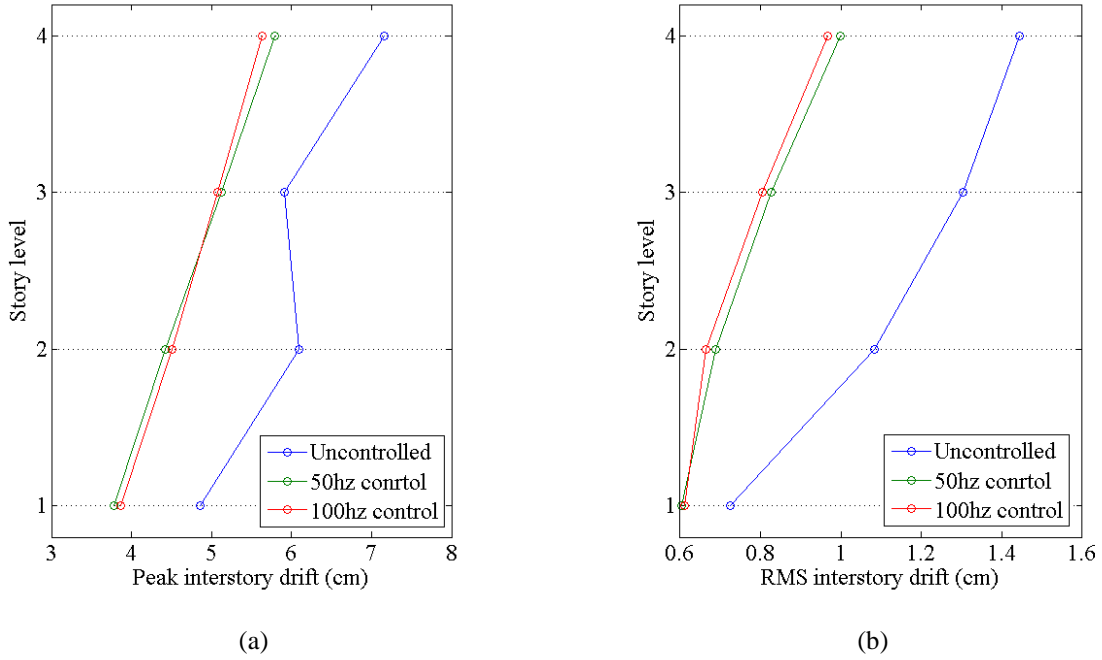


Figure 14. Comparison of control results: (a) peak inter-story drift, (b) RMS inter-story drift.

9. CONCLUSIONS

This paper presented the development, design, and initial tests of the *Martlet* wireless device as an extensible platform to enable research into wirelessly-enabled CPS. The network stack was tailored for low power, long distance, low latency, and reliable ad-hoc communications. Additionally, the network stack is backwards compatible with other popular wireless nodes using the IEEE 802.15.4 standards including the *Narada* wireless sensor. The dual-core microcontroller at the *Martlet*'s core enables control tasks to be performed with high temporal precision on the CLA, while less critical tasks are performed on the main core. A wide variety of digital and analog input-output channels are available allowing a large set of sensors and actuators to be interfaced to the *Martlet* through custom *Martlet wings*. The *Martlet*'s embedded OS operates in a state-machine that allows for rapid response of concurrent tasks in a relatively easy to understand architecture. Middleware and applications have been developed for sensing and control. The entire development process culminated in a computationally capable node for distributed monitoring and control of cyber-physical systems.

In addition to controlled laboratory tests of the circuitry and firmware, two field experiments were conducted showing that the *Martlet* meets or exceeds the current state-of-the-art. A deployment of a network of *Martlets* on a wind turbine tower at LANL successfully met the project requirements of identifying the main source of structural response under varying EOC. A bench-scale laboratory shake table test utilizing a network of *Martlets* to control two AMDs resulted in significantly improved structural performance compared to the uncontrolled structure. Additionally, the *Martlet* was able to execute the LQR control algorithm an order of magnitude faster than previous wireless control systems. These two experiments only begin to show the potential for the *Martlet*. Work must still be done to improve usability, power management, parallel code execution, and scalability. There are also opportunities to develop new *wings* for applications that were not originally envisioned. The authors' hope is that a community will form around the use and continued development of the *Martlet*, leading to a powerful common platform for enabling groundbreaking research in cyber-physical infrastructure systems.

ACKNOWLEDGEMENTS

This research is partially funded by the US Office of Naval Research (contracts N00014-05-1-0596 and N00014-09-C0103 granted to Jerome P. Lynch), the Research and Innovative Technology Administration of U.S. Dept. of Transportation (contract #DTRT12GUTC12 granted to Yang Wang), Georgia Dept. of Transportation (contract #RP12-21 granted to Yang Wang), and Georgia Institute of Technology University Transportation Center. The authors would also like to express their gratitude to Robert Gordenker, University of Michigan, for assistance during the design and testing of *Martlet* prototypes. The LANL field validation of the Martlet would not have been possible without Chuck Farrar, Curtt Ammerman, Stuart Taylor, and Eric Flynn at the Los Alamos National Laboratory (LANL).

REFERENCES

- [1] L. Sha, S. Gopalakrishnan, X. Liu, and Q. Wang, "Cyber-Physical Systems: A New Frontier," *2008 IEEE Int. Conf. Sens. Networks, Ubiquitous, Trust. Comput. (sutc 2008)*, 1–9, Ieee (2008) [doi:10.1109/SUTC.2008.85].
- [2] J. Coats, *A New Dictionary of Heraldry*, p. 376, Ptd. for Aaron Ward, London (1747).
- [3] E. G. Straser, A. S. Kiremidjian, T. H. Meng, and L. Redlefsen, "A modular, wireless network platform for monitoring structures," in *16th Interational Modal Anal. Conf.* **1**, pp. 450–456, SEM, Santa Barbara, CA (1998).
- [4] J. P. Lynch, K. H. Law, A. S. Kiremidjian, T. W. Kenny, E. Carryer, and A. Partridge, "The design of a wireless sensing unit for structural health monitoring," in *Proc. 3rd Int. Work. Struct. Heal. Monit.*, Stanford, CA (2001).
- [5] R. A. Swartz, D. Jung, J. P. Lynch, Y. Wang, D. Shi, and M. P. Flynn, "Design of a wireless sensor for scalable distributed in-network computation in a structural health monitoring system," in *5th Int. Work. Struct. Heal. Monit.*, Stanford, CA (2005).
- [6] J. Kim, S. Nadukuru, M. Pour-Ghaz, J. P. Lynch, R. L. Michalowski, A. S. Bradshaw, R. A. Green, and W. J. Weiss, "Assessment of the Behavior of Buried Concrete Pipelines Subjected to Ground Rupture: Experimental Study," *J. Pipeline Syst. Eng. Pract.* **3**(1), 8–16 (2012) [doi:10.1061/(ASCE)PS.1949-1204.0000088.].
- [7] M. Kurata, J. Kim, J. P. Lynch, G. W. Van Der Linden, H. Sedarat, E. Thometz, P. Hipley, and L. -H. Sheng, "Internet-Enabled Wireless Structural Monitoring Systems: Development and Permanent Deployment at the New Carquinez Suspension Bridge," *J. Struct. Eng.* **139**(10), 1668–1702 (2012) [doi:10.1061/(ASCE)ST.1943-541X.0000609].
- [8] R. A. Swartz, A. T. Zimmerman, J. P. Lynch, J. Rosario, T. Brady, L. Salvino, and K. H. Law, "Hybrid wireless hull monitoring system for naval combat vessels," *Struct. Infrastruct. Eng.* **8**(7), 621–638, Taylor & Francis (2010) [doi:10.1080/15732479.2010.495398].
- [9] A. T. Zimmerman and J. P. Lynch, "A parallel simulated annealing architecture for model updating in wireless sensor networks," *IEEE Sens. J.* **9**(11), 1503–1510 (2009).
- [10] J. Kim and J. P. Lynch, "Autonomous Decentralized System Identification by Markov Parameter Estimation Using Distributed Smart Wireless Sensor Networks," *J. Eng. Mech.* **138**(5), 478–490 (2012) [doi:10.1061/(ASCE)EM.1943-7889.0000359.].
- [11] R. A. Swartz, Y. Wang, J. P. Lynch, K. H. Law, and C. Loh, "EXPERIMENTAL VALIDATION OF MARKET-BASED CONTROL USING WIRELESS SENSOR AND ACTUATOR NETWORKS," *Relation* **10**(1.59), 5825, Citeseer (2010).
- [12] R. A. Swartz and J. P. Lynch, "Partial Decentralized Wireless Control Through Distributed Computing for Seismically Excited Civil Structures: Theory and Validation," in *Am. Control Conf. 2007. ACC '07*, pp. 2684–2689 (2007).
- [13] R. A. Swartz and J. P. Lynch, "Strategic Network Utilization in a Wireless Structural Control System for Seismically Excited Structures," *J. Struct. Eng.* **135**(5), 597–608, ASCE (2009).
- [14] M. B. Kane, J. P. Lynch, and K. Law, "Market-based control of shear structures utilizing magnetorheological dampers," in *Am. Control Conf. (ACC), 2011*, pp. 2498–2503 (2011).
- [15] Y. Wang, J. P. Lynch, and K. H. Law, "Decentralized Hinf; controller design for large-scale civil structures," *Earthq. Eng. Struct. Dyn.* **38**(3), 377–401, John Wiley & Sons (2009).

- [16] M. B. Kane and J. P. Lynch, "An Agent-Based Model-Predictive Controller for Chilled Water Plants using Wireless Sensor and Actuator Networks," in *Am. Control Conf. (ACC)*, 2012, pp. 1192–1198, IEEE, Montreal, Canada (2012).
- [17] J. P. Lynch, A. Sundararajan, K. H. Law, A. S. Kiremidjian, and E. Carryer, "Embedding damage detection algorithms in a wireless sensing unit for attainment of operational power efficiency," *Smart Mater. Struct. IOP* **13**(4), 800–810 (2004).
- [18] D. K. Sohraby, D. Minoli, and T. Znati, *Wireless sensor networks*, p. 307, John Wiley & Sons, Inc., Hoboken, NJ (2007) [doi:10.1002/047011276X].
- [19] B. Peters, "Sensing without wires," in *Mach. Des.*, Cleveland, OH (2005).
- [20] IEEE Computer Society, "IEEE Standard for Information technology- Telecommunications and information exchange between systems- Local and metropolitan area networks- Specific requirements--Part 15.4: Wireless MAC and PHY Specifications for Low-Rate WPANs," New York, NY, p. 323 (2006).
- [21] ZigBee Alliance, "Specifications," 2014.
- [22] D. Chen, M. Nixon, A. Mok, and S. (Online Service), *WirelessHART™ Real-Time Mesh Network for Industrial Automation*, Springer Science+Business Media, LLC, Boston, MA (2010).
- [23] T. Nagayama, B. F. Spencer Jr., and J. A. Rice, "Structural health monitoring utilizing Intel's Imote2 wireless sensor platform," *Proc. SPIE--Sensors Smart Struct. Technol. Civil, Mech. Aerosp. Syst.* 2007 **6529**(2), 6592943 (2007).
- [24] B. A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta, "Ensuring Safety , Security , and Cyber – Physical Systems," *Proc. IEEE* **100**(1), 283–299 (2012) [doi:10.1109/JPROC.2011.2165689].
- [25] J. A. Rice, K. Mechtov, S. Sim, T. Nagayama, S. Jang, R. Kim, B. F. Spencer, G. Agha, and Y. Fujino, "Flexible Smart Sensor Framework for Autonomous Structural Health Monitoring," 423–438 (2010).
- [26] M. Kurata, J. Kim, Y. Zhang, J. P. Lynch, G. W. van der Linden, V. Jacob, E. Thometz, P. Hipley, and L.-H. Sheng, "Long-Term Assessment of an Autonomous Wireless Structural Health Monitoring System at the New Carquinez Suspension Bridge," 24 March 2011, 798312–798312–9 [doi:10.1117/12.880145].
- [27] Y. Wang, J. Lynch, and K. Law, "Decentralized H_∞ controller design for large-scale civil structures," *Earthq. Eng. Struct. Dyn.* **38**(3), 377–401 (2009) [doi:10.1002/eqe].
- [28] J. Hamano, L. Torvalds, and et al., "Git," 1.8.4 (2013).
- [29] ChaN, "Petit FAT FS," R0.04b, Saitama, Japan (2007).
- [30] Microstrain Inc., "Wireless System Networks | Microstrain," 2012, <<http://www.microstrain.com/wireless/systems>>.
- [31] Bridge Diagnostics Inc., "STS WiFi Wireless Structural Testing System," Boulder, CO (2012).
- [32] National Instruments Inc., "Wireless Measurement Device Selection Guide," 2012, <<http://sine.ni.com/np/app/main/p/ap/global/lang/en/pg/1/sn/n24:Wireless/fmid/2988/>>.
- [33] Siemens AG, "WirelessHART - Innovation for the process industry," 2012, <<http://www.automation.siemens.com/w1/automation-technology-wirelesshart-18957.htm>>.
- [34] R. Bennett, B. Hayes-Gill, J. A. Crowe, R. Armitage, D. Rodgers, and A. Hendroff, "Wireless monitoring of highways," in *Smart Syst. Bridg. Struct. Highw. Proceeding*, pp. 173–182, Newport Beach, CA (1999).
- [35] K. Mitchell, N. Dang, P. Liu, V. S. Rao, and H. J. Pottinger, "Web-controlled wireless network sensors for structural health monitoring," in *Smart Struct. Mater. – Smart Electron. MEMS Proceeding*, pp. 234–243, Newport Beach, CA (2001).
- [36] D. L. Mascarenas, M. D. Todd, G. Park, and C. R. Farrar, "Development of an impedance-based wireless sensor node for structural health monitoring," *Smart Mater. Struct.* **16**(6), 2137–2145 (2007).
- [37] D. Inaudi and L. Manetti, "Reinforced Concrete Corrosion Wireless Monitoring System," in *4th Int. Conf. Struct. Heal. Monit. Intell. Infrastruct.* 2009(July), pp. 1–10, Zurich (2009).
- [38] Civionics, "Narada Datasheet," Ann Arbor, MI, p. 1 (2012).
- [39] T. Nagayama, "iMote2 and TinyOS Programming," Tokyo, Japan (2009).
- [40] P. Horowitz and W. Hill, *The art of electronics*, p. xxiii, 1125 p., Cambridge University Press, Cambridge [England] ; New York (1989).
- [41] Southwest Windpower Inc., "Whisper 500 Owners Manual," Flagstaff, AZ, p. 46 (2000).
- [42] J. Chipka, A. Lisicki, and C. Nguyen, "Experimental Characterization and Predictive Modeling of a Residential-Scale Wind Turbine," in *Conf. Proc. Soc. Exp. Mech. Ser.* 2013, pp. 521–533, Springer New York (2013) [doi:10.1007/978-1-4614-6555-3_56].
- [43] L. Los Alamos National Security, "The Weather Machine: Los Alamos National Laboratory," 2013, <<http://environweb.lanl.gov/weathermachine/>> (16 October 2013).

- [44] C. McDonald, "Performing Structural Health Monitoring of the 2008 Olympic Venues Using NI LabVIEW and CompactRIO," *Natl. Instruments Case Study*, <<http://sine.ni.com/cs/app/doc/p/id/cs-11279>>.
- [45] J. W. Park, S. Cho, H.-J. Jung, C.-B. Yun, S. A. Jang, H. Jo, B. F. J. Spencer, T. Nagayama, and J.-W. Seo, "Long-term structural health monitoring system of a cable-stayed bridge based on wireless smart sensor networks and energy harvesting techniques," in *5th World Conf. Struct. Control Monit.*, pp. 1–6, Tokyo (2010).
- [46] S. Seth, J. P. Lynch, and D. M. Tilbury, "Wirelessly networked distributed controllers for real-time control of civil structures," in *Am. Control Conf. 2005. Proc. 2005*, pp. 2946–2952 vol. 4 (2005).
- [47] B. Li, Z. Sun, K. Mechitov, G. Hackmann, C. Lu, S. J. Dyke, G. Agha, and B. F. J. Spencer, "Realistic Case Studies of Wireless Structural Control," in *ICCPS'13*, ACM, Philadelphia, PA (2013).
- [48] M. Sensors, "Temposonics C-Series OEM-Sensor Analog," Cary, NC, p. 4 (2013).
- [49] Quanser Inc., "Bench-scale shake tables," Markham, Ontario, p. 3 (2013).
- [50] A. E. Bryson and Y.-C. Ho, *Applied optimal control: optimization, estimation, and control*, p. 481 p., Hemisphere Pub. Corp. ; distributed by Halsted Press, Washington : New York (1975).
- [51] MathWorks, *Control system toolbox: for use with MATLAB : getting started*, MathWorks, Inc. (2000).
- [52] C. H. Loh, J. P. Lynch, K. C. Lu, Y. Wang, C. M. Chang, P. Y. Lin, and T. H. Yeh, "Experimental verification of a wireless sensing and control system for structural control using MR dampers," *Earthq. Eng. Struct. Dyn.* **36**(10), 1303–1328, Citeseer (2007).