

UC Berkeley

Controls and Information Technology

Title

Design of wireless sensor networks for building management

Permalink

<https://escholarship.org/uc/item/21v2j5v2>

Author

Sharma, Anshuman

Publication Date

2003-10-01

Peer reviewed

Design of Wireless Sensor Networks for Building Management Systems

by Anshuman Sharma

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences,
University of California at Berkeley, in partial satisfaction of the requirements for the
degree of **Master of Science**, Plan II

Approval for the Report and Comprehensive Examination:

Committee:

Professor Ed Arens
Research Advisor

(Date)

* * * * *

Professor Kris Pister
Second Reader

(Date)

Design of Wireless Sensor Networks for Building Management Systems

Copyright Fall 2003

Anshuman Sharma

Abstract

Design of Wireless Sensor Networks for Building Management Systems

by

Anshuman Sharma

Master of Science in Electrical Engineering

University of California at Berkeley

Professor Ed Arens, Research Advisor

We provide a detailed study of the application of wireless sensor networks to a real-world building management system. A set of design requirements was discussed for the development of a sensor network architecture for an HVAC control system. We propose a system architecture that can be easily integrated into any existing building control system. We implemented a testbed to evaluate control strategies using ad hoc wireless sensors. The testbed was designed within the controlled environment chamber in the College of Environmental Design and UC Berkeley. Preliminary results from control strategies have allowed us to determine the effectiveness of the sensor network. The design of the system was suited for the kind of control applications that are indicative of the field. We discussed the lessons learned from our tests and propose future experiments with multiple ad hoc sensors to fully investigate the value added benefit of integrating this technology into the legacy architecture.

Acknowledgements

This work would not have been possible without the guiding light of Cliff Federspiel and the help and support of Fred Bauman in setting up the chamber and its controls.

Also instrumental were Kris Pister who egged me on in times of duress and Nick Sitar and Dave Doolin who were always understanding and motivated me to create this piece of work.

Table of Contents

Approval Page.....	ii
Abstract.....	iii
Acknowledgements.....	iv
Table of Contents.....	v
List of Tables.....	vi
List of Figures and Illustrations.....	vii
Chapter 1: Introduction.....	1
Chapter 2: Building Management Systems.....	4
2.1 Objective.....	4
2.2 Methods.....	5
2.2.1 Time Control Methods (for heating).....	5
2.2.2 Temperature control methods.....	6
2.2.3 Lighting control systems.....	7
2.2.4 Building Management Systems.....	7
2.3 History of development.....	8
2.4 Control Networks: Today.....	9
2.5 Role of Sensor Networks.....	10
Chapter 3: HVAC Control Architecture.....	12
Chapter 4: HVAC Control Testbed.....	18
4.1 Sensor Network Node.....	18
4.2 Sensor Board.....	22
4.3 Actuation Node.....	24
4.4 HVAC Control Module.....	26
4.4.1 Process Model.....	26
4.4.2 PID Control.....	28
4.5 Controlled Environment Chamber.....	30
4.5.1 Chamber Design and Setup.....	33
Chapter 5: Testbed Results.....	36
5.1 Experiment Runs.....	37
5.1.1 First Set.....	39
5.1.2 Second Set.....	39
5.1 Sources of Error.....	42
5.2 Further Experimentation.....	43
5.3 Future Motivation.....	43
Chapter 6: Related Work.....	45
Chapter 7: Conclusion.....	48
Bibliography.....	49

List of Tables

Table 2.1: Comparison of LONWORKS and BACnet Architectures	11
Table 4.1: Family of Motes.....	19
Table 4.2: Chamber Specifics	35
Table 5.1: First Run	38
Table 5.2: Second Run.....	40

List of Figures and Illustrations

Figure 3.1: Evolution of Control Systems	13
Figure 3.2: Integration of Sensor Network into HVAC Control	16
Figure 4.1: Sensor Node and Basic Sensor Board	23
Figure 4.2: Process Model: Feedback Loop & Static Process Characteristic.....	27
Figure 4.3: Controlled Environment Chamber	31

Chapter 1: Introduction

The emergence of wireless sensor networks [9] has enabled new classes of applications for distributed systems [10] that filter into very many interdisciplinary fields. These networks have been used for solving problems in the fields of distributed control [4][5], tracking and inventory [3], structural monitoring [26], fire-safety [25], habitat monitoring [2][14] etc.

In the larger class of distributed control applications exists the area of Building Management Systems [21]. The objective of a BMS is to achieve optimal level of control of occupant comfort while minimizing energy use. These control systems are the integrating component to fans, pumps, heating/cooling equipment, dampers, mixing boxes, and thermostats. Monitoring and optimizing temperature, pressure, humidity, and flow rates are key functions of modern building control systems. These systems are inherently accustomed to using networked sensors in some wired configuration and thus the integration of wireless sensor networks becomes harder to sell to potential users. Users in the field need to perceive sensor networks as an attractive addition with definite added benefits. Some of the possible benefits are ease and cost of installation. These networks are ad hoc systems, which can be built up and torn down with relative ease without disturbing the environment. However, a key difference between the existing sensor peripherals and wireless sensor networks is the flexibility of the platform [11]. The underlying hardware and software platform allows local processing and storage, and can perform complex filtering and triggering functions, even application-specific or sensor

specific aggregation, filtering, and compression of collected data [8]. The ability to communicate not only allows sensor data and control information to be communicated across the network of nodes, but nodes to cooperate in performing more complex tasks, such as statistical sampling, data aggregation, and system health and status monitoring. Overall low-power design from radios, to protocol stack to sensing utilities allow these nodes to run for extended periods of time on regular batteries. The computing and networking capabilities allow sensor networks to be reprogrammed or retasked after deployment in the field, and that is one of the biggest benefits.

Our goal is to develop an effective sensor network architecture for building management applications, in general. As a starting step we focus on one aspect of building management, HVAC control. The ability of sensor networks to provide a value added benefit to this application will enable people in the broader area of building management to embrace the new technology and further its spread.

This thesis develops a specific HVAC control application, one that is largely representative of the domain. It presents a collection of requirements, constraints and guidelines for integration of sensor networks into existing building management infrastructure. Included as part of the design, are the hardware and sensor platforms, control units, and data access and management facility. A simple solution to using multiple ad hoc sensors integrated seamlessly into the control system provides insight into the behaviour of long-term sensor network deployments.

The rest of this thesis is organized as follows. Chapter 2 introduces the general area of BMS, breaks down the different control methods, a brief history that has brought the field to today where it is one of the classic examples of distributed control. Chapter 3

describes variants of a typical BMS (HVAC control) architecture and a wireless sensor network framework for easy integration. A brief description of each of the key components in an HVAC control system, and the sensor network that will facilitate its operation is provided. To test out the architecture a testbed was implemented in a controlled environment chamber that mimicked an urban office setting. A detailed description of the sensor network: the computational node, the sensor hardware, actuation node, and the test setup are described in Chapter 4. Our initial results and insights into future tests and control strategies are showcased in Chapter 5. Chapter 6 discussed the relevant related work in the field, their approaches and design decisions. Chapter 7 provides concluding remarks.

Chapter 2: Building Management Systems

The technology area of building automation/management systems (BAS/BMS) and controls includes a variety of systems, over a wide range of complexity, designed for the control, monitoring and optimization of various functions and services provided in a building, including heating and cooling, ventilation, lighting and often the management of electric appliances. They make environments more comfortable, safe and efficient by integrating systems such as heating, air conditioning, lighting, security and telecommunications rolled into one centrally controlled, automated system. In order to do this effectively different systems need to be able to communicate and interact with each other.

2.1 Objective

The primary objective of such a system is to achieve an optimal level of control of occupant comfort while minimizing energy use. Monitoring temperature, pressure, humidity occupancy and flow rates are key functions of modern building control systems. A BMS has to be properly installed and commissioned for optimal operation and to realize potential savings. Energy efficiency can be optimized by a combination of scheduling, controlling temperature and using system economizer functions. Sensors out of calibration can lead to enormous energy waste. Integration of other auxiliary functions such as fire detection and suppression and security and occupancy detection can result in substantial cost savings.

The basic control technologies have been in existence for some time. Systems available range in complexity, from the extreme case of the timer-controlled water heater or thermostatic radiator valves (TRVs), to the so-called “intelligent houses” which manage everything from the security and safety systems to the air conditioning, lighting and ventilation system, to telemetric services and to most appliances of a house according to efficiency criteria.

The use of these technologies allows the optimization of various services often with large energy savings. There are numerous methods by which building services within buildings can be controlled. Most systems seek to control either by:

- time: when a service such as heating or lighting is provided and when it should not be provided
- a parameter representative of the service like temperature for space heating or luminance for lighting. This can also vary with time.

2.2 Methods

Control and BMS methods are described hereafter [21]:

2.2.1 Time Control Methods (for heating)

- Time switches turn on and off the heating (or water heating) system at pre-selected periods (of the day, of the week)

- Optimizers: these controls start the heating system in a building at a variable time to ensure that, whatever the conditions, the building reaches the desired temperature when occupancy starts.

2.2.2 Temperature control methods

- Frost protection generally involves running heating system pumps and boilers when external temperature reaches a set level (0°C) or less in order to protect against freezing
- Compensated systems: which control flow temperatures in the heating circuit relative to external temperature thus allowing a rise in the circuit flow temperature when outside temperature drops
- Thermostatic radiator valves: these units sense space temperature in a room and throttle the flow accordingly through the emitter (radiator and converter) to which they are fitted
- Modulating control: can be applied to most types of heat emitters and is used to restrict the flow depending on the load demand and this controlling the temperature
- Proportioning control: involves switching equipment on and off automatically to regulate output
- Other methods are thermostats, occupation sensing (described hereafter for lighting control) and interactive control

2.2.3 Lighting control systems

Different control systems exist, either based on time control or a required level of luminance or use of lighting.

- Zoning: Lights are switched on in zones corresponding to the use and layout of illuminated areas, in order to avoid lighting a large area if only a small part of it requires light
- Timed control: to switch on and off automatically in each zone to match a prerequisite schedule for light use
- Occupancy sensing: In areas which are occupied intermittently, occupancy sensors can be used to indicate whether or not anybody is present and switch the light on or off accordingly. Detection systems are based on ultrasonic movement and infrared sensing
- Light level control: this consists of switching or dimming artificial lighting to maintain a light level measured by a photocell. It is particularly necessary to give value to ambient daylighting.

2.2.4 Building Management Systems

These technologies consist of both hardware and software.

The hardware is typically represented by one (or more) control and processing units and by a number of peripheral devices (which control the operation of say, heating or cooling systems, artificial light-sources or other appliances and which can be represented by sensors, thermostats, etc.,) connected to the control units. The control unit, based on the information supplied by some of the peripherals or based on pre-set instructions, runs the system. The control unit can be as simple as a relay or a timer switching on or off an

electric water heater or as sophisticated as a microprocessor operating on “fuzzy logic”. Commands can be sent from the central unit to the peripheral units through Ethernet cable, power-lines or telephone lines, fibre-optic cables or even using radio transmissions. The software is simply the program and the instructions that allow the control unit to manage the operations of the peripheral devices and of the appliances.

2.3 History of development

Thermostats and TRVs were the first control equipment to disseminate in the building owing to the first energy crisis.

Building management systems developed in the ‘80s in the residential and services sectors, as simplified applications of systems and technologies already developed in the industrial sector in the ‘70s to automate production processes and to optimize plant performances [19][21].

These technologies were mostly based on the concept of Programmable Logic Controllers (PLC). Subsequent developments, as well as most residential applications were later based on “distributed-intelligence microprocessors”, whereby “intelligent” peripheral units are capable of managing a variety of tasks and functions while a central unit acts as a supervisor.

Once the trend to industrial automation had expanded sufficiently in industrial countries, these technologies sought new applications and markets in other sectors. As these systems were rather expensive, the richest segments of the services and the residential sector were the first target. Hence the first buildings to be equipped with systems and devices were banks, large businesses and prestige buildings, and a few villas

and mansions for wealthy people. The most widely used applications at the beginning were for security/safety purposes and later on for air conditioning (heating/cooling) purposes. The concept of “intelligent building” in its current definition was born about 8-10 years ago.

Due to their cost, the first applications to become popular (and the ones with faster market penetration rates) in the services and residential sector were security and safety devices (alarm systems against burglars and intruders, fire alarm systems and smoke detectors, gas-presence detectors). Other applications, especially those for energy management, have had a much slower diffusion. Besides the cost of the equipment, which remains high even when it is not very sophisticated, installation costs must be supported. These are substantial and when already existing buildings have to be retrofitted they are usually quite high. This is one of the primary reasons why any means of reducing the installation cost for these systems could help increase the diffusion rate. Wireless peripheral devices are key to bringing this to fruition with their minimal setup cost.

2.4 Control Networks: Today

There are many ways to create automated systems, from pneumatics to custom, proprietary hardware and software solutions to open interoperable standards-based control networks. The open device networks have common traits including an open protocol [19]; a prescribed architecture (flat or tiered); device level interoperability; and a network operating system for easy management, installation and remote services. In this sense, automation networks have evolved similarly to PC networks. There are two

competing standards today, LONWORKS[®] and BACnet[®]. The two architectures aspire the same goals of vendor independence and interoperability with very different implementation requirements. Table 1 compares the two architectures [19][20].

2.5 Role of Sensor Networks

The field of building automation can reap great rewards from the advent of the wireless sensor networks. A BAS already employs a wide variety of sensing peripherals that are networked together to provide a wired sensor network. Adding wireless sensory terminals/peripherals can greatly reduce the cost and time required for installation and maintenance of such systems.

Addition of wireless nodes to the existing systems has caught the attention of control systems manufacturers, who are now actively involved in seeking solutions in this space. As a result sensor network companies are gearing towards providing wireless nodes that can be easily integrated into existing open system architectures.

Table 2.1: Comparison of LONWORKS and BACnet Architectures

	LONWORKS	BACnet
History	1988 – Released by the Echelon Corporation 1994 – LONMARK® Organization formed 1999 – ANSI/EIA Standard	1987 – ASHRAE committee formed 1995 – BACnet Standard released 1995 – ASHRAE and ANSI standard
Goals	Vendor Independence Interoperability	Vendor Independence Interoperability
Architecture	A “bottom up” solution focused on the controller Open protocol Flat architecture	A “top down” solution focused on HMI integration Open standard Tiered architecture
Protocols	LonTalk® Embedded into every Neuron® Chip Can be ported to any processor, from 8-bit microcontrollers to 32-bit microprocessors Every LonWorks device uses LonTalk Supports various media – 1 protocol	Multiple protocols supported Ethernet, ARCnet, MS/TP, PTP, LonTalk, IP All industry standard protocols Each with specific implementation and media requirements
Controllers	Neuron Chip processor Neuron C programming language I/O Channels Transceiver Hosted controller Controllers have “out of the box” commonality	Processor independent Programming language independent I/O Channels Final controller specification at manufacturer’s discretion Controllers have nothing in common “out of the box”
Distribution	Direct from manufacturer as part of a complete solution Many companies produce LONWORKS devices (lighting, drives, power metering, diagnostic, etc.) Individual devices from an independent distributor who represent multiple vendors	Direct from manufacturer as part of a solution Very few companies produce solution independent BACnet devices (lighting controllers, diagnostic tools, gateways) No independent distribution
Topology	Flat architecture No gateways Change media, use same protocol Every device is a peer on the network Each device can communicate directly with the HMI using SNVT and SCPT data formats	Tiered architecture Gateway solution Native BACnet solution Change media, new protocol Controllers are grouped behind supervisory devices Typical implementation uses multiple protocols

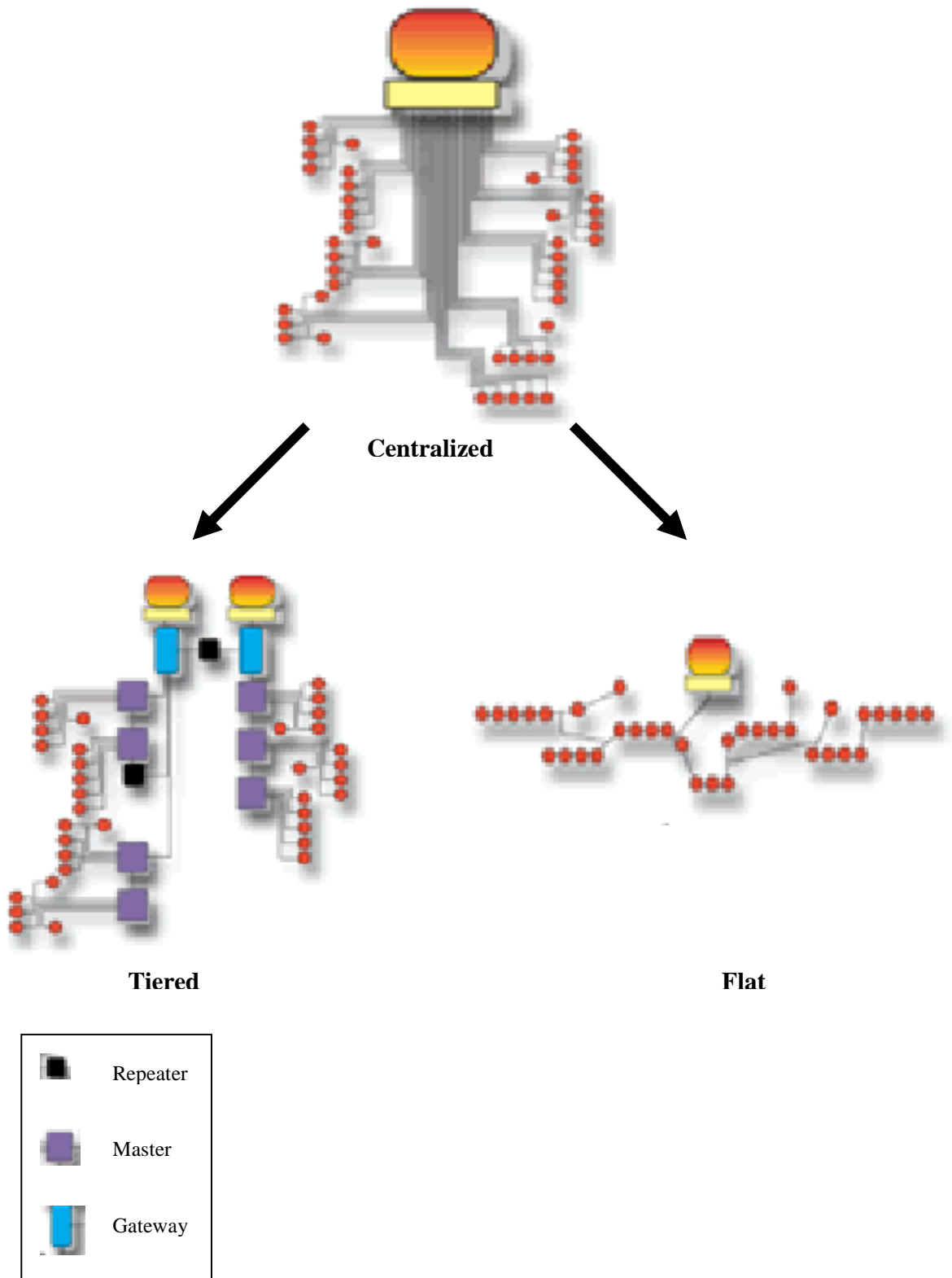
Chapter 3: HVAC Control Architecture

In order to integrate wireless sensors into the existing BMS, the new wireless sensor nodes/peripherals have to follow an architecture that can be easily assimilated into existing architectures. Here we describe variants of the BMS architecture and a wireless sensor network framework for easy integration.

We look at the HVAC control architectures and how they have evolved over time. In the early days, cabling was installed in a point-to-point fashion between electrical panels, the sensor inputs and actuator outputs. The functionality of these control systems was relatively rudimentary and inflexible, and adds, moves, and changes required extensive rerouting of wiring and connections. Over time, the control architecture has moved from a closed, wiring-intensive, centralized system towards a more distributed approach. Figure 3.1 shows the evolution of control systems [19].

The earlier systems comprised of smaller installations, hence a purely centralized scheme with a single master and multiple slaves sufficed. As systems became more complex with the increase in capabilities of BMS, multi-master multi-slave systems started to proliferate. This gave rise to tiered architectures, with gateways, controllers and repeaters. A contrary school of design has professed the use of a flat peer-to-peer (P2P) architecture. P2P architectures lack single points of failures inherent in any hierarchical architecture. Further, device failures in a P2P design are likely to affect just the one device, instead of many as in the case non-P2P architectures.

Figure 3.1: Evolution of Control Systems



Both tiered and P2P architectures lend themselves to the proposed integration of sensor networks, for which distributed control is a primary feature. The primary peripheral addition will be in the form of an autonomous sensor node. These small, battery-powered devices will be placed in areas of interest. Each sensor node collects sensory information, based on the sensing capabilities it is allowed. One of the important issues for deployment is the density of deployment, which is primarily dictated by the particular application, in this case HVAC control. High spatial resolution for the physical phenomenon can be achieved through dense deployment of sensor nodes [7]. Deviating from traditional approaches of less number of quality sensors with sophisticated application specific processing, this architecture provides higher robustness against occlusions and component failures.

The sensor network architecture for an HVAC application is an optional, flat or tiered. Samples of sensor data originate at the lowest sensor peripherals or *sensor nodes*. These nodes perform general purpose computing and networking in addition to application-specific sensing. Nodes are deployed in clusters, where each cluster is representative of a particular geographic region of interest. The sensor nodes transmit their data through the clusters in a multi-hop fashion to the *cluster-head*. The cluster-head can perform some sort of aggregation function over the data received from each of the sensors [33] before it reports it the control unit. The period of transmission at the sensor node is large, since the physical quantities that are sensed: airflow (stays near constant most of the time) and temperature do not vary much at a smaller timescale. The cluster-head is responsible for transmitting the sensor data from the cluster to a control unit that

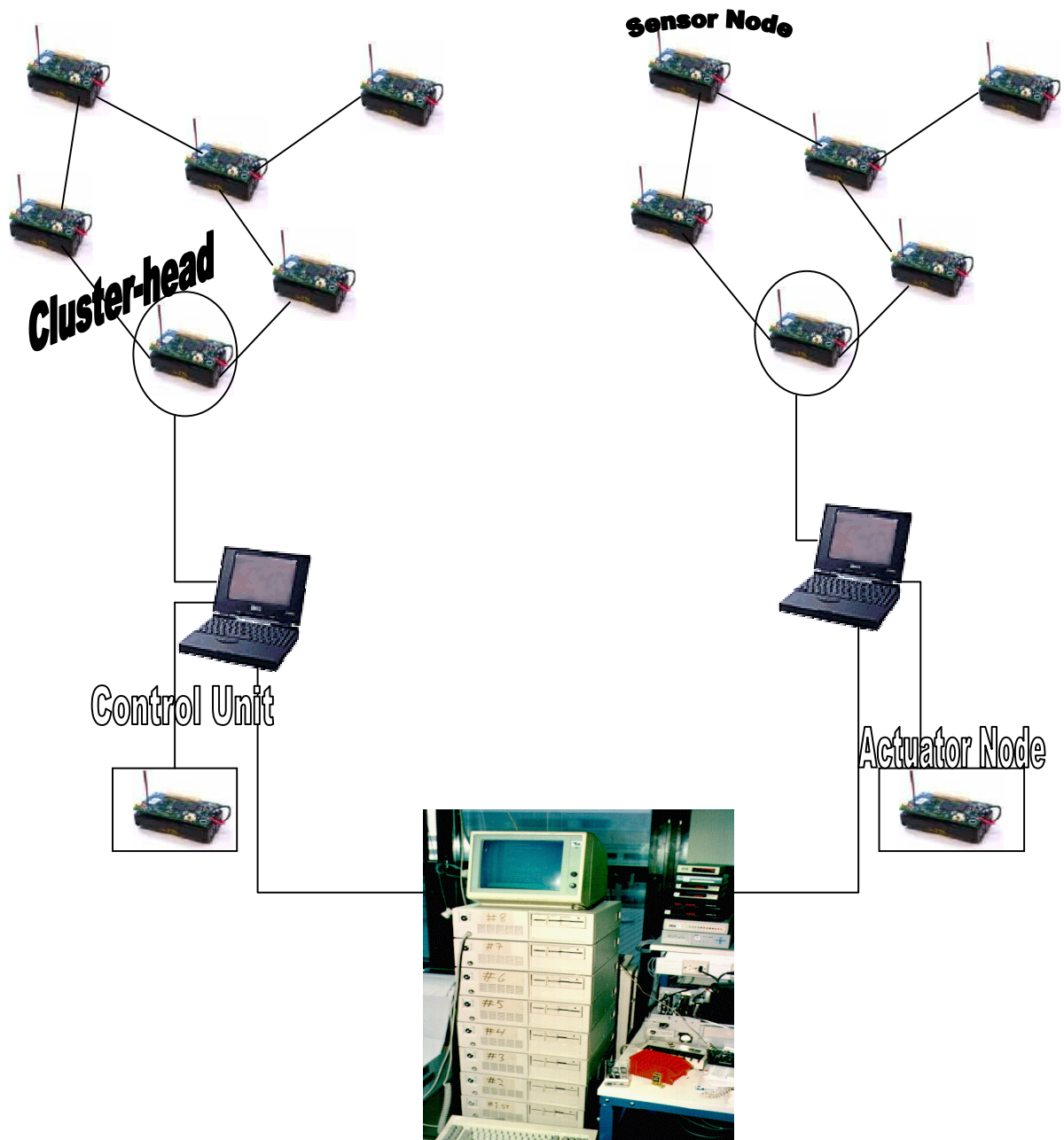
is linked to the wired HVAC architecture. The control unit allows data from the sensor network to be shared across the BMS. The full architecture is depicted in Figure 3.2.

The sensor node computational module [11] is a programmable unit that provides computation, storage and bi-directional communication with other nodes in the system. It interfaces with analog and digital sensors on the sensor module, performs basic signal processing (e.g. simple translation based on calibration data, threshold filtering, or delta encoding), and dispatches the data according to the application's needs. It not just a data logging system, and offers two major advantages: it can communicate easily with the rest of the system and can be retasked even after they are deployed.

Data is sampled both spatially and temporally [7]. In order to meet sufficient lifetime requirements, nodes may operate in a phased manner. It is important to note that these nodes are primarily battery powered and have to use the resources at hand in an energy efficient schedule. Nodes primarily sleep, periodically sample, perform necessary calculations, and then send or relay readings through the network at regular intervals. Data may travel spatially through multiple routes in the cluster, to the cluster-head, which then connects to one of the many control units of the in-house HVAC system.

Redundancy is built into the system at different levels. The spread of the sensor nodes themselves provide redundancy in terms of spatially and temporally correlated data, plus each node has non-volatile storage. This allows the nodes to maintain a cache of the sensor data itself. This guarantees seamless operation in the wake of disconnection. Thus, each node and the cluster-head can operate independently from the others.

Figure 3.2: Integration of Sensor Network into HVAC Control



Users can interact with this system in two different ways. Remote users can have access to the data as it appears on the database of the HVAC control system. The system can setup trend logs and statistical analysis tools for each of the data streams, and treats the data coming from sensor nodes just as it was coming from a legacy peripheral. Remote control of the network is also provided through the Human Machine Interface (HMI) at the database. Administrators for the control system can setup permissions on what is allowed to be changed by remote users. On site users can be allowed a more direct control and have higher access rights.






Chapter 4: HVAC Control Testbed

The architecture in Chapter 3 is based on a deployment that spans over an entire building, just like an HVAC control system. It can be used to sense and control airflow and thermal conditions in each segment of a floor and then tailor the heating/cooling based upon the need of the inhabitants, while at the same time conserving energy. This brings up an interesting question of how we can use such an ad hoc sensor network to help increase energy savings and maximize comfort. In this chapter we describe the setup and implementation of an HVAC system testbed. We discuss the sensor node including its hardware and software design, and the actuator/control node that acts on the data received from the control unit. We describe in detail the design and setup of the Controlled Environment Chamber (CEC) [15], a lab facility that allows us to simulate temperature controlled office space and test various control strategies.

4.1 Sensor Network Node

In our deployment, we are using UC Berkeley/Crossbow[®] motes as the sensor nodes. The latest member of the mote family, called *Mica2* (shown in Table 4.1), uses multiple channels, a choice between 916 or 433 Mhz radio from Chipcon [30] to provide bi-directional communication at 38/19Kbps, an Atmel Atmega128 [31] microcontroller running at 7.3728Mhz, and a considerable amount of non-volatile storage (512KB). Pair of conventional AA batteries provide the necessary voltage source, though other renewable energy sources can be easily used.

Table 4.1: Family of Motes

Mote Type	WeC	Renee	Mica	Mica2	Mica2Dot
					
Microcontroller					
Type	AT90LS8535	Atmega163	Atmega128	Atmega128	Atmega128
CPU Clock (Mhz)	4	4	4	7.3827	4
Program Memory (KB)	8	16	128	128	128
Ram (KB)	0.5	1	4	4	4
UARTs	1	1	2 (only 1 used)	2	2
SPI	1	1	1	1	1
I2C	Software	Software	Software	Hardware	Hardware
Nonvolatile storage					
Chip	24LC256		AT45DB041B		
Size (KB)	32		512		
Radio Communication					
Radio	RFM TR1000			Chipseon CC1000	
Frequency	916 (single freq)			916/433 (multiple channels)	
Radio speed (kbps)	OOK		ASK	FSK	
Transmit Power Control	Programmable resistor potentiometer			Programmable via CC1000 registers	
Encoding	SecDed (software)			Manchester (hardware)	

Small size (approximately 2.0 x 1.5 x 0.5 inches) and wireless communication capabilities allow us to deploy motes in remote locations with minimal interference with the existing habitat.

The UC Berkeley/ Crossbow[®] mote family (Table 1) has evolved over the past four years to offer stable platforms for sensor network deployment and research. There is a strong software base to build applications from, a component-based operating system called TinyOS [6], and a programming language for networked embedded systems called nesC [12]. Due to the open availability of the hardware and software, and a comparable match between the system requirements and the properties of the *Mica2* mote, we chose to use the UC Berkeley platform running TinyOS.

TinyOS allows the setting of low level hardware functionality to achieve low power sleep states. In most cases, the application data rates are small enough to ensure that the sensor nodes can be sleeping most of the time and only periodically sample, compute and communicate. Minimizing power in sleep mode involves turning off the sensors, the radio, and putting the processor into a sleep mode. During sleep, current consumption is measured in μA , and therefore battery life is significantly extended. The result is very low-current draw for the majority of the time, and short duration spikes while processing, receiving, and transmitting data. This method does result in extended battery life; however, due to the current surges, it also reduces specified battery capacity. Battery capacity is generally specified for a nominal current drawn constantly by the manufacturer.

The Chipcon (CC1000) [30] radio on the *Mica2* is a single-chip UHF transceiver specifically designed for low-power and very low-voltage wireless applications. It is

capable of multiple channel operation and hence suited for network protocols that wish to implement frequency hopping. However, the adjacent channel spacing should be greater than 150 KHz to avoid adjacent channel interference. The radio is highly programmable and offers the adjustment of output power levels among other things. It also provides a measurement of the received signal strength, referred to as RSSI, on ADC channel0. This is available to any application through the TinyOS platform that wishes to sample the signal strength. Another important deployment detail is that the radio has an extremely sensitive receiver, which can be interfered with by an adjacent local oscillator from another *Mica2* node. A prescribed distance of 2 feet is needed between any two nodes to avoid local oscillator interference [32].

Our implementation of the application-specific communication protocols is simplified from what is expected of a full-fledged system. Each sensor node acts as a transmit-only device in a single-hop broadcast network. The node samples the temperature data using the temperature sensor on the sensor board (Section 4.2) periodically, creates a bundle of entries (the size of the bundle can be defined as part of the application), and schedules a transmission when it has the required number of entries. The data is received at the control unit, which processes the data and passes it onto BMS. As part of its processing, the control unit may choose to affect an actuation for which it sends a message to the actuation node. We describe the details of the implementation of the actuation node in the Section 4.3.

4.2 Sensor Board

Since our application is only concerned with temperature data, we chose to use the “basic sensorboard” (Figure 4.1) as our sensor peripheral to the sensor node. The sensor board provides two sensors: a light photo resistor (Clairex CL9P4L) [13] and a thermistor (YSI 44006) [23]. These two sensors are sampled using the 10 bit ADCs on the sensor board to read data off of them. Both sensors are highly non-linear which makes calibration hard. The board also has an extensive prototyping area to enable use of other sensors and actuators. The sensor board operates at low duty-cycles and low sampling rates so that power may be conserved as much as possible.

The photoresistor is a variable resistor in a simple voltage divider circuit. The divided voltage is measured by the ADC. It has a startup time of about 10ms and draws 1.235 mA of current. The thermistor provides a resolution of about 0.04C and also has a startup time of about 10ms. In practice there tends to be some noise in the system that brings the accuracy of a measurement to about 0.5 degree Celsius. Since the thermistor curve is not linear, the accuracy decreases at the extremes.

The sensor boards offer changeability, to ensure that readings of the physical phenomenon are accurate to within bounds of the sensor precision. For HVAC applications changeability and accuracy are in fact very important, as the data from the sensors is processed a signal generated based on this data controls airflows and heating/cooling in the air-ducts.

Figure 4.1: Sensor Node and Basic Sensor Board



Use of the sensor board in such a low duty-cycle application has suggested that the startup times of each sensor dominate power consumption. The startup time is the time a sensor must be powered before its reading stabilizes. Many sensor manufacturers assume that the sensor will be turned on once at power-up and then be on indefinitely. They optimize their sensors to perform efficiently at high sample rates. Low power applications, where the platform runs off of batteries that need to last a considerable amount of time require that the sensors turn on and off quickly without drawing too much current. Since changes in temperature in an air mass such as an office space occur over large timescales, the sensors have an extremely low-duty cycle and sampling rate. For our application the sensors are powered for a few milliseconds every one and half minutes. Sensors with long startup times (some even require up to a few seconds before the readings stabilize) require current for a longer period of time, resulting in higher power consumption.

4.3 Actuation Node

The actuator node that we employed was essentially a *Mica2* sensor node without the sensor board peripheral. The task of the actuation node was to receive messages from the control unit and affect the HVAC operation accordingly. The messages relayed the temperature in the simulated office space, which then needed to be passed onto one of the HVAC control modules controlling heating/cooling.

The Atmega128 [31] on the *Mica2* has access to 4 different hardware timers, Timers 0-3. Out of the 4 timers, Timer0 is used by TinyOS to provide a software abstraction of timers to any application that wishes to use it. Timer1 is exclusively used

by the Chipcon radio as part of the RadioTiming module to sync a receiver of a packet to the sender and the SpiByteFifo module uses Timer2 to provide a byte-level abstraction to the radio, shifting bits out to the radio when sending and shifting bits in from the radio when receiving. The only remaining timer is Timer3.

The message received from the control unit corresponds to a processed temperature reading, which needs to be converted to a pulse width modulated (PWM) output to be used by the HVAC control module. We used the unused Timer3 off of the Atmega128 microcontroller, which is routed to pin26 on the 51-pin connector. Timer3 allows accurate program execution timing (event management), wave generation, and signal timing measurement. It allows up to a 16-bit resolution, with relevant hardware interrupts. We used the on chip clock of 7.3728MHz to generate a PWM signal in the Fast PWM mode. The generated frequency can be set according to the following formula:

$$f_{PWM} = f_{clk_I/O} / (N*(1+TOP)),$$

where $f_{clk_I/O}$ is the input frequency to the PWM module, TOP is the maximum value that the timer counter counts to, and N is the prescaler divider, chosen from (1, 8, 64, 256 or 1024). We ran the PWM such that the resolution was set to 8 bits, and hence TOP was 255 and the prescaler set to be 8. That made the PWM output frequency to be 3.6KHz.

The HVAC control module expects an analog value for the temperature input, thus the PWM waveform needs to be converted to an analog signal. We design a simple low pass filter off the board to generate the signal. In a typical PWM signal, the base frequency is fixed, but the pulse width is a variable. The pulse width is directly proportional to the amplitude of the original un-modulated signal. In other words, in a PWM signal, the frequency waveform is a constant while the duty-cycle varies according

to the amplitude of the original signal. To remove higher order harmonics of the base signal it is low-pass filtered, thus eliminating these inherent noise components. The filter cut-off frequency was picked to be 10hz, comfortably 2 orders of magnitude below the input frequency.

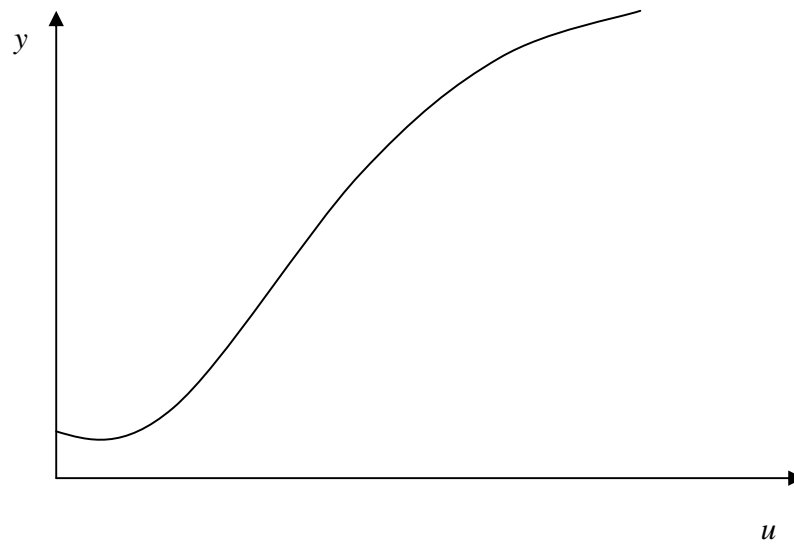
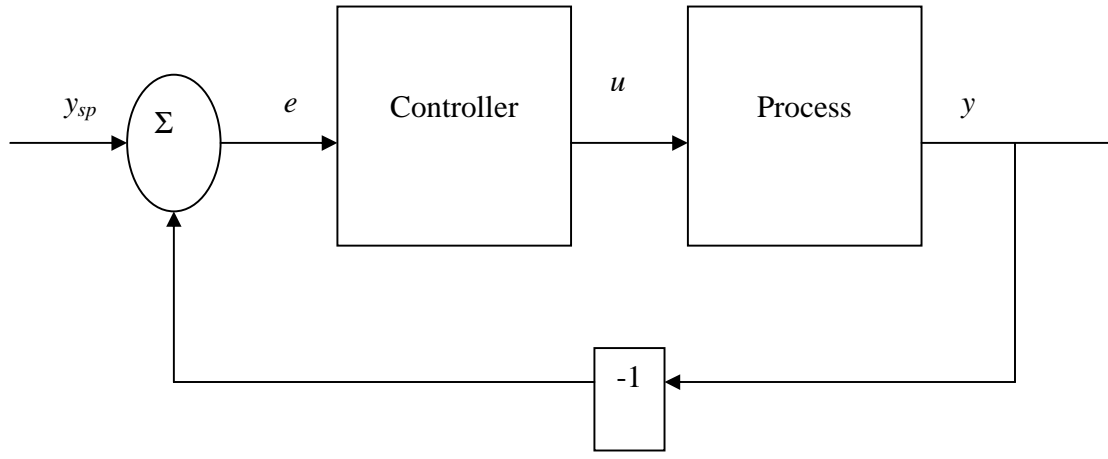
4.4 HVAC Control Module

4.4.1 Process Model

The HVAC control module is part of simple control loop that is the essence of the application. The sensor nodes generate sensor readings that are processed at the control unit and are noted as the process variable (PV) y . This PV is fed back to the system using a closed feedback loop in the form of the actuation node.

The control module has two major components, the process and the controller. The process has one input, the manipulated variable, also called the control variable. It is denoted by u . The desired value of the PV is called the setpoint (SP) or the reference value. It is denoted by y_{sp} . The control error e is the difference between SP and PV, i.e., $e = y_{sp} - y$. The controller has one input the error, and one output, the control variable [18].

Figure 4.2: Process Model: Feedback Loop & Static Process Characteristic



The purpose of the system is to keep the process variable – PV – close to the desired value – SP – in spite of disturbances. This is achieved by the feedback loop, which works as follows. Assume that the system is in equilibrium and that a disturbance occurs so that PV becomes larger than SP. The error is then negative and the controller output decreases, which in turn causes the process output to decrease. If the opposite were to happen, that is, process variable becoming smaller than the setpoint, then an appropriate action would be taken to effectively increase the process output.

Our implementation strives to achieve a closed-loop control, using as input readings from the sensor nodes, which are then processed at the control unit form the process variable, which is fed back into the system through the actuator node. We are interested in looking at the static process characteristic for the experiment, which will give us the steady state relation between the process input signal u and the process output y . The setpoint will be set to a constant value and the corresponding control variable is measured in steady state. Figure 4.2 [18] illustrates our application specific control loop and the static process characteristic for a stable process.

4.4.2 PID Control

The HVAC control module implements the closed-loop control as a PID loop control function block. The block provides Proportional plus Integral plus Derivative (PID) control action of a controlled output, based on a sensed input and control setpoint [1].

Proportional control is typically used for conventional closed loop control systems. With proportional control, a control signal, based on the difference (error term)

between the actual condition (PV) and a desired condition (SP), is produced. The PID control block creates an output signal directly proportional to the error term's magnitude. The assigned *throttling range* controls the relationship between the error and the output. The throttling range is the amount of input change needed to cause the output of the block to go from 0 to 100%.

A characteristic of proportional-only control is that it exhibits an offset (error) condition as the output moves through its throttling range. Because of this, a proportional-only control can never actually maintain the setpoint, except at the output reference position (which is the output value at which the input is equal to the setpoint). The use of *integral* action is designed to eliminate this offset. The integrating term observes how long the error condition exists and sums the error over that time period. The summation generates an additional control signal, which is added to the signal produced by the proportional term. The control loop now produces a control action that allows the elimination of offset over time.

A proportional-integral control can: (1) respond to the presence of error in the control loop; (2) relate the magnitude of the control signal to that of the error; and (3) respond to offset over time to achieve zero error – setpoint [1].

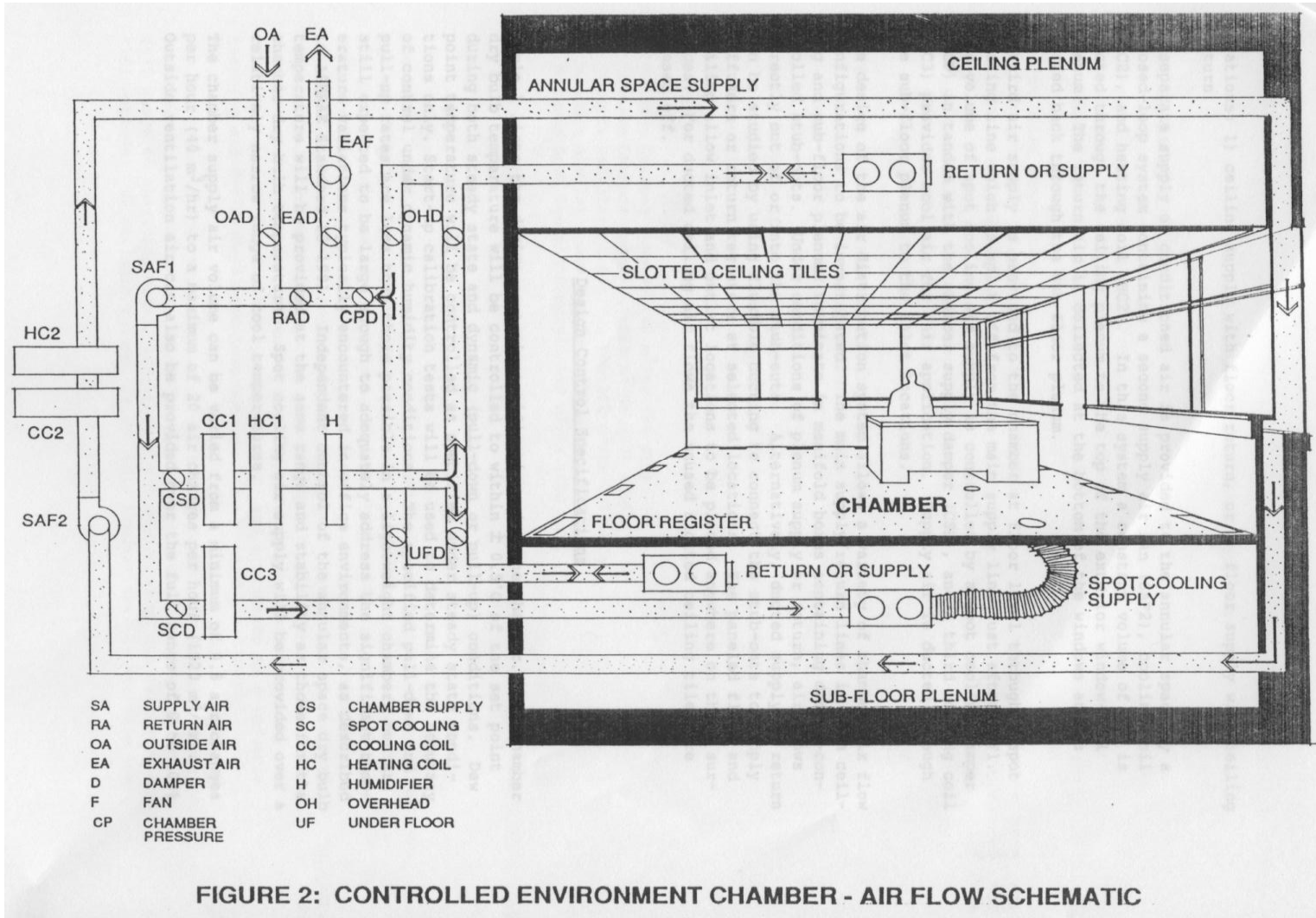
Another process characteristic, *overshoot*, is often present in modulating control loops. *Overshoot* refers to a control loop's tendency to overcompensate for an error condition, causing a new error in the opposite direction. *Derivative* action provides an anticipatory function that exerts a "braking" action on the control loop. The *derivative* term is proportional to the error's rate of change. It is used to observe how fast the actual condition approaches that desired, as a result producing a control action proportional to

that. Thus the derivative action anticipates the proximity of actual and desired conditions, and tried to counteract the control signal produced by the proportional and integral terms. The result is a significant reduction in *overshoot*. Thus, when combined, the proportional, integral and derivative actions provide quick response to error, close adherence to setpoint and control stability [1].

4.5 Controlled Environment Chamber

Controlled environment laboratories provide consistent measurement conditions and the opportunity to precisely define the range of thermal parameters that need to be investigated. The Controlled Environment Chamber (CEC) is designed to resemble a modern office building and to provide a high degree of control over the chamber's thermal environment [15]. A reconfigurable air distribution system allows the CEC to simulate not only the typical high volume uniform air flows found in many laboratories, but also a variety of air flow configurations associated with modern office buildings, including 1) alternate air supply/return locations, 2) spot cooling and heating, as a solution to non-uniform thermal conditions produced by obstructing office layouts (partitions, cubicles, etc.), and local heat sources such as computers, and 3) low-volume ventilation strategies such as displacement ventilation [16].

Figure 4.3: Controlled Environment Chamber



4.5.1 Chamber Design and Setup

The chamber (Figure 4.3), constructed within a larger existing laboratory space, measures 18ft x 18ft x 8ft 4 in, and has a volume of 2700ft³. The raised access floor system consists of 2.0 ft square panels supported by galvanized steel pedestals. Floor registers, fan-powered supply modules, and other spot cooling airflow connections are installed directly into the floor panels permitting maximum flexibility in the selection of supply and return locations. The 2.0 ft high sub-floor area serves as a supply or return plenum, while also providing adequate space for connecting ducted floor registers and running instrumentation, power and communication cables. A 1.5ft high ceiling plenum is provided for similar purposes above the suspended ceiling, made up of 2.0 ft square acoustical ceiling tiles [15].

A plenum wall construction of the two chamber walls facing the exterior allows a continuous stream of temperature-controlled air to pass between the inner (single-pane) and outer (double-pane) glazing of the windows.

For our HVAC testbed, the chamber was setup to have ceiling return and supply. The floor space was divided into 4 sections. Ideally, each section should have been of equal size; because of the main supply duct that runs in the ceiling plenum over the center of the chamber, two sections were made smaller than the others. The measurements for the sections are shown in Table 4.2. The supply locations were located quasi-centrally in each of the four sections, and the return was strategically located over the intersection of the zone partitions in the chamber. The smaller sections had 4ft slot diffusers with 8-inch openings that were connected to the main supply duct using a combination of 6-inch and 8-inch flex duct. The larger sections used 2ft x 2ft square diffusers connected by 6-inch

and 8-inch ducts to the main supply. The return was a 12-inch duct connected to the recirculation system through an opening out of the chamber.

The partitions were put in place to have minimum airflow movement from one section to the other. The space between the partitions and the walls was minimized. As, the partitions went up about 5ft 4in off the floor they had to be extended using foam core boards all the way up to the ceiling.

Each of the diffusers in the sections had to be balanced to have equivalent airflows for each of the partitions. An AirData Flowmeter [17] was used to calibrate the airflow quantities in each section. The device captures and directs the airflow from an outlet across the highly sensitive flow-sensing grid within the device. It also senses the total pressure and static pressure to average a single velocity pressure for calculation of accurate airflow measurement.

Table 4.2: Chamber Specifics

	Size	Air Flow	Walls/Windows
Zone1	11ft x 9ft	80 cfm	Two walls
Zone2	11ft x 9ft	80 cfm	One wall, one window
Zone3	7ft x 9ft	70 cfm	One wall, one window
Zone4	7ft x 9ft	70 cfm	Two windows

Chapter 5: Testbed Results

We deployed 4 motes, one in each partition in the chamber at randomly picked locations at table height. We ran a series of experiments, whereby we controlled the feedback based on one or more sensors. The hypothesis for these experiments was that the use of multiple ad hoc sensors should allow for the implementation of better control strategies. These control strategies aim to provide a two-fold benefit of reduced energy consumption and increased thermal comfort. The experiments ran in two sets with different controller parameters for each. Adding infrared bulbs and planar radiant heaters to two of the sections simulated random but equivalent heat load conditions.

The first set of experiments comprised of four runs, where the process variable was derived from only a single sensor and each run picked a different sensor to the process variable. The second set of experiments included two single sensor runs and a run that used as output readings from all 4 sensors. Results obtained from these two experiments can help corroborate or negate our hypothesis of using a multi-sensor wireless architecture instead of a traditional fixed wired sensor [28][29].

The first set was used, as a process check to make sure that the system indeed would be able to come close to equilibrium. Some of the control parameters were modified to run the second set to ensure a more stable steady state output.

5.1 Experiment Runs

The experiment was divided into two halves. The first run included collecting data from each zone. However, the actuation was based off of the signal generated from only one of the four zones. The controller setpoint was set to be at 25 °C. Random loads were included as part of two zones, namely 1 and 3. Zone 1 had a 200Watt planar radiant heater and laptop computer, which emanates close to 75Watts of energy. Zone 3 had a 250Watt infrared bulb.

We logged in temperature data from each of the four zones, as well as the supply air temperature, return air temperature and the control variable generated by the control loop. For each of the four cases we calculate the energy required to heat. The equation for calculating the amount of energy required is:

$$H = C_p \rho Q \Delta T,$$

where C_p is the specific heat, ρ is density, Q is volume flow rate and ΔT is the difference between the supply and return temperatures. Table 5.1 shows the results from the first set.

Table 5.1: First Run

	Test1	Test2	Test3	Test4
Date & Time	18 th Dec 12-8pm	19 th Dec 12-8pm	20 th Dec 12-8pm	21 st Dec 12-8pm
Command Node	1	2	3	4
ΔT	1.1	2.6	0.8	2.9
H	187.66W	452.09W	136.48W	494.74W
Outside Mean Temp	51.5F	50.5F	53.5F	48.5F
Conditions	Mostly cloudy	Some rain	Some rain	Mostly cloudy

	Test1	Test2	Test3	Test4
Zone1	25.04 ⁰ C	26.90 ⁰ C	24.09 ⁰ C	26.49 ⁰ C
Zone2	23.41 ⁰ C	24.85 ⁰ C	23.57 ⁰ C	25.40 ⁰ C
Zone3	25.99 ⁰ C	26.62 ⁰ C	24.97 ⁰ C	28.01 ⁰ C
Zone4	22.89 ⁰ C	24.01 ⁰ C	22.81 ⁰ C	25.02 ⁰ C

	Values
Specific Heat (C_p)	1004 J/Kg/C
Density (ρ)	1.2 Kg/m ³
Flow rate (Q)	0.1416 m ³ /s
Heat Transferred	$C_p \rho Q \Delta T$

5.1.1 First Set

The experiment runs indicate that zones 1 and 3 run the warmest, keeping in mind outside temperature and conditions, whereas zone 4 runs the coolest. This is what is expected since zones 1 and 3 contain comparable heat sources, while zone 4 has two window edges and hence the additional heat required to keep it warm. Also, because of the heat sources, if the HVAC was controlled based on zones 1 or 3 the energy required would be considerably less. In the next set of runs we investigate how an averaging of the four zones performs.

5.1.2 Second Set

The second set of runs involved tweaking some of the control parameters of the controller such as the throttling range and values for *integral* and *derivative* action. The values from the first set were able to get the system closer to the setpoint quicker, however, the system was not in equilibrium and the control variable saw huge oscillations for small deviations from the setpoint. 100% swings in the control variable were noted for even a 0.2° change in the process variable. This was due to the fact the first set of runs employed just a PI controller with no *derivative* action. Making the integral action lower and derivative action higher rectified this non-ideal behavior.

Table 5.2 shows the comparison of three control strategies. The first two represent actuation based on the warmest and coolest zones respectively, and the third one is a simple averaging strategy where all four zones were taken into consideration. The table shows the amount of energy required to heat using that strategy, number of zones that are comfortable, outside conditions, and the control variable denoted in terms of percentage (%), where 0% implies “full” heating” and 50% implies no heating.

Table 5.2: Second Run

	Coolest	Warmest	Average
Date & Time	23 rd Dec 12-8pm	24 th Dec 12-8pm	25 th Dec 12-8pm
Command Node	4	3	Avg
ΔT	3.11	1.13	3.2
H	530.56W	192.78W	545.92W
Outside Mean Temp	55.4F	55.1F	48.9F
Conditions	Mostly cloudy	Some rain	Cloudy

	Coolest	Warmest	Average
Zone1	27.41 ⁰ C	23.95 ⁰ C	25.42 ⁰ C
Zone2	25.58 ⁰ C	22.39 ⁰ C	24.23 ⁰ C
Zone3	27.80 ⁰ C	24.99 ⁰ C	26.72 ⁰ C
Zone4	25.00 ⁰ C	21.93 ⁰ C	23.59 ⁰ C

From the data for these three days it is evident that the amount of energy required in each case has increased. For instance, the amount to heat the coolest room has gone up by 7%, however the next day's test indicates that the amount required to heat the warmest room has gone up by 41%. The last experiment pertains to controlling based on the average of the 4 zones. Intuitively the average was expected to be somewhere around the midpoint of the first two cases, however this is a clear anomaly.

In order to explain the changes over the last three runs in terms of excessive energy required to heat, we investigated that the building which housed the chamber had indeed shutdown its internal heating starting from the first day of the second run of experiments. This meant that over the next three days, the large air mass was slowly beginning to cool. This affected our data, as the air flowing through the upper and lower plenums, as well as the annulus was no longer being heated by the building's internal heating system.

However, we were still able to note that the averaging strategy allowed the four zones to be comfortable at the same time. This was determined by computing the simulated average number of complaints per zone per year assuming a single-shift, 8-hour workday and 250 work-days per year [34]. We compare the behavior obtained by a combination of the actuation based on the coolest and warmest strategy with that obtained by a strategy using an average temperature in all four zones. The combination of single sensor cases translates to an equally likely probability that any one of them is chosen to be the control point for actuation. Data indicates that the average strategy leads to 25% less number of complaints than a single sensor strategy. Further, tests need to be run to conclusively determine the quantitative benefits of using any multi-sensor scheme.

5.1 Sources of Error

As in any system, there are numerous sources of error or noise that can creep into the variables being computed and operated upon. We try and give a thorough analysis of most of the sources of error and whether or not they can be compensated for. These can be grouped into two categories: precision-related errors (resolution), errors due to non-linear behaviour.

Precision related errors occur due to limited resolution during sampling and exchanging data. For instance, the ADC sampling is strictly 10 bits, hence the temperature data gathered using the on-chip ADC could not be more precise than that. Data is received at the control unit and averaged over, the resulting PWM action is calculated as *double*, however, we chose to send the result as a *short* due to conservation of message space, thus resulting in loss of precision by dropping the decimal places. Even though our application only sent limited data in each packet, in general the packet sizes for sensor network applications are fixed and small. The loss of precision in this case was due to the rounding off of the PWM control signal at the control unit. Also at the actuation node, the 8-bit precision of the PWM restricts the granularity of the change that can be reported by the actuation node. In our case the operating range was indicated to be between 15-27 °C. That amounts to representing a 12-degree span by 256 units, which gives a potential swing of .05 degrees. These cases of error are such that they cannot be compensated for in firmware, as they are inherent to the design decisions of the sensor hardware.

Errors that result from non-linear behaviour occur on the behaviour of the thermistor and the batteries. Both the thermistor and batteries exhibit non-linear behaviour. In the case of the thermistor, each of the sensor boards was hand-calibrated

before the test began and had a different transformation function to convert between ADC reading and the equivalent value on the temperature scale. The batteries themselves add to the compounded noise of the system. Drained out batteries do not put out the expected amount of voltage and hence the ADC counts computed for a particular sensor reading are affected by the battery strength. This however can be compensated for, by adding the sampling of the battery strength as part of the application, and scaling the ADC count value based on the current strength.

5.2 Further Experimentation

Our initial results have allowed us to understand the dynamics of controlling a simulated air mass in a building. Further fine-tuning of the controller can lead to better steady state characteristics, which in essence would translate to more energy conserving operation. All of our experiments thus far ran with an integral part of the HVAC system non-operational, and that being the cooling. Hence, once the setpoint was passed, the system would be in a sensing only mode consuming miniscule amount of power. However, to truly model a real deployment cooling is paramount as it speeds up the process. However, even though we lacked cooling capabilities we can still infer the amount of energy that it would have required to cool, based on the linearity of the heat transfer process.

5.3 Future Motivation

It is important that any further work in developing the technology core in the field of sensor networks is done keeping in mind the potential users of these networks, and what their needs are. The users of these systems are more interested in the quality of the

data that is delivered and the associated benefits that might exist in deploying such a system. Thus, it is very important that both those things be part of the design cycle.

Low-power design should be the mantra at all levels, be it hardware or software. Specially, in a field like BMS, where control and monitoring networks already exist, any change in the status quo will only come about if there are obvious benefits in bringing about the change. Some of the potential questions that users in this field might ask if given a deployment choice would be:

Does it make sensing quantities in buildings easier?

Does it reduce costs for installation of sensors?

Does it allow us to do more than can be done with a wired system?

Also, since in this field, its very important that each component, be it hardware or software be accorded some sort of standard, there is a clear need for conformance to minimal design requirements.

Chapter 6: Related Work

Wireless sensor networks have found numerous applications in the areas of distributed control and monitoring. Some of the notable ones are habitat monitoring, fire monitoring and control, structural monitoring and HVAC controls. It is easy to see why sensor networks can be seamlessly integrated into existing systems in these fields, or even be the first system of choice in some cases (structural monitoring, Golden Gate Bridge [26], etc.). For each of these fields, the architecture and the system design of the sensor networks has been pursued differently, which indicates that these systems are very application specific.

For the purpose of BMS and HVAC control systems, there is already a rich framework of network-based deployment of nodes and peripherals. The only novelty is the addition of an ad hoc distributed sensor network framework that can be very easy to deploy. A lot of the big control companies, such as Siemens and Johnson Controls have already ventured into wireless sensors, however these products have tended to be infrared based requiring line-of-sight. However, new sensor network initiatives such as Ember and Sensicast [27] are developing nodes that are directly pluggable into an existing HVAC system that runs LONWORKS or BACnet. It would be interesting to note whether the operating system for these integratable sensor networks will be something that would be ported from an existing version of HVAC control software or whether individual vendors would integrate some sort of firmware interface between the two systems.

Most of the work with wireless sensor networks for HVAC has focused on just the monitoring aspect of it. Thus the sensor network behaves as an outside entity observing the changing physical phenomenon. An effort at the Mechanical Engineering department at UC Berkeley, termed *Etchnet* is indicative of these works [24]. The sensor network was set up to evaluate the communication performance of low power ad hoc wireless networks in building settings. The site was chosen for a number of reasons, including its similarity to that of an industrial setting, both in structure and in contents. Most relevant is the presence of large operational machinery, such as machine tools. An important observation made as part of the study was the quality of the wireless links. It was noted that due to the presence of heavy machinery there was enough interference created to cause irregular packet drops. In our testbed, this phenomenon was not witnessed due to small area in which the network was deployed.

The only known work that involves actual control of the HVAC system as well is at the Pacific Northwest National Laboratory [22]. Their deployment consists of two different buildings, about 5 floors inhabited by personnel. Their control strategy was something similar to averaging temperature from individual zones to reset the temperature of the chilled water. Their architecture consists of battery powered temperature sensors and transmitters, which include a 900MHz radio operating in a frequency hopping mode with an effective range of 2500ft. Also, included as part of the design are line-powered repeaters, and receivers that translate from a wireless to an N2 bus that links to the building automation system.

As part of this effort the PNNL project also plans to install wireless sensor networks to help validate and document that the lighting systems are successfully

performing their monthly and annual tests of emergency egress lighting ballasts. Also, as part of their efforts, is an experiment to turn off the HVAC when doors and window outlets are open. This strategy can greatly reduce the amount of tempered air that escapes the building.

The motivation for our work was a simulation study done by Lin *et al* [29]. They showed that it is indeed beneficial to use multiple sensors to aggregate an output instead of a single sensor to control an HVAC system. However, in most office spaces, the placement of the thermostat is not ideal, in fact, any two thermostats are too far apart to be able to monitor adjacent areas that require heating and cooling at the same time. As a result, some areas are either left too warm or too cold, plus, added energy is wasted in doing so.

Chapter 7: Conclusion

We have demonstrated how wireless sensor and actuators can be integrated into the existing BMS framework. An integratable architecture that can be combined with legacy systems will allow deployment of future systems, provided that power for perimeter sensor nodes can be scavenged, or battery life extended into multiple years, meaning that these networks will have to be rigorously energy efficient [35].

Our results are preliminary and further experimentation is required to successfully validate our hypothesis of the benefit of using an ad hoc multiple sensor architecture that provides a two-fold benefit of higher energy conservation and added thermal comfort. Future experiments will monitor actual building spaces with inhabitants in order to get substantial empirical data and strengthen further claims. One thing that is certain is that deployment is easier than ever, and in time when things get streamlined and the technology matures sensing any physical attribute would be as simple as plug-n-play.

Bibliography

- [1] Barber-Colman Company LCM/GCM Programmer's Manual. F23135, LOOP-1, 1989.
- [2] Alan Mainwaring, Joseph Polastre, Robert Szewczyk, David Culler, and John Anderson. Wireless sensor networks for habitat monitoring. In *Proceedings of the ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, Atlanta, GA, September 2002.
- [3] Kris Pister, Barbara Hohlt, Jaein Jeong, Lance Doherty, and J.P. Vainio. Ivy: A sensor network infrastructure for the University of California at Berkeley, College of Engineering. <http://www-bsac.eecs.berkeley.edu/projects/ivy/>
- [4] Slobodan Simic and Shankar Sastry. Distributed environmental monitoring using random sensor networks. In *Proceedings of Workshop on Information Processing in Sensor Networks*, Palo Alto, CA, April 2003.
- [5] Bruno Sinopoli, Cory Sharp, Luca Schenato, Shawn Schaffert, and Shankar Sastry. Distributed control applications within sensor networks. *Proceedings of the IEEE*, November 2003.
- [6] University of California, Berkeley. TinyOS: A component-based OS for the networked sensor regime. <http://webs.cs.berkeley.edu>, 2003.
- [7] Brent West, Paul flikkema, Thomas Sisk, and George Koch. Wireless sensor networks for dense spatio-temporal monitoring of the environment: A case for integrated circuit, system and network design. In *Proceedings of IEEE CAS Workshop on Wireless Communications and Networking*, Notre Dame, IN, August 2001.
- [8] Julius Kusuma, Lance Doherty, and Kannan Ramachandran. Distributed compression for wireless sensor networks. In *Proceedings of ICIP 2001*, Thessalonika, Greece, October 2001.
- [9] Joseph M. Kahn, Randy H. Katx, and Kris S. J. Pister. Emerging challenges: Mobile networking for smart dust. *Journal of Communications and Networks*, 2(3), September 2000.
- [10] Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, and Kris Pister. System architecture directions for networked sensors. In *Proceedings of the International Conference on Architectural Support for Programming Languages and Operating Systems*, Boston, MA, USA, November 2000.
- [11] Jason Hill and David Culler. Mica: a wireless platform for deeply embedded networks. *IEEE Micro*, 22(6):12024, November/December 2002.
- [12] David Gay, Phil Levis, Rob von Behren, Matt Welsh, Eric Brewer, and David Culler. The nesC language: A holistic approach to networked embedded systems. In *Proceedings of the ACM Conference on Programming Language Design and Implementation*, June 2003.
- [13] Clairex Technologies, Inc. CL94PL epoxy-encapsulated photoconductor. http://www.clairex.com/datasheets/cl9p_series.pdf, July 2001.

- [14] Alberto Cerpa, Jeremy Elson, Deborah Estrin, Michael Hamilton, and Jerry Zhao. Habitat monitoring: Application driver for wireless communications technology. In *2001 ACM SIGCOMM Workshop on Data Communications in Latin America and the Caribbean*, San Jose, Costa Rica, April 2001.
- [15] Fred Bauman and Ed A. Arens. The Development of a Controlled Environment Chamber for the Physical and Subjective Assessment of Human Comfort in Office Buildings. In *A New Frontier: Environments for Innovation, Proceedings: International Symposium on Advanced Comfort Systems for the Work Environment*, W. Kroner ed. Troy, NY; Center for Architectural Research, 1988.
- [16] Fred Bauman, *et al.* Air Movement, Ventilation, and Comfort in a Partitioned Office Space. In *ASHRAE Transactions*, v. 98, part 1, 1992.
- [17] Shortridge Instruments, Inc. Backpressure Compensated Air Balance System: Operating Instructions, July 1993.
- [18] Karl J. Astrom and Tore Haggglund. PID Controllers: Theory, Design, and Tuning. 2nd ed. The International Society for Measurement and Control.
- [19] Michael R. Tennefoss. Echelon White Paper: Implementing Open, Interoperable Building Control Systems. Echelon Corporation, Palo Alta, CA.
- [20] David Fisher. BACnet[®] and LonWorks[®]: A White Paper. PolarSoft[®] Inc. July 1996.
- [21] The ATLAS Project. Building Management Systems & Controls – Introduction. http://europa.eu.int/comm/energy_transport/atlas/htmlu/manint.html
- [22] Ron Underhill. Wireless Building Control. http://www.pnl.gov/conserved-energy/energy_tech.stm, Pacific Northwest National Laboratory, October 2003.
- [23] YSI Inc. YSI 44000 Series Epoxy-Encapsulated Thermistors. <http://www.ysi.com>
- [24] The Millennium Project. Section 10.1.3 Etchnet. http://www.citris.berkeley.edu/Annual_Report/Entire_AnnualReport/sections/sec10.pdf
- [25] Nicholas Sitar, George Brimhall, Steve Glaser, John Radke, Raja Sengupta, David M. Doolin, Anshuman Sharma, and Min Chen. Firebug: Wildland fire monitoring system. <http://firebug.sourceforge.net/>
- [26] David Culler, James Demmel, Gregory Fenves, Sukun Kim, Thomas Oberheim, and Shamim Pakzad. Structural Health Monitoring of the Golden Gate Bridge. <http://www.eecs.berkeley.edu/~binetude/ggb/>
- [27] Sensicast Systems Inc. http://www.sensicast.com/news_events/news_events-detail-091503.php
- [28] R. Martin, C. C. Federspiel, and D. M. Auslander. Responding to Thermal Sensation Complaints in Buildings. In *ASHRAE Transactions*, 112(1).
- [29] Craig Lin, Clifford. C. Federspiel, and David M. Auslander. Multi-sensor Single-Actuator Control of HVAC Systems. In *International Conference for Enhanced Building Operations*, Austin TX, October 14-18, 2002.
- [30] Chipcon, Inc. Single Chip Very Low Power RF Transceiver. http://www.chipcon.com/files/CC1000_Data_Sheet_2_1.pdf
- [31] Atmel, Inc. 8-bit AVR Microcontroller with 128K Bytes In-System Programmable Flash: Atmega128L. http://www.atmel.com/dyn/resources/prod_documents/2467S.pdf

- [32] Crossbow Technologies, Inc. MPR/MIB Mote Hardware Users Manual. http://www.xbow.com/Support/Support_pdf_files/MPR-MIB_Series_User_Manual_7430-0021-04_A.pdf
- [33] Samuel Madden, Wei Hong, Joseph Hellerstein and Kyle Stanek. TinyDB: a query processing system for extracting information from a network of TinyOS sensors. <http://telegraph.cs.berkeley.edu/tinydb>
- [34] Clifford C. Federspiel, Rodney Martin, and Hannah Yan. Re-Calibration of the Complaint Prediction Model. (Accepted) In *International Journal of HVAC&R Research*.
- [35] Ferro Solutions, Inc. Millennial Net and Ferro Solutions Join Forces to Deliver Battery-Free Wireless Sensor Networks. <http://www.ferrosi.com/pr1.htm>