# A Sensor Suite for Smart Buildings*

X
y
z

## ABSTRACT

Small footprint wireless sensors enable many applications that help make a building smart as well as energy efficient at a lower cost. In this paper, we present a repertoire of resources which can be used to build smart building energy management systems. Specifically, we propose the design as well as an implementation of generic wireless nodes like actuators, controllers, sensors which are easy to deploy, configure, scale and maintain. An application built from these can be used for various purposes like retrofitting BMSs in existing buildings, data collection for analysis, or in home automation systems. We have experimented with sensors of various types and describe their evolution. Their deployment has demonstrated their high ease of use, large versatility, and long lifetime.[1]

## CCS CONCEPTS

•**Computer systems organization** → **Embedded systems;** *Redundancy;* Robotics; •**Networks** → Network reliability;

## KEYWORDS

ACM proceedings, LaTeX, text tagging

**ACM Reference format:**
X. 1997. A Sensor Suite for Smart Buildings. In *Proceedings of ACM Woodstock conference, El Paso, Texas USA, July 1997 (WOODSTOCK'97),* 7 pages.
DOI: 10.475/123_4

## 1 INTRODUCTION

Small footprint Wireless Sensor Nodes are evolving technologies nowadays. One of advantages of the wireless sensor network is their ability to bridge the gap between the physical and logical worlds with an ease, by gathering certain meaningful information from the physical world and communicating that information to the more powerful logical system that can process it. Inexpensive, low power, smart devices with multiple sensors provide opportunities for instrumenting, monitoring and controlling the environment. These sensors have capability of sensing, processing and transmitting data over a wireless network. In the near future, wireless sensor nodes will be produced in large quantities at very low cost and densely deploy to improve the robustness and reliability of the system.

*Produces the permission block, and copyright information
[1]This is an abstract footnote



The motivation to make a modular sensor suit comes to the mind when albeit the advancements in the wireless sensor networks nodes, there are very few practical examples of such sensors and actuators have been deployed in the developing countries like India. So the goal of the paper is to make an sensor suit which can be designed, prototyped, developed and deployed easily at the same time maintaining the aesthetics of the building and keeping the cost as low as possible. The possibilities and challenges offered by this field both in theory and in practice are widely recognized and many research teams and companies are active in the design and implementation of units that encompass attributes such as *sensing, processing, communication* and *actuation*. These attribute of the nodes can be further used to monitor and reduce energy consumption of the building.

**Principle of Sensor Suit Design**
*"What type of hardware and software should be used to design and program the sensor nodes such that same node can be deployed in different environments(Home, Office, Labs)?"*

### 1.1 Related Work

Researchers have developed a system that is designed to work in the similar area. Wireless sensor node which uses ZigBee as a communication module and protocol. Power management capabilities built into the protocol stack, which is very suitable for battery power applications. There is a coverage limitation of 10 m which is fine if the system has to be deployed in a small space else lots of repeater needs to be added. Options of selecting a frequency of operations, as some controlled environments can also use this. Mesh network topology supported which helps in developing robust systems. There is a security concern with Zigbee. Maintenance and replacement of a failure instrument will cause a lot. It cannot be interfaced with LAN or WiFi infrastructure that is already available in the buildings and follows complex license policy. Similarly, Berkeley's Mote Platform which runs RTOS(TinyOS) is supposed to run for years on battery power. Scheduling algorithms with hard deadlines can utilize this hardware. This also used to feature multichannel radio transceiver. This product was discontinued because of unknown reasons. There is very less support available for this device and programming this device need expertise in very low-level programming languages. BLE Bluetooth low energy also being a part of research as it can stay live for a year on a coin cell battery. Lots of wearables use it as it can easily get connected with smart mobile phones. There are lots of limitation to use of BLE. A limited number of devices can be attached and supports only star network topology. It 's hard to scale BLE to a level of sensors that can be deployed in the building as there are lots of obstruction present inside the building like walls, furniture, etc. It requires middleware i.e. extra hardware, to be set up so that system can be scaled to a level of a floor of a building.

Looking at all possible technologies and systems that are build using these are very particular application oriented. We could not found a generic solution to the problems faced in the building or help research to gather data from the building. Solutions that are cheap to build, easy to deploy, low maintain and preserves the indoor aesthetics of the building.

## 1.2 Details about next sections

Which sections describes what

# 2 MODULES

This section explains hardware as well as software part of the system. We developed the system as portable as possible. The main aim here is to use hardware modules which can be deployed for an existing building and software tools which are open source to reduce the licensing cost. They are described as follows:

## 2.1 Hardware

Sensor suit is designed in such a way that different set of dumb hardware modules communicate with each other to make the environment smart i.e it takes its own decisions such as whether to turn the Lights and Fans ON/OFF depending upon the occupancy data and temperature data. To realize this we divided sensor suit in three general type of nodes:

(1) Controller Nodes
(2) Sensor Nodes
(3) Actuator Nodes

The hardware architecture of each node is briefly described in the following segments:

*2.1.1 Controller Node:* Controller node can be considered as *Central Node* in our sensor suit as it handles entire wireless communication as well as data processing part. Therefore, the hardware for the node must be powerful enough to handle multiprocessing, wireless communication. Apart from these, it should have low cost and small hardware footprint, so that it does not hamper the aesthetics of the buildings and increases the portability of the node. Table **??** shows the comparison between different controllers.

Hence, we chose Raspberry Pi 3 Model B as our controller node. The $ 35 Raspberry Pi 3 has high processing power (A 1.2GHz 64-bit quad-core ARMv8 CPU), inbuilt 802.11n Wireless LAN and Bluetooth Low Energy (which means, no external hardware) with dimensions equivalent to a credit card. These properties make Raspberry Pi suitable for our controller node hardware.

The raspberry pi runs multiple threads. Some threads initiate the system on wake up/reboot, others listen to the data transmitted by the sensor nodes, process on that data and then transmit the decisions such as ON/OFF to the actuator nodes over WiFi. The node stores the temperature sensor data in a file and then sends it to the server side over LAN. This is used for *Thermal Profiling* explained in later sections. As the entire process of collecting data from sensors and transmitting control signals based on that data to the actuators is wireless, this makes the controller node portable and allows the user to place it anywhere in the building as long as it is connected to the same wireless network in which sensor nodes and actuator nodes are connected.

*2.1.2 Sensor Nodes:* We believe that to make appropriate control decisions to increase the system's reliability; we should have ample amount of data of the environmental parameter such as temperature, humidity, occupancy. Therefore, we must sense these parameters from different locations of the building. To achieve this, we designed the *Sensor Nodes* which a) accurately measure above discussed parameters, b) can be deployed in remote locations, c) powered using battery d) have hardware footprint as small as possible. There sensor nodes are divided into two parts a) *Master Sensor Nodes* and b) *Slave Sensor Nodes*.

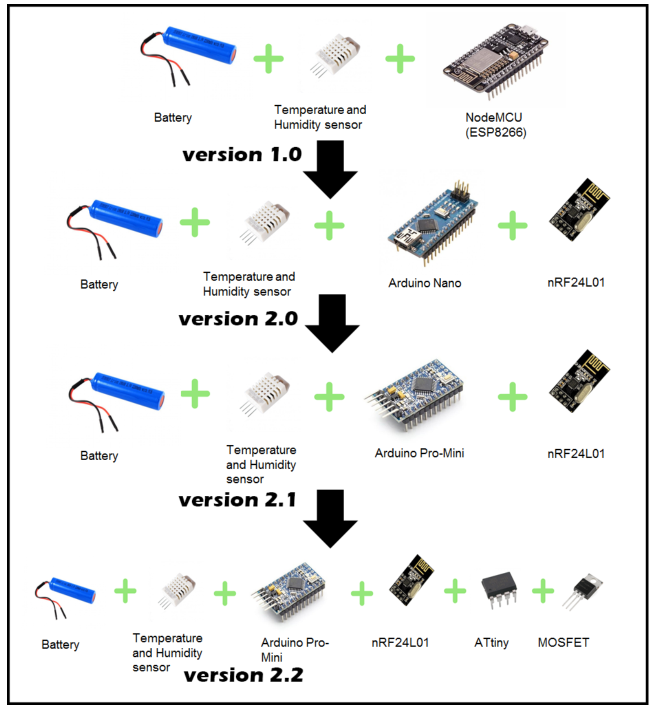


**Figure 1: Sensor Node Versions**

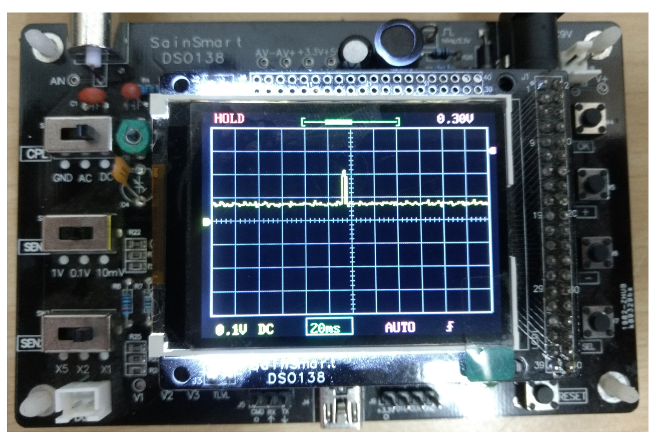


**Figure 2: NodeMCU v1.0 Current Consumption**

**Table 1: Sensor Node Properties**

| Versions | Transmit Mode Current | Wake Time | Sleep Mode Current | Estimated Battery Life | Observed Battery Life |
|---|---|---|---|---|---|
| **NodeMCU v1.0** | 200 mA | 5 sec | 2.7 mA | 13 days | 6 days |
| **Ardiuno Nano v2.0** | 30 mA | 0.5 sec | 4 mA | 30 days | 20 days |
| **Ardiuno Pro Mini v2.1** | 30 mA | 0.5 sec | 0.1 mA | 240 days | 30 days and counting |
| **Ardiuno pro mini+ATtiny+MOSFET v2.2** | 31 mA | 0.5 sec | 0.005 mA | 315 days | N/A |

**Table 2: Sensor Types**

| Sensor | Parameter | Interface | Range | Accuracy | Operating Voltage | Active Current | Cost |
|---|---|---|---|---|---|---|---|
| DS18B20 | Temperature | One Wire | -55°C to +125°C | ±0.5% | +3.0 to +5.5 v | 1.5mA | $3.95 |
| DHT22 | Temperature & Humidity | Digital Pin | T:-40°C to +80°C H:0-100% RH | T:±0.5% H:±2% | 3.3v-6v | 2.5mA | $9.95 |
| Pololu PIR | Motion | Digital Pin | 7m & <120° | N/A | 5v-20v | 65mA | $9.95 |
| Panasonic PIR | Motion | Digital Pin | 5m | N/A | 2.3v-4v | 100$\mu$ A | $28 |

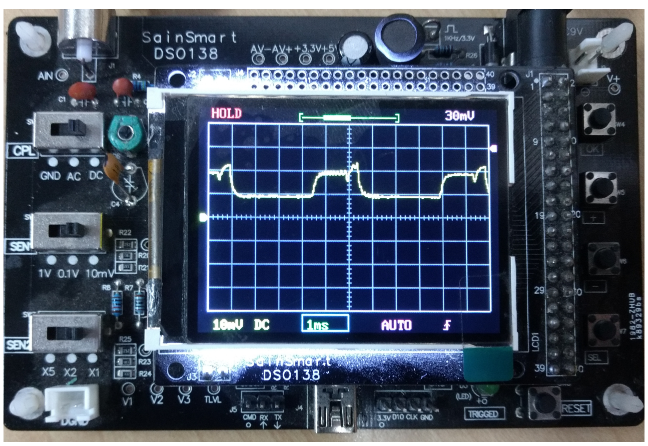


**Figure 3: Arduino Pro Mini v2.1 Current Consumption**

*Master Sensor Node* acts as a gateway, collecting the data from all the slave sensor nodes and transmitting it to the controller node over WiFi. The hardware architecture of this node is this, Arduino Pro Mini as Controller, nRF24L01+ RF communication modules to communicate with the slave sensor nodes and ESP8266 WiFi module to transmit data to the controller node. The RF module is connected to the arduino over *Serial Peripheral Interface*(SPI) whereas EP8266 module communicates with the controller using UART bus. The master sensor node has to be powered using DC Power Adapter.

*Slave Sensor Nodes* 4 are low energy consuming battery powered nodes which are deployed across the building to read the different parameters accurately and then send it to the master sensor node. This node consists of Arduino Pro Mini as a controller, nRF24L01+ RF communication module, and a sensor. Depending on the parameter, this sensor could be a temperature sensor or motion detector sensor. Dallas Semiconductor's *DS18B20 One Wire* temperature sensor is used to sense the temperature of the area. The advantages of this sensor are that multiple such sensors can be connected to the controller using just single wire bus which reduces the hardware footprint significantly. On the other hand, *Passive Infrared (PIR)* sensors are used to detect the motion inside the room. The slave sensor nodes consume 30 mA while transmitting data and 0.1 mA while in sleep mode. Therefore, estimated life of the nodes with 2200 mAH battery at 1-minute data transmitting frequency is almost eight months!


**Figure 4: Slave Sensor Node**

*2.1.3 Actuator Node.* The last element in the sensor unit hardware is an Actuator Node. As the name suggests, the actuator nodes are used to control the appliances like Fans, Lights, ACs. We have designed a couple of hardware architectures to regulate a different set of devices. Fans and lights are controlled with the *Relays* whereas the high power devices such as ACs are turned ON-OFF using *Infrared LED* blasters. Therefore, the actuator nodes at relays are controlled with a nodeMCU 6. The reason using a nodeMCU for the relay is that nodeMCU can directly take actuation signals from the controller node over the WiFi.

The actuator nodes employed for ACs use Arduino Pro Mini as controller along with the Infrared LEDs to send the control signal 5. This setup uses nRF24L01+ RF communication module to receive data from the controller. The main reason to use Arduino Pro Mini with nRF24L01+ RF module, as explained in the earlier section, is that the power consumption of the node is very low as compared to the nodeMCU and hence, the node can be placed in remote locations using a battery.

**Figure 5: Air Conditioner Actuator Node**

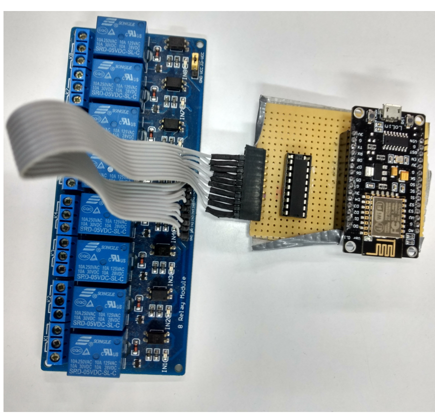

**Figure 6: Lights & Fans Actuation Nodes**

## 2.2 Software

*2.2.1 Node-Red middle-ware utility:* As the above section which mentions about hardware modules; Modular design of the hardware needs a utility software which has features to test and debug every module separately before deployment. Node-red gives a visual interface that can be used to do the same. How this visual tool can be used beyond testing and debugging? As building have different design and architecture, generic solution needs some tweaks to be made in order to suit the infrastructure. This tool allows us to make those changes visually at the application level architecture which gives a flavor of customization to any system that is build. This is not well developed to an extend were a layman can use it. But it is very useful during for testing, deployment and monitoring purposes. (data collection direct support for data logging , json,)

*2.2.2 Various Analysis and Decision making module:* Do we need an Intelligent decision making system? Isn't it sufficient to actuate a module depending on some sensed value? So the answer is, it is not because one type of sensor is not enough to give you 100% correct results. Consider an example of a camera being used for image processing and trying to answer a question like, is anyone there in a room? A system has to answer yes or no in 30 sec. There is high chance that a camera is not able to capture the complete room. For e.g. certain blind spots are left, or image processing has given a false positive. In such a case the system will take action which may lead to wastage of energy. Considering this if we use an intelligent decision-making system which combines the output of various sensors before making a decision. We can improve the accuracy of the complete system. (concept of inference engine from observability)

*2.2.3 Modular and reconfigurable programming code for hardware:* This hardware has a combination of software running; one is the firmware, and other is the application software. We chose lite firmware(nodeMCU) and programming language(Lua), so that battery power applications are possible and get all the possible flexibility at the same time. The reconfigurable feature allows us to make changes in the network configuration setting of the hardware module so that any changes can be done remotely as an when required. As the hardware is open-source and easily available and very cheap compared to customized very accurate sensors; accuracy, a latency of sensing and actuation is poor. Such device needs to be programmed very carefully and intelligently so that it can perform the same is costlier versions of the same. For example, consider PIR sensor, it takes the time to stabilize after turning ON. So the nodeMCU which is attached to it must be programmed in a way such that it gives proper time for PIR to settle and sense the fluctuation in infrared radiation correctly and send it over wireless.

As the hardware and software are modular we need a glue to stick the system together. Software at various module are communicating state maintain messages so that complete system is on the same page. Hardware is talking to each other over MQTT a communication protocol which is explained thoroughly in the next section.

## 2.3 Communication

Communication is a very powerful entity of all as it connects the dots make the complete picture clear. Every independent module is equipt with some specialized hardware specifically for communication. This specific hardware which is common in all the entities gives uniformity to the system. All the modules can talk the same language to each other. And the protocol used by it id MQTT. Very efficient protocol when it comes to the building perspective. MQTT follows a topic format to address the entity in the network. Sensors and Actuator nodes are efficiently addressed using this topic format. An address like "A/2/205/temp/ID" can be used to pinpoint a temperature sensor in Building A at second floor in room no. 205, type of sensor and its unique ID. Looking at the topic name you can pinpoint the location of the sensor of the actuators. This protocol is suitable for group broadcasting; you can send msg to all second-floor sensors to stop sending data when you don't want it. You can start all the lights of a first and third floor just by sending one command. The third party entity in MQTT is a broker, who is a centralized entity for communication between the publishing client and subscribing client. A broker can be installed on any Linux box that is why we used a local and global broker in our system. Local broker's scope is related to the particular entity and various modules of the same application. And global broker takes care of the data logging part, upgrade part. Push-pull mechanisms are explored around so that application can have good performance in communication. As MQTT runs on TCP/IP makes is more reliable. Apart from this it also provides quality of service guarantee at three different levels.

MQTT in conjunction with node-red can give a great picture of the system together. Mqtt helps to generate a separate stream of data on demand. Which can be used for data analysis purposes.

## 2.4 Network technology(WiFi)

Why to chose wireless over the wired system? Is it necessary? How much difference does it make? We will try to answer all these questions in this section. When it comes to installation of any Building management system or just the sensors various aspects come in mind. The system must be economical, efficient enough to pay back in at least one and a half year, etc. Installation of wireless sensor system will cost approximately 30% less compared to wired sensor installation which includes the cost of devices and labor charges. And maintenance will make it a lot worst. The wireless system will prove to be a lot better as the system is scaled up.

Over the cost, it provides a lot of technical advantages like mobility, distributed architecture modularity re-usability compactness adaptable, scalable. Let us see each one of them one by one. Things need not fixed at one place. It is not necessary to hardwire controller to the actuators. It can be kept anywhere as far as it is in the vicinity of the building wireless network. Wireless nature itself gives it a flavor of distributed architecture. Every module has a small controller which is sufficient enough to hold the state if there is any failure in other parts of the system. In such distributed system we can eliminate complete system failure due to one point of failure. It is very easy to identify and replace malfunctioning modules. A failure module can be instantly replaced with a similar non-important module with little reconfiguration over the visual interface. Lots of physical infrastructures reduce as we eliminate wires for long distance communication. Redundant use of a high-end controller which are underutilized is replaced by a little controller which performs the same task by taking command from one central commanding node. All these small things make a greater effect on the footprint of the hardware adding to the compactness to the system. The wireless system is more adaptable because of the use of mobile communication and INTERNET over WiFi, a lot of infrastructures is already available in place to use in offices, houses, school, etc. How wireless makes it easy to scale? If a wired system needs to be scaled from room to a building or group of buildings, technicians have to pull the wires from each place setting up everything near the controller. They need to keep track of which wire is attached to which modules. It is very time taking and painful, and also needs a lot of labors with skills. What if all this can be avoided? We can deploy system four times faster requires less number of people with a moderate skill set. All this is possible if we go wireless.

# 3 APPLICATIONS AND TESTBEDS

Here, we report the performance of our wireless sensor suit when deployed in the different scenarios such as classroom, labs, etc.,. We have upgraded our Wired *Smart Classroom Complex* to a wireless system and implementation of wireless sensor suit, a comparison between wired and wireless systems as well as challenges faced while deploying the sensor are documented in the following sections.

## 3.1 Smart Classroom Complex(SCC)

The Smart Classroom Complex is designed with the goal of ensuring a) zero occupancy implies zero consumption and b) minimal power consumption in an occupied room. Compared to the traditional classrooms, SCC collects user activity pattern by physically sensing their movement and then controlling particular devices (Fans, Lights, HVACs) based on sensed information. If occupancy is detected, lights and fans are turned ON and subsequently, fans and HVACs are turned on based on temperature. If room is detected to be vacant , all appliances are switched OFF. This is explained using a state diagram 7

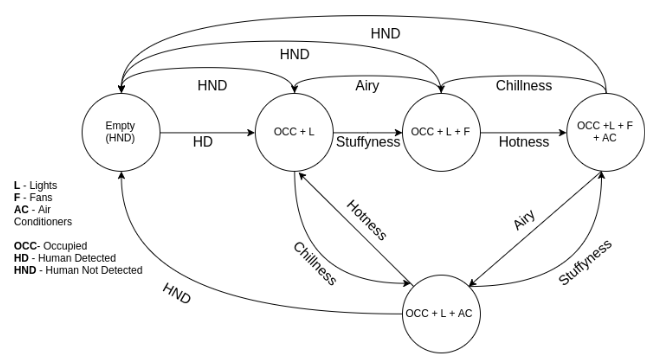


**Figure 7: SCC State Diagram**

Development till the current SCC hardware is explained below:

In the classrooms in our department building, lights, fans and all HVACs were switched ON for 5 hours and 30 minutes i.e. 8:30 AM to 12:30 PM and 2:00 PM to 3:30 PM. A designated person from the building office is expected to turn OFF all the appliances during the 1.5 hour break. But, many a time we observe that this job was never done and the appliances were ON for entire 7 hours. Hence, we come up with the *Smart Classroom Complex*.

Smart Classroom Complex was initially employed with wired sensors. The actuating elements such as relays for fans and lights and IR LEDs for ACs were directly connected to the pins of the Arduino Mega which used to receive control signals (ON-OFF) from the controller node Raspberry Pi. Temperature sensors used for temperature profiling of the classroom was controlled using One Wire Interface provided by the raspberry pi. Also, the raspberry pi used to monitor the state of PIR for motion detection.

As all the sensors and actuator were directly connected to the raspberry pi, it is difficult to install the system in any new area. A failure in any part is very difficult to detect as well as address. This also increased the maintenance cost as even to change a single part of the system, entire process had to be stopped. As we were facing before mentioned problems, the system was not easily scalable in different areas.

To tackle these shortcomings and make system portable, we come up with the *Wireless Sensor Suit*. As mentioned in section 2.1, the system has three nodes a) *Controller Node*, b) *Sensing Node* and c) *Actuator Node* completely independent from each other. All three nodes are connected on existing WiFi. Hence, no extra hardware is required to install the current wireless system.

The protocol used for the communicating is MQTT for reasons explained in section 2.3. The remote nodes such as AC actuator nodes or temperature sensing nodes send or receive data using Nordic's nRF24L01+ RF communication chip. These nRF24L01+ chips are connected to an Arduino Pro Mini using *Serial Peripheral Interface*. The main motto for using Arduino with nRF24L01+ is that, we wanted the remote nodes to be battery powered, which is not possible with the power hungry WiFi modules. The power consumption of Arduino Pro Mini and nRF24L01+ while communicating is upto 30 mA and in sleep mode it is 0.1 mA. Hence, this nodes can be battery powered for long duration.

Removing the wires from the system provided us with additional benefits, including greater flexibility on where controls can be placed, and significant savings in installation by avoiding the expense and disruption of wiring.

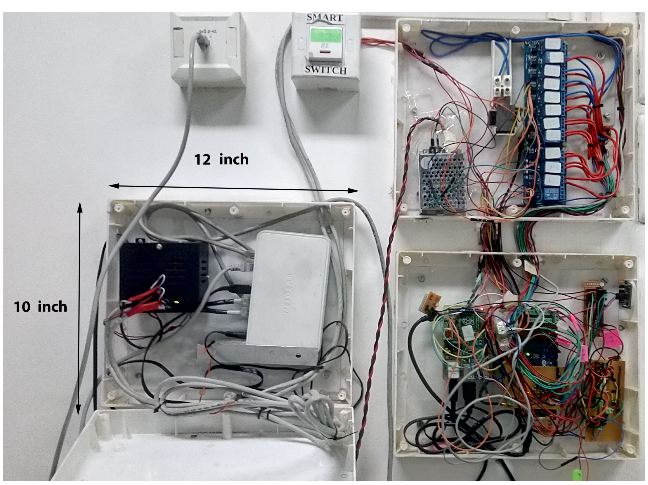


**Figure 8: SCC 1$^{st}$ Version**

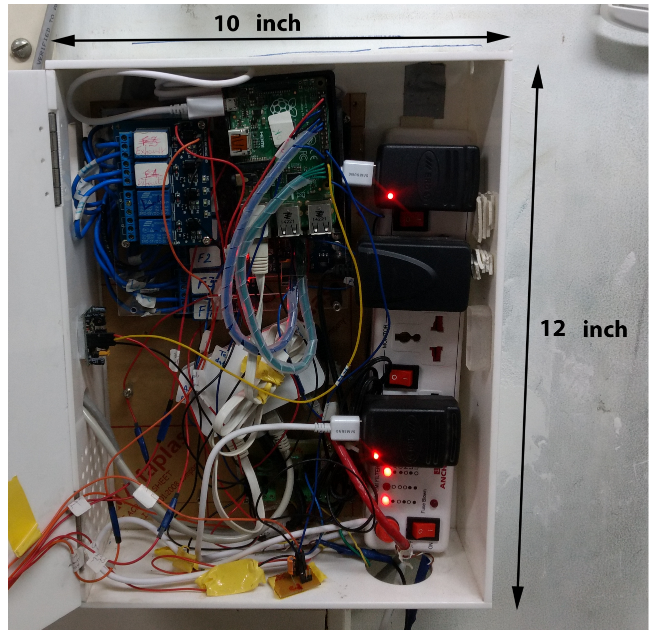


**Figure 9: SCC 2$^{nd}$ Version**

## 3.2 Appliance Portal

Appliance portal gives a web interface to control appliance in a lab environment. The user can use this portal to login and turn required devices ON and OFF. This application wasn't just built as an automation project; it was built to understand user's preferences regarding appliances that they use. The data collected is used to understand and build an algorithm which tries to cater and fulfill every occupant's needs. The idea is possible because our testbed has a cubical arrangement for occupants. Hence people sit at specific location like an office environment. Appliance control is further connected to Smart Door application which identifies who is coming inside and who is going outside the lab. Which in complete gives an automatic indoor environment control for spaces similar to our testbed. And the goal of energy efficient lab is achieved.

Can we use the sensor suit to build this application? For answering the question let us understand what are the core requirements. We need an actuator node which can turn a particular appliance on and off on the digital commands. As this is used for indoor climate control, we will need a sensor node placed at a various location which can send temperature data for decision making. We need a controller node to run the algorithm, take a decision and send commands to actuator node. All these requirements can be satisfied with the help of sensor suit.

The previous version of this application was build before wireless revolution at our labs. It was build using a high-end controller which cost around 35 $, which used to run a client-side algorithm. This controller was connected using a LAN cable which used to communicate with the web server. And temperature sensors are connected using one wire protocol. For this, a long wire needs to be routed across the room so that temperature sensors can be attached and data can be collected at the central node. This system may suffice as all the physical control is at the same location. Intervention at a single can get the system going but what if the physical control was distributed in two different places. Then it will need two intervention or stretch the control wires from one location to the other which are almost on the other side of the space. Which nearly doubles the investment made for the one and aesthetics of the space also get affected. As temperature sensors were connected using one-wire, the physical position of the temperature sensors was manually mapped to the temperature sensor's ID. This system was not scalable which means we had to find a better solution which will be cost efficient and scalable at the same time.

Sensor suit came to our rescue and solved both of our problems. 35 $ controller is replaced with a 5 $ micro-controller and no extra wires. Reduction of the hardware footprint allows us to preserve the aesthetics of the space. Currently, the web server running the appliance portal commands the actuator node over Wifi using web sockets and MQTT. Temperature sensors are battery powered and independently placed. They have their unique ID's which can be used by the node-red visual interface for configuration or showing on dashboard etc. And all the temperature data is logged into database wirelessly.

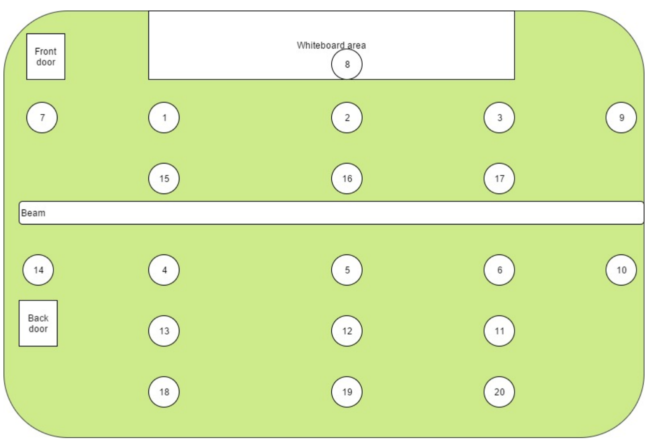


**Figure 10: Auditorium sensor placement**

### 3.3 Thermal Profiling

People spend 90% of their time inside buildings. Buildings are equipt with devices that consume energy to cater to the comfort of occupants. These devices include refrigerators, water coolers, Lights Fan, HVAC, etc. The previous study reveals that HVAC system consumes 40% to 70% of total energy consumed by the buildings. Understanding energy utilization, building management decides upon policies such that the trade-off between energy consumption and occupants comfort can be achieved. For example, people are suggested to utilize spaces which best suits the requirements, a set point temperature to be maintained at the HVAC system. All these policies are implemented so that energy utilization can be improved and energy consumption can be reduced. Our building also has the policy to regulate and maintain the indoor temperature of the auditorium between 22 to 23 degrees Celsius. But some complaints were registered by occupants regarding discomfort felt due to the decrease in temperature. A set of experiments was conducted to inspect auditorium space to address the complaints. The first step was to confirm that the discomfort faced by occupants is due to maintained temperature for a longer period or considerable decrease in temperature beyond the set point temperature. Placement of temperature collection system was made near the seats to observe the changes in temperature at occupant level in the auditorium. Graphical analysis showed that temperature was gradually decreasing from room temperature to set point temperature, but reduces further to 18 to 19 degrees Celsius before air conditioners turn off the compressors. To understand the malfunctioning air conditioning system was thoroughly analysis. Placement of thermostat in was near the inlet of the blower inside the air conditioning system which is stationed above the auditorium. The cooler air used to take a lot of time to reach to the thermostat. Till the time thermostat reacts to the change in temperature, room temperature at occupants seats goes below 22 degrees Celsius. To further understand thermal behavior of the space and develop an intervention which can regulate and maintain the temperature throughout the space, there was a need to carry out thermal profiling of the space.

For doing thermal profiling, there was a need to place sensors indoors and transmit data wirelessly. A market survey was carried to outsource such node which can be configured according to the experimental specification. As mentioned in the related work section products available in the market did not fulfill the purpose either because they were costly, nonreconfigurable, power hungry, wired. Then the decision was made to design and build a prototype of a node which is cost-effective, open-source hardware as well as software, battery powered, have a small footprint, ability to sense the value from the temperature sensor and transmit it wirelessly.

### 3.4 Solar Infrastructure

### 3.5 Home Appliance Automation

## 4 FUTURE WORK AND CHALLENGES FACED

## 5 CONCLUSION

[1].

## ACKNOWLEDGMENTS

The authors would like to thank Dr. Yuhua Li for providing the matlab code of the *BEPS* method.

The authors would also like to thank the anonymous referees for their valuable comments and helpful suggestions. The work is supported by the National Natural Science Foundation of China under Grant No.: 61273304 and Young Scientsts' Support Program (http://www.nnsf.cn/youngscientsts).

## REFERENCES

[1] Leslie Lamport. 1986. *LaTeX User's Guide and Document Reference Manual.* Addison-Wesley Publishing Company, Reading, Massachusetts.

n