International Journal of Engineering Research and General Science Volume 3, Issue 6, November-December, 2015
ISSN 2091-2730

Web Integrated Smart Home Infrastructure Using Internet of Things

Manveer Joshi, Bikrampal Kaur

Department of IT, Chandigarh Engineering College, Landran, Mohali, Punjab, India

Email: joshi.manvir@gmail.com

Abstract— Internet of Things is an evolving concept with an ever increasing range of applications leading to development of new technologies and methods for the development of IoT environment. A smart home is one such application of IoT. In this paper a framework for smart home is proposed using Constrained Application Protocol which provides a method to control sensors and actuators remotely over the Internet. An application using Constrained Application Protocol is developed to demonstrate the smart home concept by incorporating light, temperature and humidity sensors. The application is developed using Contiki and simulated with the help of Cooja simulator. Also an experimental evaluation of the suitable medium access layer protocol and radio duty cycle protocol for smart home infrastructure.

Keywords—Internet of Things, Smart Home, Constrained Application Protocol, Zigbee, Contiki, IPv6, WSN

INTRODUCTION

Internet has changed human's life by providing anytime, anywhere connectivity with anyone. Recent innovations in the information communication and technology field are strongly focused towards the Internet of Things (IoT), which will definitely lead to an enhancement in the home environments. Internet of Things (IoT) offers us a world in which one will be connected with every kind of object around us [1][2]. In IoT world one can access information about anything from anywhere at any time but there exists many challenges to interface these object with the Internet.

A smart home is one such application. It is basically a home or living environment where all the things such as home appliances to be controlled automatically or remotely [3]-[6]. In this paper, we present a home automation framework where things are controlled remotely through Internet.

The applications of an interconnected IoT ecosystem are so extensive that they challenge our preconceptions of interfacing with the Internet. The embedded and resource-constrained devices central to the IoT ecosystem provides highly specific functions for sensing and actuating. These low power and low cost devices forms a network of interconnected objects to transform our home to smart home. In this context, the home automation system/devices are getting popular both in industry and among researchers due to their ability to change our homes by increasing the safety, energy efficient and comfort of our homes.

In this paper the objective is to provide a versatile, low-cost and flexible automated home infrastructure for use in private homes which can be controlled remotely through internet. A basic idea for such type of Smart Homes using loT is shown in figure 1.

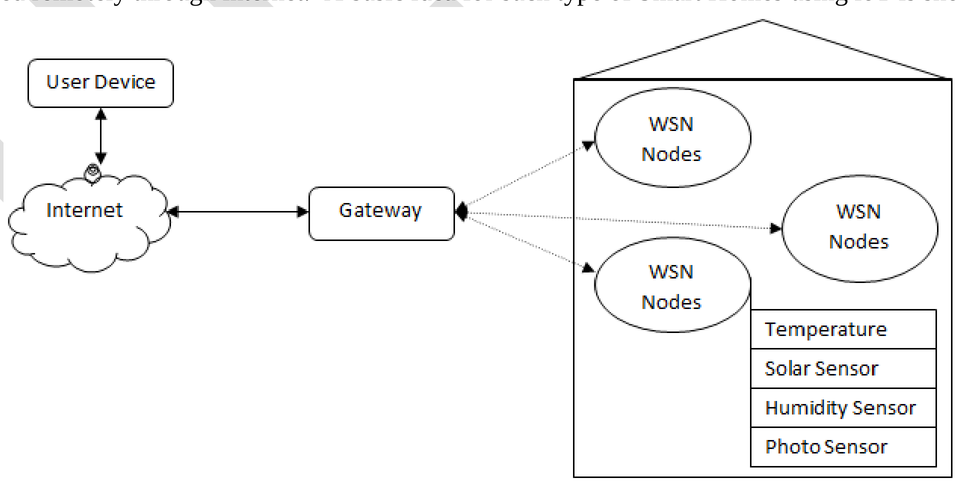


Figure 1: Smart Home Framework

As shown in the figure 1, this type of home automation system will consist of a number of smart nodes in a network, incorporating several sensors. In this paper for consideration only four sensors: light, solar, temperature and humidity sensors are shown but in reality any number of sensors can be connected depending on the technology and hardware used. The objective here is that nontechnical user should be able to use the system easily. These sensors nodes are easy to install and cable free, easy to use and maintain. These sensors nodes are connected to a gateway device by ZigBee which connects them to Internet through Ethernet or Wi-Fi. The gateway device basically acts as a bridge for sending data from wireless sensor network to Internet. A simple, secure web based system is used to read node and sensor data from the internal database and display in an easily interpreted graphical fashion. The website can be accessed through the internet by any suitable browser but is optimized for Mozilla Firefox.

A low-end implementation of a home automation network will subject to the same design considerations as bigger networks for instance those used for urban traffic control and power-grid management networks. One primary design consideration, for the selection of hardware, is to choose networking and routing protocols which must offer a scalable, versatile, and auto management solution. Second important choice is the selection of a set of parameters which are suited for the purpose and size of the smart node network. As home automation network is much smaller and simpler than its above-mentioned counterparts, it is more than likely that a set of networking and routing parameters can be optimised for such a situation. As a physical network is hard to evaluate for network and routing parameters, a software simulator/emulator is used to obtain the required data. The use of web services on the simulated network is tested and found to be feasible. Web services provide a flexible and reliable way to remotely obtain sensor and network data and are easy to program once the underlying system firmware is available. Finally, to access web based services Constrained Application Protocol is used to demonstrate its utility in network consisting of resource constrained nodes.

RELATED WORK AND METHODOLOGIES USED

Gaikwad et al. surveyed the smart home systems based on Internet-of-Things [7]. He discussed the IoT architecture along with the problems and challenges faced by IoT concept as applicable to smart homes. He also proposed the solutions for some of the problems and challenges faced.

Kamilaris and Pitsillides discussed the web of things architecture application for smart homes with the help of a case study [8]. They discussed the idea of energy-aware smart homes which have the potential to offer solution for energy awareness and integration in addition to their integration in smart electricity grids. They proposed the idea of combining sensor devices with residential smart power outlets.

Wang et al. builds a smart robot consisting of two parts: a smart phone acting as brain and a robot car as body [6]. While user is away from home, he can monitor different parameters of home measured by different sensors in robot body while robot brain acted by smart phone is with him.

Wang et al. proposed a smart home control system solution consisting of smart central controller and wireless sensor and actuator network (WSAN) to manage and control home appliances [9]. They developed different control modules in the WSAN to control varied type of home appliances. Any number of home appliances can be added or removed thereby giving the required flexibility.

Ventylees remotely monitored the smart home and provided biometrics and public key encryption based security system and SMS based alarm system while authenticated user is away from the home [5]. He used a gateway which uses ZigBee IEEE 802.15.4 based Sensor Network, GSM and Wi-Fi wireless networks to control the wireless communication in smart home systems.

Ye and Huang presented a Cloud based smart home framework for home automation, household mobility and interconnection without the need of high performance computers [4]. Bing presented smart home system architecture by using gateway, 3G and ZigBee.

Bergmann used CoAP and the proprietary FS20 protocol for the demonstration of home automation concept [10]. The FS20 device addresses was mapped to path segments of CoAP URIs whereas its commands to the basic CoAP operations.

Han and Lim designed a smart light control system based on sensor network for smart home and energy control production. ZigBee based smart home architecture is proposed by them for control of different appliances by assigning them coordinator, router and control device roles [11]. In another paper, they proposed an IEEE802.15.4 and ZigBee based Smart Home Energy Management System (SHEMS) [12]. They proposed Disjoint Multi Path based Routing protocol to improve the performance of Zigbee sensor network.

PRESENT WORK

In this paper, we present a smart home framework concept based on CoAP and Zigbee. Not many researchers have covered this though many design concepts and framework exists in the literature as discussed in previous section. Moreover, the paper offers to manage all the home appliances monitored and controlled by different sensors and actuators using any device capable of running a web browser. The framework is easily extendable to large number of wireless sensor network nodes supporting the ZigBee protocol. Also, Internet connectivity can be provided to smart home network through Ethernet or Wi-Fi.

As shown in the Figure 1 it consists of three principal parts: ZigBee network, IP network and a gateway device. The ZigBee wireless sensor network forms a IEEE 802.15.4 6lowpan network where all the sensor and actuator nodes exists. To provide Internet connectivity to these nodes or 6lowpan network, we need some mechanism to connect them to Internet. This is done by a device known as border router or gateway, the purpose of which is to interface IP network to the 6lowpan network. The data packets are routed through this device to both the networks.

The software framework selected for development is Contiki OS which is an open source operating system designed for memory constrained networked systems with special emphasis on low power IoT devices. It only needs 10KB RAM and 30KB ROM for proper operation, therefore a lot of hardware options exists for its implementation. In this paper, we have selected Tmote sky mote as the hardware for sensors and actuator connectivity to support ZigBee based 6lowpan network.

The smart home framework concept is proved by simulation using Contiki OS provided Cooja simulator with Tmote sky nodes. The Cooja also offers the opportunity to emulate the Tmote sky nodes. The Contiki software architecture is shown in Figure 2 as shown by Shelby & Bormann [13].

<table>
<tr><td colspan="4">User Apps</td><td>Built-in-Apps</td></tr>
<tr><td colspan="4">Socket API</td><td rowspan="5">Contiki OS</td></tr>
<tr><td colspan="2">UDP</td><td colspan="2">TCP</td></tr>
<tr><td>IPv6</td><td colspan="2">RPL</td><td>ICMPv6</td></tr>
<tr><td colspan="4">6LoWPAN Adaptation</td></tr>
<tr><td colspan="4">Rime (MAC)</td></tr>
<tr><td colspan="4">Platform</td><td>CPU</td></tr>
<tr><td colspan="5">Hardware Drivers</td></tr>
</table>

Figure 2: Contiki Software Architecture

As shown in the Figure 2, at the top layer of software architecture is user apps. In the proposed framework at the application layer CoAP (Constrained Application Protocol) is used for interaction with 6lowpan network. CoAP is a light-weight application layer protocol based on UDP transport intended to be used for resource constrained devices which need to be controlled remotely over the internet. CoAP offers multicast, low overhead and simplicity which are advantageous for IoT networks.

To further reduce the memory footprint and energy consumption we have evaluated Contiki OS medium access layers protocols available. Media access control (MAC) layer algorithms play a vital and crucial role in optimizing communications to increase battery life. The prime goal of a MAC algorithm is to reduce contention in the wireless medium between simultaneously communicating devices, which must both back off , retry later time if any another node is transmitting already.

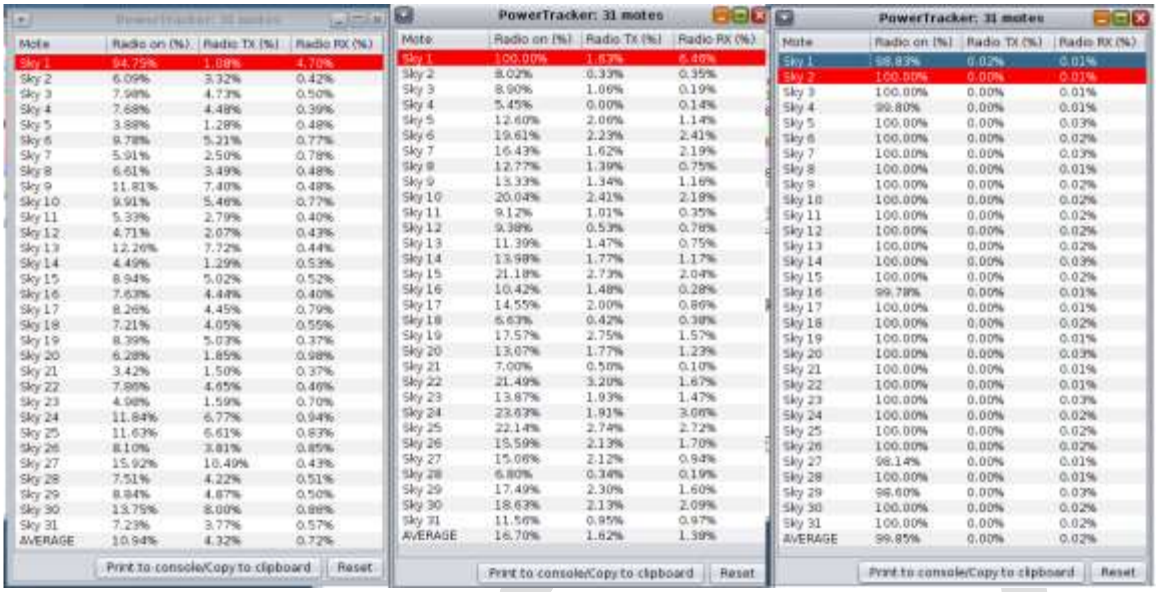
Contiki MAC consists of three layers: MAC, Radio Duty Cycle (RDC) and framer. The MAC used is CSMA and nullmac. An experimental evaluation and optimization is done to select the best MAC. Radio duty cycling (RDC) algorithm a part of MAC algorithms is also optimized. These algorithms determine how often a radio of a device ‘wakes up’ to check whether there are any other devices trying to communicate with it. RDC layers implemented in Contiki are ContikiMAC, xmac, Low Power probing (lpp), nullrdc-noframer, nullrdc and sicslowmac.

RESULTS AND DISCUSSIONS

Since sensor nodes are constrained for computing power, memory and energy, it is very important that proposed framework implementation should have very low memory footprint as well as low energy consumption so that nodes can have longer operational life with no maintenance since they can be used at places that could be hard to reach. Therefore, several optimizations in Contiki IPv6 stack and application code size is done to fit the firmware in the selected sensor node memory. Furthermore, optimization at the MAC and RDC layer of Contiki is done to increase the power efficiency of the network and reduce the firmware size.

As stated in previous section an experimental evaluation is done at Contiki MAC layer to select best option for home automation network and it is found that ContikiMAC and CSMA is the best choice for the network. ContikiMAC with CSMA average radio duty cycle is found minimum as shown in Figure 3 at 10.94 % as compared to other options with CSMA MAC for 30 client and one server node. The first, second and third part of the Figure 3 corresponds to ContikiMAC, xmac and nullmac average radio duty cycle tables while CSMA is selected as MAC.

Also the PRR statistics were also analyzed for all 30 sensor nodes and the average PRR of ContikiMAC is found **50.48%** and it is concluded that as compared to other MAC ContikiMAC implementation is more energy and throughput efficient. Therefore Contiki MAC is selected as RDC layer for home automation system.



PowerTracker: 31 motes

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	94.75%	1.08%	4.70%
Sky 2	6.09%	3.32%	0.42%
Sky 3	7.98%	4.73%	0.50%
Sky 4	7.68%	4.48%	0.39%
Sky 5	3.88%	1.28%	0.48%
Sky 6	9.78%	5.21%	0.77%
Sky 7	5.91%	2.50%	0.78%
Sky 8	6.61%	3.49%	0.48%
Sky 9	11.81%	7.40%	0.48%
Sky 10	9.91%	5.46%	0.77%
Sky 11	5.33%	2.79%	0.40%
Sky 12	4.71%	2.07%	0.43%
Sky 13	12.26%	7.72%	0.44%
Sky 14	4.49%	1.29%	0.53%
Sky 15	8.94%	5.02%	0.52%
Sky 16	7.63%	4.44%	0.40%
Sky 17	8.26%	4.45%	0.79%
Sky 18	7.21%	4.05%	0.55%
Sky 19	8.39%	5.03%	0.37%
Sky 20	6.28%	1.85%	0.98%
Sky 21	3.42%	1.50%	0.37%
Sky 22	7.86%	4.65%	0.46%
Sky 23	4.98%	1.59%	0.70%
Sky 24	11.84%	6.77%	0.94%
Sky 25	11.63%	6.61%	0.83%
Sky 26	8.10%	3.81%	0.85%
Sky 27	15.92%	10.49%	0.43%
Sky 28	7.51%	4.22%	0.51%
Sky 29	8.84%	4.87%	0.50%
Sky 30	13.75%	8.00%	0.88%
Sky 31	7.23%	3.77%	0.57%
AVERAGE	10.94%	4.32%	0.72%

Print to console/Copy to clipboard | Reset

PowerTracker: 31 motes

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	100.00%	1.83%	6.46%
Sky 2	8.02%	0.33%	0.35%
Sky 3	8.90%	1.06%	0.19%
Sky 4	5.45%	0.00%	0.14%
Sky 5	12.60%	2.06%	1.14%
Sky 6	19.61%	2.23%	2.41%
Sky 7	16.43%	1.62%	2.19%
Sky 8	12.77%	1.39%	0.75%
Sky 9	13.33%	1.34%	1.16%
Sky 10	20.04%	2.41%	2.18%
Sky 11	9.12%	1.01%	0.35%
Sky 12	9.38%	0.53%	0.78%
Sky 13	11.39%	1.47%	0.75%
Sky 14	13.98%	1.77%	1.17%
Sky 15	21.18%	2.73%	2.04%
Sky 16	10.42%	1.48%	0.28%
Sky 17	14.55%	2.00%	0.86%
Sky 18	6.63%	0.42%	0.38%
Sky 19	17.57%	2.75%	1.57%
Sky 20	13.07%	1.77%	1.23%
Sky 21	7.00%	0.50%	0.10%
Sky 22	21.49%	3.20%	1.67%
Sky 23	13.87%	1.93%	1.47%
Sky 24	23.63%	1.93%	3.06%
Sky 25	22.14%	2.74%	2.72%
Sky 26	15.59%	2.13%	1.70%
Sky 27	15.06%	2.12%	0.94%
Sky 28	6.80%	0.34%	0.19%
Sky 29	17.49%	2.30%	1.60%
Sky 30	18.63%	2.13%	2.09%
Sky 31	11.56%	0.95%	0.97%
AVERAGE	16.70%	1.62%	1.38%

Print to console/Copy to clipboard | Reset

PowerTracker: 31 motes

Mote	Radio on (%)	Radio TX (%)	Radio RX (%)
Sky 1	98.83%	0.02%	0.01%
Sky 2	100.00%	0.00%	0.01%
Sky 3	100.00%	0.00%	0.01%
Sky 4	99.80%	0.00%	0.01%
Sky 5	100.00%	0.00%	0.03%
Sky 6	100.00%	0.00%	0.02%
Sky 7	100.00%	0.00%	0.03%
Sky 8	100.00%	0.00%	0.01%
Sky 9	100.00%	0.00%	0.02%
Sky 10	100.00%	0.00%	0.02%
Sky 11	100.00%	0.00%	0.02%
Sky 12	100.00%	0.00%	0.02%
Sky 13	100.00%	0.00%	0.02%
Sky 14	100.00%	0.00%	0.03%
Sky 15	100.00%	0.00%	0.02%
Sky 16	99.78%	0.00%	0.01%
Sky 17	100.00%	0.00%	0.01%
Sky 18	100.00%	0.00%	0.02%
Sky 19	100.00%	0.00%	0.01%
Sky 20	100.00%	0.00%	0.03%
Sky 21	100.00%	0.00%	0.01%
Sky 22	100.00%	0.00%	0.01%
Sky 23	100.00%	0.00%	0.03%
Sky 24	100.00%	0.00%	0.02%
Sky 25	100.00%	0.00%	0.02%
Sky 26	100.00%	0.00%	0.02%
Sky 27	98.14%	0.00%	0.01%
Sky 28	100.00%	0.00%	0.01%
Sky 29	96.60%	0.00%	0.03%
Sky 30	100.00%	0.00%	0.02%
Sky 31	100.00%	0.00%	0.02%
AVERAGE	99.85%	0.00%	0.02%

Print to console/Copy to clipboard | Reset

Figure 3: RDC layers comparison

A comparison is also done with respect to channel listen frequencies of 16 Hz, 8 Hz and 4 Hz. It is found that the average listening power consumption increases with the RDC frequency whereas the average transmission power decreases with a net result of increase in total average power consumption frequency. The lowest RDC frequency of 4 Hz is found most appropriate in our case for battery-powered nodes. Figure 4 shows the Instantaneous Power Consumption at RDC frequency of 4 Hz.

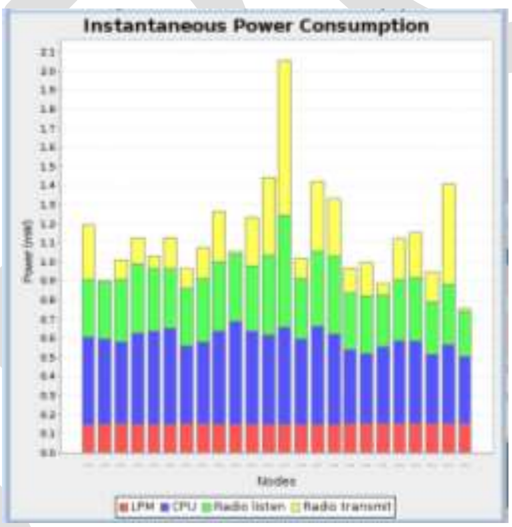


Figure 4: Instantaneous Power Consumption at RDC frequency of 4 Hz

After selection of the suitable protocols CoAP is used to control different sensors mounted on sensor nodes interfaces with Copper add on running on the Mozilla web browser over the Internet. The sensor node resources are mapped to CoAP uniform resource identifiers (URI) which can be accessed by the CoAP methods defined as GET, POST, PUT and DELETE. CoAP makes use of GET, PUT, POST, and DELETE methods in a same manner to HTTP and are used to manipulate the resources. A CoAP URI used in this example is coap://[aaaa::212:740b:b:b0b]5683/sen/humidity for humidity sensor. The user enters URI in browser and requests for a resource. The browser displays the result. Figure 5 and 6 shows the browser snapshots of the application while copper add on is working as client. In figure5 humidity sensor returns the value of 150.62 while accessing this resource with CoAP GET method. Similarly in the Figure 6 temp sensor is accessed via CoAP URI coap://[aaaa::212:740b:b:b0b]5683/sen/temp to get a value of 24.00^{o}C.

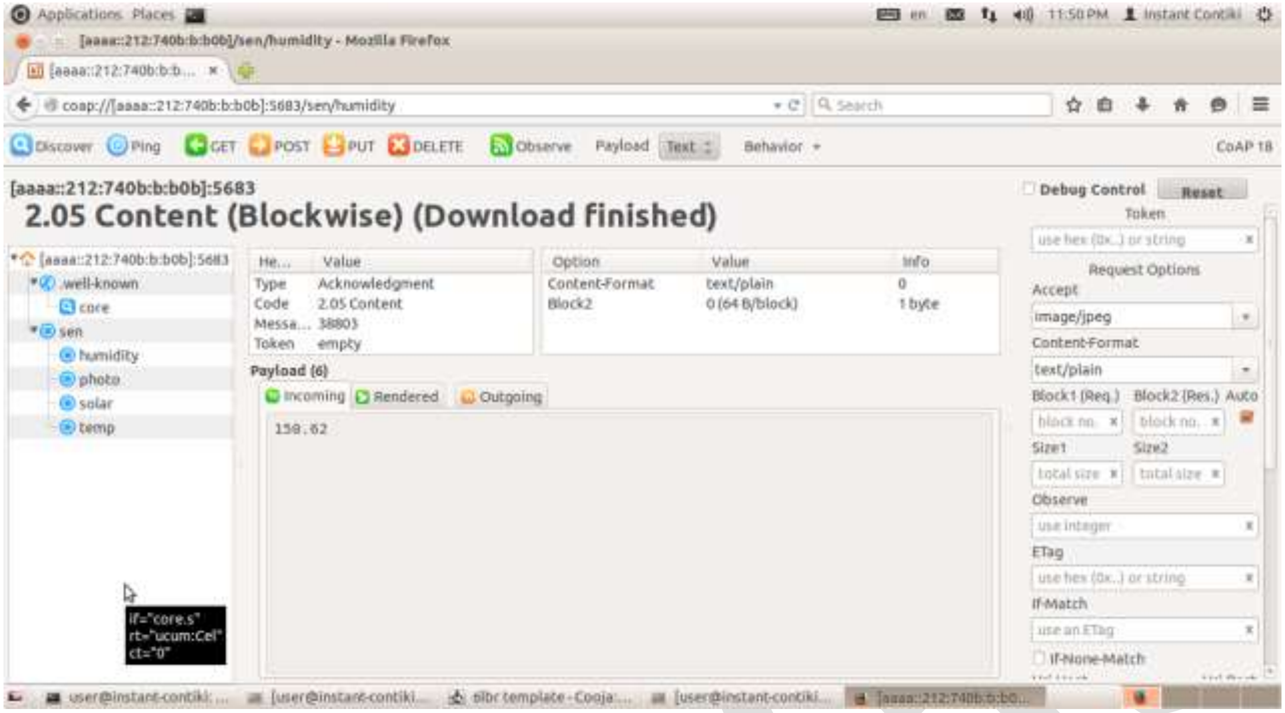


Figure 5: A successful response from a CoAP server to a request for its '/sen/humidity' resource, which returned the value "150.62"

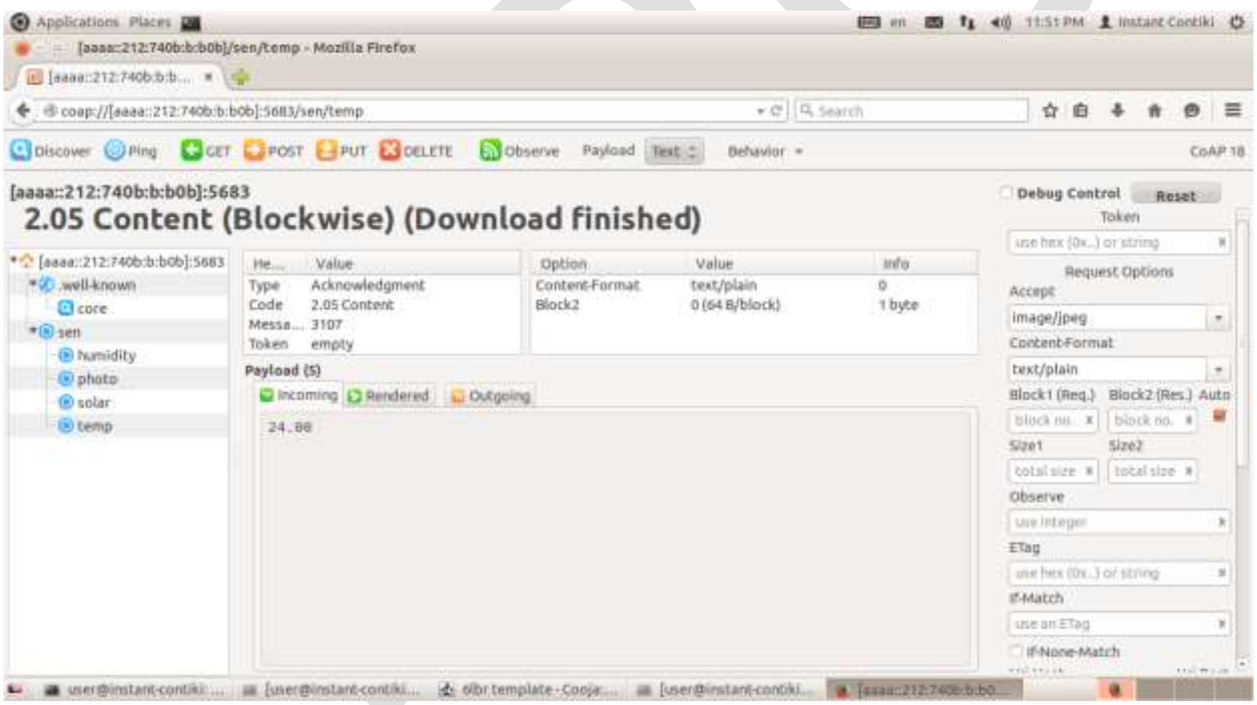


Figure 6: A successful response from a CoAP server to a request for its '/sen/temp' resource, which returned the value "24.00"

CONCLUSION

A smart home infrastructure is proposed which can be monitored and controlled over the Internet using CoAP. Since the wireless sensor nodes are constrained devices in terms of processing power and memory, it is essential to use suitable technologies and to achieve this Contiki OS is selected. Furthermore, an experimental evaluation is done to find the most suitable MAC and RDC layer protocols of Contiki OS MAC layer. In addition to it, variable, stack and application size optimization is done to fit the firmware in the sensor nodes memory. Further studies can be carried out to propose new low memory footprint protocols which will enhance the life of sensor nodes.

REFERENCES:

[1] Dhananjay Singh, Gaurav Tripathi and Antonio J. Jara "A survey of Internet-of-Things: Future Vision, Architecture, Challenges and Services" IEEE Proc. of the IEEE World Forum on Internet of Things (WF-IoT), pp. 287-292, 2014

[2] Moataz Soliman, Tobi Abiodun, Tarek Hamouda, Jiehan Zhou1 and Chung-Horng Lung "Smart Home: Integrating Internet of Things with Web Services and Cloud Computing" IEEE International Conference on Cloud Computing Technology and Science, vol. 2, pp. 317 – 320, 2013

[3] Kang Bing, Liu Fu, Yun Zhuo, and Liang Yanlei "Design of an Internet of Things-based Smart Home System" 2nd International Conference on Intelligent Control and Information Processing (ICICIP), vol. 2, pp. 921 – 924, 2011

[4] Xiaojing Ye and Junwei Huang "A Framework for Cloud-based Smart Home" International Conference on Computer Science and Network Technology (ICCSNT), vol. 2, pp. 894 – 897, 2011

[5] Ventylees Raj.S "Implementation of pervasive computing based high-secure smart home system" IEEE International Conference on Computational Intelligence & Computing Research (ICCIC), pp. 1-8, 2012

[6] Haidong Wang, Jamal Saboune and Abdulmotaleb El Saddik "Control your smart home with an autonomously mobile smartphone" IEEE International Conference on Multimedia and Expo Workshops (ICMEW), pp. 1-6, 2013

[7] Pranay P. Gaikwad, Jyotsna P. Gabhane and Snehal S. Golait "A Survey based on Smart Homes System Using Internet-of-Things" IEEE International Conference on Computation of Power, Energy Information and Communication (ICCPEIC), pp. 0330 – 0335, 2015

[8] Andreas Kamilaris and Andreas Pitsillides "Towards Interoperable and Sustainable Smart Homes" 1ST IEEE Africa Conference and Exhibition (1ST-Africa), pp.1 -11, 2013

[9] Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang, Chengdong Li "An IoT-based Appliance Control System for Smart Homes" Fourth International Conference on Intelligent Control and Information Processing (ICICIP), pp. 744-747, 2013

[10] Olaf Bergmann, Kai T. Hillmann, Stefanie Gerdes "A CoAP-Gateway for Smart Homes" International Conference on Computing, Networking and Communications, Communication Software and Services Symposium, pp.446 -450, 2012

[11] Dae-Man Han and Jae-Hyun Lim "Smart Home Energy Management System using IEEE 802.15.4 and ZigBee" IEEE Transactions on Consumer Electronics, vol. 56, no. 3, pp. 1403-1410, August 2010

[12] Dae-Man Han and Jae-Hyun Lim "Design and Implementation of Smart Home Energy Management Systems based on ZigBee" IEEE Trans. Consum. Electron, vol. 56, no. 3, pp.1417 -1425, 2010

[13] Z Shelby and C Bormann, "6LoWPAN: The Wireless Embedded Internet", John Wiley & Sons, November 2009