A Reconfigurable IoT Architecture with Energy Efficient Event-Based Data Traffic Reduction Scheme

https://doi.org/10.3991/ijoe.v13i02.6460

Varun Tiwari
Visvesvaraya National Institute of Technology, Nagpur, India
varun.etrx@gmail.com

Avinash Keskar
Visvesvaraya National Institute of Technology, Nagpur, India
agkeskar@ece.vnit.ac.in

NC Shivaprakash
Indian Institute of Science, Bangalore, India
shiv@iisc.ernet.in

Abstract—Designing an Internet of Things (IoT) enabled environment requires integration of various things/devices. Integrating these devices require a generalized approach as these devices can have different communication protocols. In this paper, we have proposed generalized nodes for connecting various devices. These nodes are capable of creating a scalable local wireless network that connects to the cloud through a network gateway. The nodes also support over the air programming to re-configure the network from the cloud. As number of devices connected to the cloud increases, the network traffic also increases. In order to reduce the network traffic we have used different data transfer schemes for the network. We have also proposed an event-based data transfer scheme for situations where there is low probability of change in sensor value. The experimental results shows that the event-based scheme reduces the data traffic by up to 48% under practical conditions without any loss of information compared to priority based data transfer. We have also shown that the proposed scheme is more reliable for data transfer in a large network with a success rate of 99.5% measured over 200 minutes for 1201 data packets.

Keywords—Data Transfer Scheme, Internet of Things, Network Architecture, Wireless Network

1 Introduction

With the development of internet technology since 1960's nothing has remained the same. More and more information are now being transferred with ease. Initially the internet was a concept to connect various computers in a network so that information can be shared between them. But with the advancement in technology it has now become possible to connect not only the computers but also many other things with the internet and the concept in known as internet of things. With this technology any device can be connected to any other device via internet to share information. These devices can be mobile phones, laptops, Vehicles, Washing machines, Television, bicycles, wearable devices; in-fact the device can be anything that we want. The advancement in technology has provided a scope to convert anything to an internet connected device. Presently around 15 billion devices are connected to internet and it is expected to be more than 50 billion by 2020 [1].

Integration of things opens a wide area of application for this technology [2] such as Health care systems [3, 4], transportation systems [5, 6], farming systems [7, 8], energy management [9], wearable technology [10] and smart environments [11, 12]. Most of the work done in the field of IoT is related to resource management and device interconnections that include design of new IoT enabled devices for integration. Sumi Helal [13] in 2005 urged that the cloud and sensor based architecture should be designed to support programmability along with integration. The programmability of the architecture can enable the programmers to use the sensors and actuators as a service. It can also enable the architecture to be scalable and reconfigurable. In [14] a cloud edge beneath architecture is designed to support programmable scalability of cloud sensors. Reference [15] presents a top-level generic cloud based architecture for smart cities with different IoT platforms. Reference [16] discusses a service management framework for IoT devices with data generated real time as well non-real time in a cloud architecture. Most of the IoT architectures are designed by keeping cloud servers as a reference and the actual generalized approach towards hardware nodes design remains untouched. The heterogeneity of the hardware is a big hurdle in creating a programmable integrated environment. In this work, a generalized approach towards programmable hardware nodes in cloud architecture is proposed. It provides the flexibility to modify the wireless nodes architecture or to add more devices and nodes wirelessly from the cloud itself.

As more and more devices are connected, the amount of data traffic will also increase. This huge data traffic can cause flooding in the network and can delay the data reception or loss of data [17]. In order to deal with this huge amount of data traffic we need to have a very fast network connection or we can follow certain methods in the design or communication process to reduce the data itself without sacrificing the information. Yashiro et al. [18] in their work used constrained application protocol (CoAP) [19] which is a lightweight protocol for resource oriented applications. Using lightweight protocols can be one way to reduce the data transfer load to the cloud. However, having some additional efficient data reduction scheme for the IoT nodes without loss of information can be very beneficial. In this work, a generalized reconfigurable architecture is proposed to connect various devices with an efficient data reduction and transmission scheme to reduce the data traffic.

The rest of the paper is organized as follows. Section 2 describes the architecture of the network to connect various devices. Section 3 describes the generalized network nodes required to design the architecture. Section 4 discusses the different data transfer schemes for the network. Section 5 shows the simulation results of the different data transfer schemes with a probabilistic model followed by section 6 with reliability test of the network.

2 System Architecture

In order to convert an environment into a complete internet connected one there can be several things that need to be connected wirelessly. One way of achieving this is to enable all the things with Wi-Fi capability and create a network. Now a days, a lot of devices comes with Wi-Fi capabilities e.g. TVs, AC, etc. But this approach is good for devices that already have this capability or can be easily integrated with this capability, but it cannot be considered as a general approach because in a local environment there can be various devices which cannot be integrated easily e.g. integrating an old AC or washing machine with Wi-Fi is not easy. So another approach is required where these kinds of devices are connected through a relay/switch which is connected in a local network which later connects to internet through a network gateway. Along with this connectivity special nodes can be created to have awareness about the local environment, these nodes can have different sensors to measure the local parameters so that a better understanding for interconnectivity of the devices can be achieved.

Generally in order to interconnect various devices special cloud servers are used which works as a medium of interaction between the devices. The devices can have different methods of connecting and communicating, so the cloud provides an interface to connect them independently. Initially a small local network can be formed that later connects to internet or cloud server. A general architecture for converting such environment through local network is shown in the Fig.1. Multiple devices can be connected to a master node through repeaters. The master will be connected to the cloud through network gateway. Some devices sharing or controlling critical information can be directly connected through the gateway. Once the devices are connected through the network gateway they can be controlled or monitored wirelessly through mobile devices.

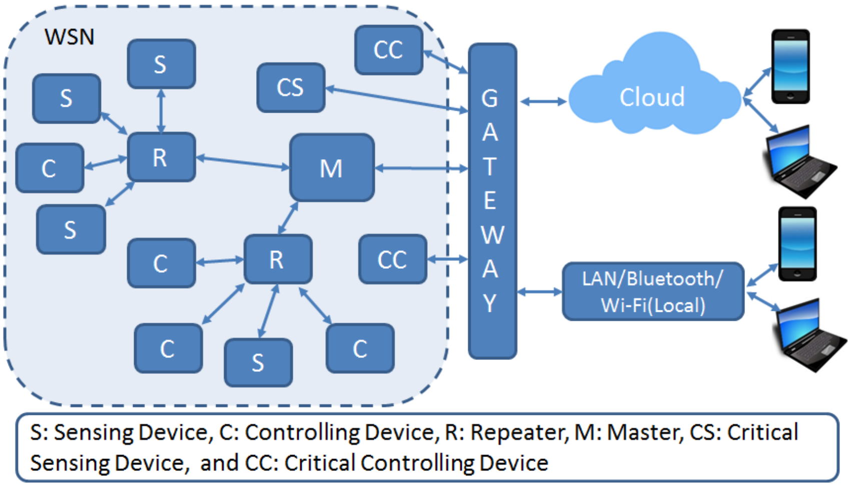

WSN
S
S
C
S
R
CS
CC
M
C
R
CC
C
S
C
GATEWAY
Cloud
LAN/Bluetooth/ Wi-Fi(Local)
S: Sensing Device, C: Controlling Device, R: Repeater, M: Master, CS: Critical Sensing Device, and CC: Critical Controlling Device

Fig. 1. Architecture of IoT enabled environment

The master can have multiple wireless connectivity options such as Wi-Fi, Bluetooth or ZigBee so that, it can be connected to things which already have Wi-Fi capability or with other nodes or repeaters that also have wireless network connectivity. The network can be integrated using star and tree topology. It enables the architecture for connecting as much devices as required. All the nodes, repeaters and master can have the capability to be re-programmed over the air so that any new device can be integrated to the system whenever required. This capability of over the air programming increases the flexibility of the system and opens the scope for deploying new network algorithms through cloud. Figure 2 shows the steps involved while uploading a new network algorithm. Here all the nodes, repeaters and master needs to be updated with a new program, so initially the master receives all the program files. Then it sends the specific node program files to the repeaters through which the nodes are connected. The repeater programs the nodes with the new program files that they received. Now the master programs the repeaters with the new program file. After this the master needs to update its own program so it sends the program file to one of the repeater and then this repeater programs the master with the new program file. Thus the complete network can be updated wirelessly from the cloud. The hardware components required for the design of architecture are discussed in next section.

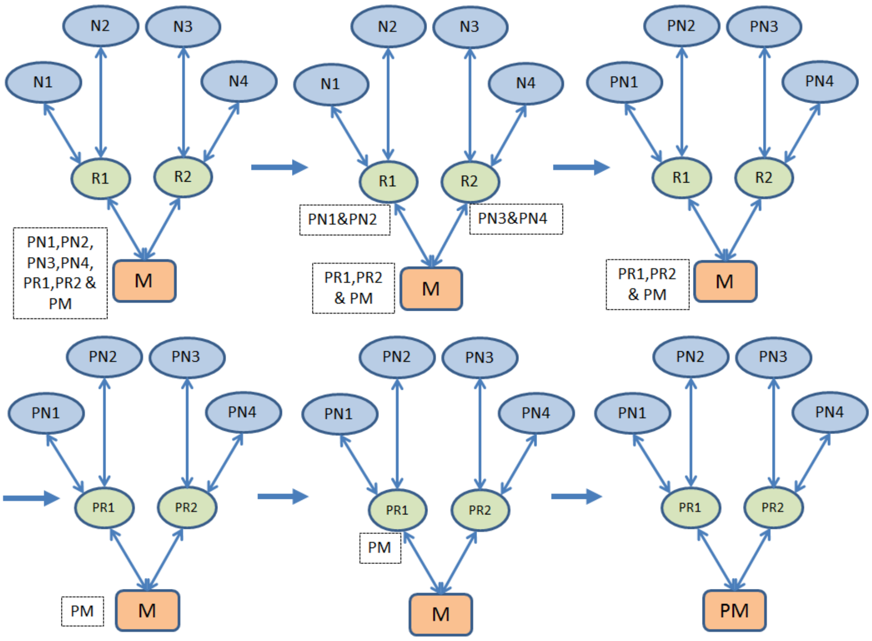


Fig. 2. Network re-configuration for IoT enabled environment

3 Network Nodes

In a local environment, there can be many devices that can be connected to a mobile device, but it is not necessary that all these devices connect using a same wireless protocol. Some devices may have a direct Wi-Fi connectivity; some may have Bluetooth connectivity or any other protocol. Along with these devices, we may be interested to add some more devices to measure local environmental parameters. This requires many sensors and relays to monitor and control the other things/devices. Designing separate dedicated nodes for every place and purpose will be a very time taking and complex process; instead, we can design only two generalized network modules. These modules are designed to have multiple wireless connectivity as well as multiple digital and analog I/O capability. The complete network can be designed by using multiple such nodes and configuring them as required. The two network nodes are described as follow:

3.1 Node 1

This node can be used to integrate various sensors and control modules to the network. The node can have the capability to integrate digital as well as analog sensors and control devices. Figure 3 shows the block diagram of such node. The node has more than one wireless connecting protocol capability, so that more number of devices with different connectivity can be connected to the node. Table 1 shows some of the possible sensors and controls that can be easily integrated with this network node.

There can be many possibilities for input-output handling capability of this type of node. However, in order to have a general architecture the nodes are designed to have the configurations as shown in Fig.4. As different sensors/controls require different interfacing protocols, the node has dedicated pins for the sensors/controls. Multiple analog and digital sensors can be interfaced with the node. An analog sensor will generate a 10-bit data using 10-bit analog to digital converters. A digital sensor can generate 1-bit or multi-bit data depending upon the sensor. Similarly, a digital control can have 1-bit or multi-bit data and an analog control can have analog data through digital to analog converters. This generalized configuration helps in designing a common protocol to connect all the local devices.

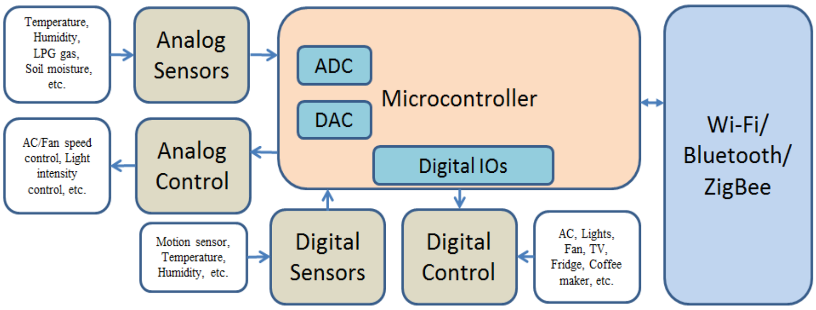

Temperature, Humidity, LPG gas, Soil moisture, etc.
Analog Sensors
AC/Fan speed control, Light intensity control, etc.
Analog Control
ADC
DAC
Microcontroller
Digital IOs
Wi-Fi/ Bluetooth/ ZigBee
Motion sensor, Temperature, Humidity, etc.
Digital Sensors
Digital Control
AC, Lights, Fan, TV, Fridge, Coffee maker, etc.

Fig. 3. Block diagram of Node 1

Table 1. Types of Digital/Analog sensors and controls for interfacing with node 1

	Sensors/Controls	Interfacing
Analog Sensors	Temperature, Humidity, Soil-moisture, LPG detector, Air quality, LDR, Flame sensor, Current sensors, Distance sensors (Ultrasonic/proximity), etc.	Through Analog to digital converters, conversion formulas are used to find the actual value
Analog Control Device	Fan speed, Light Intensity, Motor speed, etc.	Through digital to Analog converters, requires multi-bit data to decide the level of output
Digital Sensors	Temperature, Humidity, Motion, Magnetic reed relays, Digital distance sensors, Flow sensors, etc.	Single bit interfacing, or communication protocol based (UART,SPI, I2C or Other)
Digital Control Devices	Relays, LEDs, Alarms, etc.	Single bit interfacing

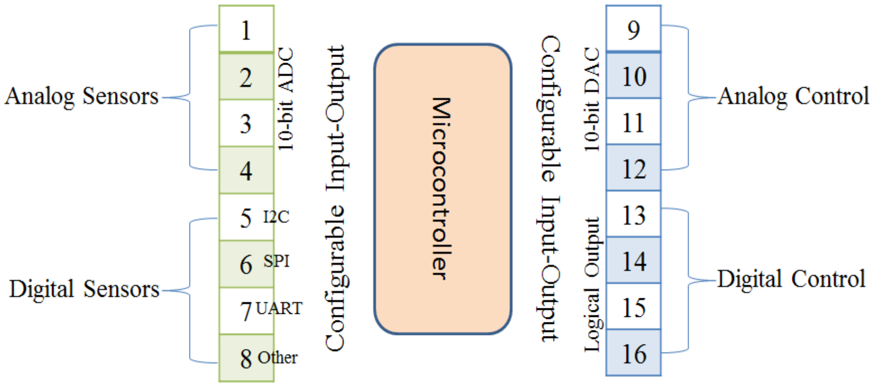


Fig. 4. Node configuration for sensors and controls interfacing

3.2 Node 2

This node is designed to work as a master node to collect all the information from the network devices and then later connect to cloud through the network gateway. The node can also be used as a repeater in the local network to increase the network area. The node can have multiple wireless connecting protocols capability to increase its usability with different devices. Figure 5 shows the block diagram of such node. This node does not need any input-output operation so it will have only wireless modules and a controller to perform the communication operations.

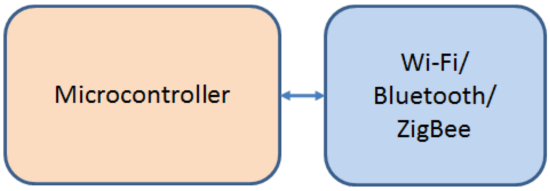


Fig. 5. Block diagram of Node 2

These network nodes can be used in a star or tree topology to form a large wireless network. These nodes are designed to have over the air programmable capability [20] so that the network can be reconfigured easily whenever required. Node 1 can be configured as general or a critical sensor/control nodes and Node 2 can be used as repeaters or master. A multi-master network can also be configured for a large network using these nodes. A large network connected using star and tree topology with multiple repeaters can have a large delay in data transmission from end node to the master, so having a multi-master network can reduce this delay and enable parallel data transfer capability from master to cloud. In the next section, different data transfer schemes are discussed for this wirelessly connected network.

4 Proposed Data Transfer Schemes

In a configured network, there will be data that needs to be communicated to the cloud. Every node can have many digital and analog data that it has to transfer to the master. For a large network, if all these sensor and things continuously tries to send the data to the master then the network can get flooded over time. This flooding can cause delay in data transmission and loss of data packets. So in order to avoid this we can have the following data transfer schemes:

4.1 Periodic Data Transfer

Continuous transfer of data is not possible as more and more devices are added to the network. In addition, it is not necessary to have the information of each node and sensors at every time. So every node is configured to send all its data periodically. The node combines all its data from sensors and forms a single data packet. It sends this data packet periodically to the master. The node containing critical data or monitoring operation is given more frequency to transmit the data. For device control operation, the master can send data dynamically to any node required. This scheme can be useful for a small network where we have less number of nodes with limited devices.

4.2 Priority Periodic Data Transfer

For a larger network where we have more number of nodes and every node is fully occupied with sensors and control task. The sensors and devices can be prioritized based on their information requirement. In this scheme, every node can send its high priority data more frequently than the other data. Here instead of sending all the node data as a single packet, small priority based data packets are formed and transmitted to the master. Table 2 shows the scheme with different sensors and control devices with different priority and data transfer period. The critical sensors like fire detection, intruder detection are given highest priority and can send data every 10 seconds whereas sensors and devices with less critical information send data with less frequency thereby reducing the data traffic. Here also for device control the master can send the data dynamically to the nodes as required.

Table 2. Sensor/ Device Priority Chart

Priority	Sensors	Control Device	Traffic Frequency
Highest	Fire, Intruder	Safety Alarm	Every 10 seconds
High	LPG	Leakage Alarm	Every 30 Seconds
Medium	Temperature, Humidity	AC, Coolers	Every 1 minute
Low	Soil Moisture, Water Tank level	Water pump	Every 2 minutes

4.3 Event Based Data Transfer

In an actual environment most of the time the data that is being monitored does not change very frequently, so an event of change based scheme can be employed to further reduce the network traffic. In this proposed scheme every node takes the reading and checks it with the previous reading. If the value has not changed, then in the next cycle of data transfer the node sends only a flag instead of the value. The flag value is set as ‘1’ to represent no change from previous value. The change in value is send by sending ‘0’ followed by the value. The scheme can be applied to all the nodes with multi-bit analog or digital values to send. For every no change of data 9 bits can be saved (considering 10-bit ADC for analog data). The scheme can be very useful for a large network of interconnected things. In order to compare the data transfers schemes probabilistic simulation is discussed in the next section.

5 Simulation Results and Discussion

For simulation of the proposed network and data transfer schemes, every node is considered to have 16 devices connected to it, with four digital inputs, four analog inputs, four digital controls and four analog controls as shown in Table 3. Every nodes needs to transfer the input values that it receives from the sensors. Considering a 10-bit analog to digital converters and 8-bit digital sensor value, every node can have a maximum of 72-bits to transfer to the master. Again considering 10-bit data for analog control and 1-bit data for digital control every node can have 44-bits of data that it can receive from the master for control task. The data that a node receives from the master for control operation is dynamic as it can be user triggered or condition based, so for simulation purpose it is assumed that the master sends the control data every 2-mins to each node. Table 4 shows the network simulation parameters for the schemes. The distance between the node and master is not considered as a parameter and is assumed equal for all the nodes. All the simulations were performed using MATLAB 2013.

5.1 Periodic Data Transfer

In this scheme, every node will send all its data periodically one after other after every 10 seconds. Every node transfers the 72-bit of data and then goes to sleep mode to save energy. The master also receives the data from the nodes and then goes to sleep to save energy. The master sends the control data to every node after 120 seconds. Therefore, the network performs the data transfer in a periodic manner. The node consumes energy Es when it is in sleep mode, energy Ec when it does transition from sleep mode to wake mode or from wake to sleep, energy Et for transmitting a bit and energy Er for receiving a bit. So periodically, every 10 seconds node consumes energy given by (1) for transmitting and every 120 seconds consumes energy given by (2) for receiving. The master consumes energy given by (3) every 120 seconds for transmitting and energy given by (4) for receiving data from nodes. Figure 6 shows the energy consumption of the master and a node for a simulation time of 200 minutes. The node has to transmit 72-bit of data every time so it consumes a lot of energy and it decreases rapidly with time. It is obvious that a continuous data transfer would require further more energy.

Table 3. Node configuration for simulation of data transfer schemes

Sensor/Control	Priority	Traffic rate (Priority Based)	No. of bits for transmission	Event Based	
				Change (Bits)	*No Change (Bits)*
AS-1	Highest	1secs	10	11	1
AS-2	High	5secs	10	11	1
AS-3	Medium	1 minute	10	11	1
AS-4	Low	2 minute	10	11	1
DS-1	Highest	1secs	8	9	1
DS-2	High	5secs	8	9	1
DS-3	Medium	1 minute	8	9	1
DS4	Low	2 minute	8	9	1
AC-1, 2,3 & 4	Dynamic	2 minute	10 each	--	--
DC-1, 2, 3 & 4	Dynamic	2 minute	1 each	--	--

Table 4. Simulation parameters

Parameter	Value
Simulation time	12, 000 seconds
No. of Nodes	4
No. of Master	1
Energy consumed by node for 1-bit transmission (Et)	1e-3 unit
Energy consumed by node for reception of 1-bit (Er)	1e-4 unit
Energy consumed at sleep (Es)	1e-5 unit
Energy consumed in transition (Ec)	1e-3 unit
Maximum available energy	100 units each

$$Node's\ Energy\ Consumed = 72 * Et + 2 * Ec \tag{1}$$

$$Node's\ Energy\ Consumed = 44 * Er + 2 * Ec \tag{2}$$

$$Master's\ Energy\ Consumed = N * 44 * Et + 2 * Ec \tag{3}$$

$$Master's\ Energy\ Consumed = N * 72 * Er + 2 * Ec \tag{4}$$

5.2 Priority Periodic Data Transfer

In this scheme, the data is transmitted from every node depending on the priority of the data. As in a practical environment all the nodes information at all the time is not necessary. The information can have different priorities. So in order to simulate this practical scenario it is considered the nodes have equal variety of highest, high, medium and low priority data from analog and digital sensors. Table 3 shows the node configuration with different priorities for this scheme. The traffic frequency can be configured as per the requirement and sensor priority. The values used for simulation is shown in Table 3 itself. The energy consumed while transmission for node/master can be calculated using (5) and while receiving from (6). The number of bits for the node while transmitting data is 18 every 10 seconds, 36 every 30 seconds, 54 every minute and 72 for every 2 minutes. Figure 7 shows the energy consumption of the master and the node. The rate of energy consumption of the node has now reduced because of less number of data bit transmissions. The master also consumes less energy.

$$\begin{aligned}&Energy\ for\ transmission = Ec\ (Transition\ sleep\ to\ wake) +\\ &No.\,of\ bits * Et\ (Transmission\ of\ bits) + Ec\ (Transition\\ &wake\ to\ sleep)\end{aligned} \tag{5}$$

$$\begin{aligned}&Energy\ for\ \ reception = \ \ Ec\ (Transition\ \ sleep\ \ to\ wake) +\\ &No.\,of\ bits * Er\ (Reception\ \ of\ bits) + Ec\ (Transition\ wake\\ &to\ sleep)\end{aligned} \tag{6}$$

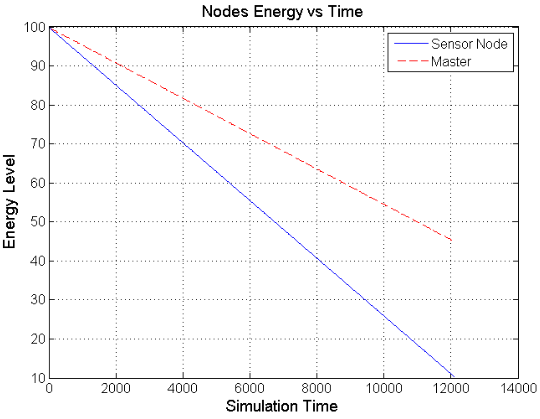


Fig. 6. Energy plot of periodic data transfer scheme

5.3 Event Based Data Transfer

For simulation of this scheme, the actual data from the priority periodic scheme was collected for a local environment. Table 5 shows the sensors used to collect the data. The collected data was taken as a reference for simulation of this scheme. In fact, deliberately while collecting the data, parameters were influenced to have more variation, but in a normal scenario, the variation is going to be very less. Figure 8 shows the energy consumption of master and node for this scheme. It can be observed that the node consumes less energy compared to the previous scheme for transmitting same information. The master also consumes less energy. The energy plot shows that the node now consumes less energy than the master. This consumption of energy actually depends on the sensors connected to the node and the change in sensor readings. If the sensors have high variations in reading then this consumption may be more.

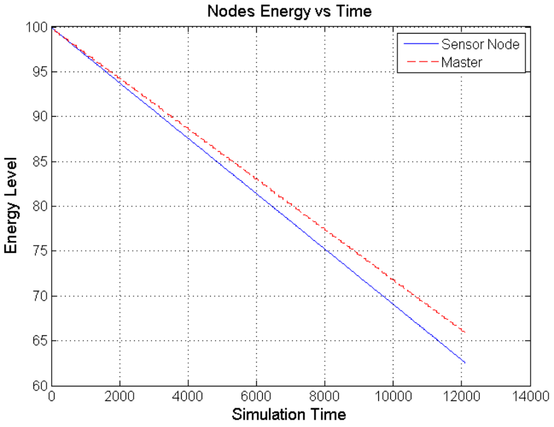


Fig. 7. Energy plot of priority periodic data transfer

Table 5. Node configuration for priority periodic scheme

Priority	Digital Sensor	Analog Sensor	No. of bits	Traffic Frequency
Highest	1 (Flame sensor)	1 (Current sensor)	18	10 sec.
High	1 (Temperature)	1 (LPG sensor)	18	30 sec.
Medium	1 (Humidity)	1 (Temperature)	18	1 min
Low	1 (Light intensity sensor)	1 (Soil Moisture)	18	2 min

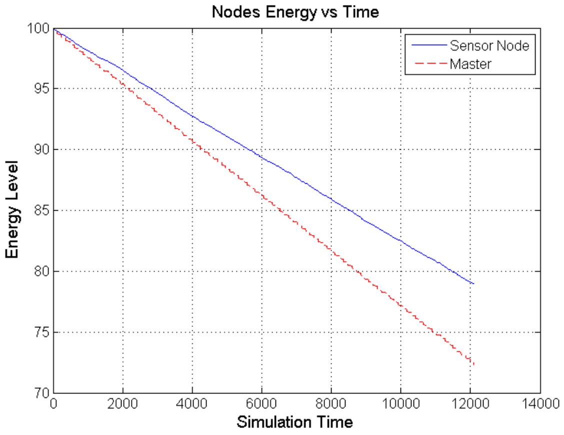


Fig. 8. Energy plot of event-based data transfer

Table 6 shows the comparison of the schemes and it is observed that the event-based scheme can provide a better data traffic reduction along with low energy consumption. In fact, as compared to priority periodic scheme the event-based scheme provided a reduction of around 44% in terms of energy requirement for node and 18% for master with load of 1:4. Also around 48% less data was required for transmission of the same amount of information.

Table 6. Comparative analysis of the data transfer schemes

Parameter	Periodic	Priority Periodic	Event Based
No. of bits sent by master	17776 bits	17776 bits	17776 bits
No. of bits sent by nodes	86472 bits	34272 bits	17891 bits
Percentage energy consumed by node	89.6284	37.4284	21.0474
Percentage energy consumed by master	55.0768	34.1968	27.6444

From Table 3 it is clear that this scheme adds one additional bit for data transfer whenever there is change in sensor value. The scheme reduces number of bits for no change in sensor value so it is beneficial for sensors with low probability of change in value. For sensors with high probability of change in sensor value, the scheme may underperform as compared to the priority periodic scheme. In order to analyze this situation a simulation is performed to see the number of bits generated by both the schemes with respect to the different probability of change in sensor values. For simulation a node with same configuration as described above is considered. The sensor values are taken as random variables. For change in analog sensor's value node sends 11-bits and for change in digital sensors value, it sends 9-bits. For no change in sensor value 1-bit is send for analog and digital sensors. P_i is defined as the probability of change in sensor value for ith sensor in the node. Each sensor value is modeled as a uniform random variable that can have a value in the range of 0 to 1 i.e. $X \sim U(0,1)$. So the number of bits required for sensor can be given as (7) for analog sensor and (8) for digital sensor.

$$N_{ba} = \begin{cases} 11 & ,\forall\, X_i < P_i \\ 1 & ,Otherwise \end{cases} \tag{7}$$

$$N_{bd} = \begin{cases} 9 & ,\forall\, X_i < P_i \\ 1 & ,Otherwise \end{cases} \tag{8}$$

The node has eight sensors four digital and four analog. Every sensor can have a different probability of change. To model the node we consider $P_1 = P_2 = \dots P_8$ and then the node is simulated for 200 minutes with P_1 varying from zero to one. Figure 9 shows the plot of probability of change versus total number of bits generated by both the scheme. The number of bits for the priority periodic scheme is same for the same amount of time and it does not depend on P_1 whereas, the number of bits increases with high probability of change for the event-based scheme. It is clear from the Fig. 9 that the event-based scheme can be preferred over the priority periodic scheme for sensors with probability of change less than 0.53.

So in a large network the system can be deployed with priority periodic scheme and then the data can be collected. After collecting the data, probability of change for all the sensors can be calculated and then depending on this probability different schemes can be adopted for different sensors.

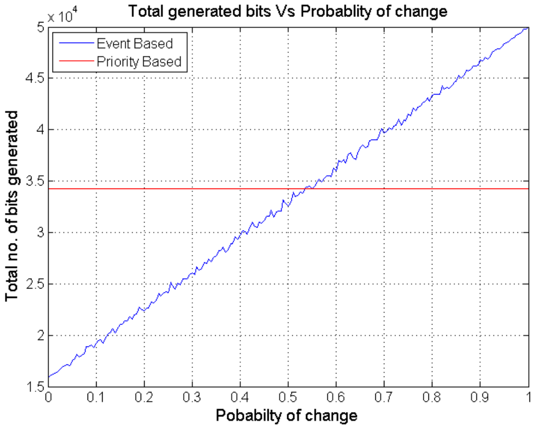


Fig. 9. Plot of total number of bits generated versus probability of change in sensor value

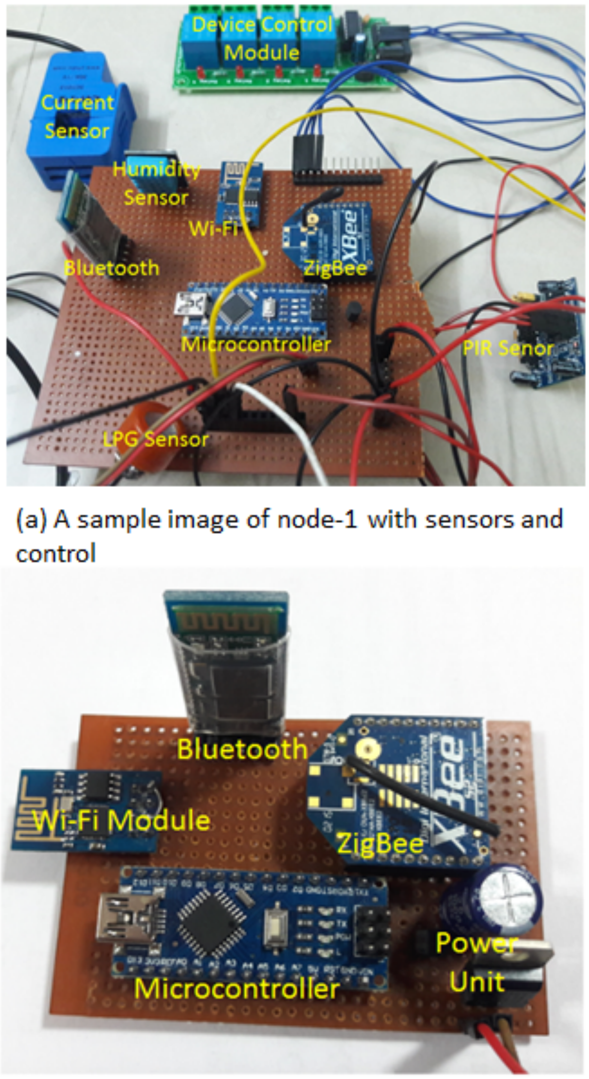
6 Implementation and Reliability

The complete system was implemented using four nodes, one master and two repeaters. The nodes are designed using Arduino Nano [21] with ZigBee, Wi-Fi and Bluetooth modules [22, 23, 24] for multiple wireless support. ZigBee is used for local network formation of nodes with repeaters and master. Wi-Fi and Bluetooth modules are available for connecting a node to other devices if required. To support wireless programmability ZigBee is connected to the serial programmable pins of the microcontroller [25].The network data transmission is performed though ZigBee. A special character sequence is used of switch from data transmission mode to programming mode. This character sequence is avoided while transmission of sensor data by sending a one unit smaller value than the actual value. This error in the transmitted value is very small for 8/10-bit of sensor data and can be exploited for wireless programmability of the network. Figure 10 shows the two sample nodes used for experiment.

The network is configured with two nodes sending data to the repeater and then repeaters sending data to the master. Different sensors were used with different priorities as shown in Table 5. All the three data transfer schemes were implemented. The master sent the data to the cloud through RESTful [26] application programming interface. A cloud server was designed to collect all the data. Figure 11 shows the first 100 samples received at different times on the cloud server for some sensors.

$$Reliability\ in\ \% = \frac{Total\ no.\ of\ data\ packets\ received}{Total\ no.\ of\ data\ packets\ generated} x100 \tag{9}$$

To test the reliability of the architecture with different data transfer schemes each scheme was implemented and the number of data packets generated and actually received on the cloud was calculated for 200 minutes. The count of data is taken in terms of data packets. Each data packet consist of the all the bits that are required to be sent to the cloud, so the length of data packet varies in accordance with the data transfer scheme. For periodic data transfer the length of packet is 72 bits, for priority periodic the length can be 18/36/54/72, and for Event based it can be 2/4/6/8/9/11/18/22/27/36/44/54/72 depending upon the event. Table 7 shows the reliability of the schemes in terms of percentage. Here reliability is defined as given by (9). It is evident that event based scheme was most reliable with less number of bits to be transmitted. Around 80% less data was generated for event-based scheme with reference to the normal periodic scheme as shown in Table 6. The event-based scheme used 48% less bits to transfer the same information as compared to the priority periodic scheme with 99.5% reliability.



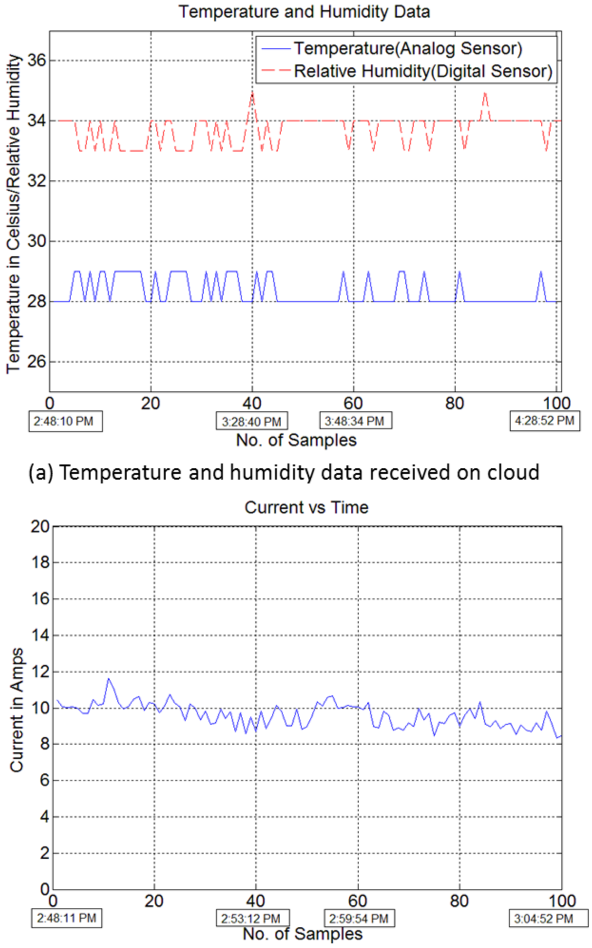
(a) A sample image of node-1 with sensors and control

(b) A sample image of node-2

Fig. 10. Sample network nodes

Table 7. Reliability table for 200 minutes of data received on cloud

Parameter	Periodic	Priority Periodic	Event Based
No. of packets generated at nodes	1201 (86472 bits)	1201 (34272 bits)	1201 (17891 bits)
No. of packets received at cloud	1062	1182	1195
No. of packets lost	139	19	6
Reliability in %	88.43	98.42	99.5



(a) Temperature and humidity data received on cloud

(b) Current sensors data received on cloud

Fig. 11. Data received on cloud server

7 Conclusion and Future Scope

The architecture provides a general approach for integration of various devices in an IoT environment. The hardware nodes with multiple wireless protocol support increases its connectivity and usability. The amount of data saved by using the proposed event-based scheme transmission can heavily reduce the network traffic. Even a very slow internet connection can be used to integrate the system with very minimum loss of information. The scheme will always send the most critical data with highest priority to the cloud. The over the air node configuration provides flexibility for network upgrade. In future, as the cloud receives more data over time it can start learning the behavior and patterns of the use of devices and can reconfigure them for optimum use and smart automation of various processes in the environment.

8 Acknowledgment

The work was financially supported by Department of Electronics & Information Technology under Ministry of communications and IT, Government of India and Center of Excellence on Combedded systems, Visvesvaraya National Institute of Technology, Nagpur, India. The authors would like to thank Dr. M. F. Hashmi for his suggestions to improve the quality of the paper, and Prasad Raspaile for his suggestions in improving the hardware design.

9 References

[1] D. Evans, "The internet of things," How the Next Evolution of the Internet is Changing Everything, Whitepaper, Cisco Internet Business Solutions Group (IBSG), vol. 1, pp. 1-12, 2011.

[2] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer networks, vol. 54, pp. 2787-2805, 2010. https://doi.org/10.1016/j.comnet.2010.05.010

[3] I. Chiuchisan, H.-N. Costin, and O. Geman, "Adopting the internet of things technologies in health care systems," in Electrical and Power Engineering (EPE), 2014 International Conference and Exposition on, 2014, pp. 532-535. https://doi.org/10.1109/icepe.2014.6969965

[4] L. Y. Mano, B. S. Faiçal, L. H. Nakamura, P. H. Gomes, G. L. Libralon, R. I. Meneguete, et al., "Exploiting IoT technologies for enhancing Health Smart Homes through patient identification and emotion recognition," Computer Communications, 2016. https://doi.org/10.1016/j.comcom.2016.03.010

[5] A. Azzara, M. Petracca, and P. Pagano, "The icsi m2m middleware for iot-based intelligent transportation systems," in 2015 IEEE 18th International Conference on Intelligent Transportation Systems, 2015, pp. 155-160. https://doi.org/10.1109/itsc.2015.34

[6] S. K. Datta, R. P. F. Da Costa, J. Hýrri, and C. Bonnet, "Integrating connected vehicles in Internet of Things ecosystems: Challenges and solutions," in World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2016 IEEE 17th International Symposium on A, 2016, pp. 1-6. https://doi.org/10.1109/wowmom.2016.7523574

[7] K. Taylor, C. Griffith, L. Lefort, R. Gaire, M. Compton, T. Wark, et al., "Farming the web of things," IEEE Intelligent Systems, vol. 28, pp. 12-19, 2013. https://doi.org/10.1109/MIS.2013.102

[8] A. Kaloxylos, R. Eigenmann, F. Teye, Z. Politopoulou, S. Wolfert, C. Shrank, et al., "Farm management systems and the Future Internet era," Computers and electronics in agriculture, vol. 89, pp. 130-144, 2012. https://doi.org/10.1016/j.compag.2012.09.002

[9] N. Bui, A. P. Castellani, P. Casari, and M. Zorzi, "The internet of energy: a web-enabled smart grid system," IEEE Network, vol. 26, pp. 39-45, 2012. https://doi.org/10.1109/MNET.2012.6246751

[10] S. Hiremath, G. Yang, and K. Mankodiya, "Wearable Internet of Things: Concept, architectural components and promises for person-centered healthcare," in Wireless Mobile

Communication and Healthcare (Mobihealth), 2014 EAI 4th International Conference on, 2014, pp. 304-307. https://doi.org/10.4108/icst.mobihealth.2014.257440

[11] L. Mainetti, V. Mighali, and L. Patrono, "An IoT-based user-centric ecosystem for heterogeneous Smart Home environments," in 2015 IEEE International Conference on Communications (ICC), 2015, pp. 704-709. https://doi.org/10.1109/ICC.2015.7248404

[12] M. V. Moreno, J. Santa, M. A. Zamora, and A. F. Skarmeta, "A holistic IoT-based management platform for smart environments," in 2014 IEEE International Conference on Communications (ICC), 2014, pp. 3823-3828. https://doi.org/10.1109/ICC.2014.6883917

[13] S. Helal, "Programming pervasive spaces," IEEE Pervasive Computing, vol. 4, pp. 84-87, 2005. https://doi.org/10.1109/MPRV.2005.22

[14] Y. Xu and A. Helal, "Scalable Cloud–Sensor Architecture for the Internet of Things," IEEE Internet of Things Journal, vol. 3, pp. 285-298, 2016. https://doi.org/10.1109/JIOT.2015.2455555

[15] I. Ganchev, Z. Ji, and M. O'Droma, "A generic IoT architecture for smart cities," in Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014). 25th IET, 2013, pp. 196-199.

[16] C. K. Dehury and P. K. Sahoo, "Design and implementation of a novel service management framework for IoT devices in cloud," Journal of Systems and Software, vol. 119, pp. 149-161, 2016. https://doi.org/10.1016/j.jss.2016.06.059

[17] S. D. T. Kelly, N. K. Suryadevara, and S. C. Mukhopadhyay, "Towards the implementation of IoT for environmental condition monitoring in homes," IEEE Sensors Journal, vol. 13, pp. 3846-3853, 2013. https://doi.org/10.1109/JSEN.2013.2263379

[18] T. Yashiro, S. Kobayashi, N. Koshizuka, and K. Sakamura, "An Internet of Things (IoT) architecture for embedded appliances," in Humanitarian Technology Conference (R10-HTC), 2013 IEEE Region 10, 2013, pp. 314-319. https://doi.org/10.1109/r10-htc.2013.6669062

[19] Z. Shelby, K. Hartke, and C. Bormann, "The constrained application protocol (CoAP)," 2070-1721, 2014.

[20] V. Rao, P. Priyesh, and S. Kar, "Pratham: A Low-Cost Wireless Node with Programmable Radio Range and Over-The-Air Programming Capability for Resource-constrained Applications," in Electronic System Design (ISED), 2014 Fifth International Symposium on, 2014, pp. 49-53. https://doi.org/10.1109/ised.2014.18

[21] “Arduino–ArduinoBoardNano”, https://www.arduino.cc/en/Main/ ArduinoBoardNano, accessed: 25 Oct 2016.

[22] Digi International, “XBee®/XBee‐PRO® RF Modules ‐802.15.4 ‐ v1.xEx”, Datasheet, September 2009.

[23] Espressif Systems, “Espressif smart connectivity platform: ESP8266”, Datasheet, October 2013.

[24] ITead Studio,”HC-05-Bluetooth to Serial Port Module”, Datasheet, June 2010.

[25] Faludi, Robert. “Building wireless sensor networks: with ZigBee, XBee, arduino, and processing.” O'Reilly Media, Inc.", 2010.

[26] M. Garriga, C. Mateos, A. Flores, A. Cechich, and A. Zunino, "RESTful service composition at a glance: A survey," Journal of Network and Computer Applications, vol. 60, pp. 32-53, 2016. https://doi.org/10.1016/j.jnca.2015.11.020

10 Authors

Varun Tiwari is with Visvesvaraya National Institute of Technology, Nagpur, India (varun.etrx@gmail.com).

Avinash Keskar is with Visvesvaraya National Institute of Technology, Nagpur, India (agkeskar@ece.vnit.ac.in).

NC Shivaprakash is with Indian Institute of Science, Bangalore, India (shiv@iisc.ernet.in).

Article submitted 28 November 2016. Published as resubmitted by the authors 13 January 2017.