www.ijaceeonline.com ISSN: 2456 - 3935
International Journal of Advances in Computer and Electronics Engineering
Volume 3, Issue 3, March 2018, pp. 1–7

Design and Implementation of a Soil Moisture and Temperature Monitoring using Wireless Sensor Network

Kandasamy Varatharajalu
Associate Professor, Electrical and Electronics Engineering
Kumaraguru College of Technology, Affiliated to Anna University,
Coimbatore, Tamilnadu, India
Email: kandasamy.v.eee@kct.ac.in

Divya Ragu
PG Scholar, Electrical and Electronics Engineering
Kumaraguru College of Technology, Affiliated to Anna University,
Coimbatore, Tamilnadu, India
Email: divyadivi28995@gmail.com

Abstract: *This paper presents the design of a shortest path tree based data collection wireless sensor network to sense soil moisture, soil temperature and relative humidity. Each node in the network follows a synchronized, periodic sleep - wake up schedule in order to maximize the lifetime of the network. The paper also demonstrates the implementation of a cluster based sensor network consisting of 10 nodes grouped as 3 clusters which collects and sends sensed data to a base station (here laptop) through a sink node. Data processing and presentation on the laptop is achieved using custom designed graphical user interface (GUI) in MATLAB. Nodes are kept approximately 8m apart covering an area of 400 m2. The implemented sensor network features using in house developed, low cost dual probe heat pulse (DPHP) soil moisture sensor, soil temperature sensor and energy efficient wireless transmission using clustering architecture for agricultural application. Field testing for 20% moisture level is demonstrated while feasibility of measuring the moisture levels from 0% to 40% is established from lab experiments.*

Keyword: *Wireless; Soil Moisture; Cluster; Graphical User Interface (GUI); Dual Probe Heat Pulse (DPHP).*

1. INTRODUCTION

Wireless sensor network (WSN) is one of the emerging technologies which finds application in variety of fields such as environmental and health monitoring, battle field surveillance, and industry process control [1]. Sensor networks consist of a large number of sensor nodes, which are normally deployed in an ad-hoc manner and they coordinate among themselves to perform a sensing task. The design of a WSN focuses mainly on extending the lifetime of the system since nodes work on battery while energy constraints are secondary criteria to the traditional wireless networks like cellular networks [2]. The architecture of WSN should be chosen in such a way that the network will be efficient in terms of energy consumption and should yield maximum lifetime for the network [3]. Soil moisture, soil temperature and relative humidity are major parameters which play a crucial role in the field of precision agriculture. Monitoring of these parameters is essential to enhance the crop productivity through irrigation management and by applying fertilizers in proper time intervals. High soil temperature destroys the crops and low temperature prevents the roots from absorbing water from the field. Similarly, it is important to measure the soil moisture at regular intervals because low moisture adversely affects the crops. Relative humidity (RH) is another parameter which indirectly affects photosynthesis and the plant growth. High RH reduces carbon dioxide uptake in the plants [4].

Many researchers have reported sensor network implementations for agricultural applications. [5] compares the performance of random and grid topologies of sensor network for precision agriculture through simulations in OPNET [6]. They measure the performance in terms of parameters like delay and throughput. However practical implementations and results are not discussed. [7] Discusses about an irri-

Cite this paper:

Kandasamy Varatharajalu, Divya Ragu, "Design and Implementation of a Soil Moisture and Temperature Monitoring using Wireless Sensor Network", International Journal of Advances in Computer and Electronics Engineering, Vol. 3, No. 3, pp. 1-7, March 2018.

gation management system in which authors have implemented a star network using Wasp mote nodes by Libelium. [8] also presents a star WSN deployment system for irrigation using ZigBee technology. Star architecture is not a suitable choice for sensor networks when scalability is considered.

n this paper we propose a methodology for tree based data collection which can improve the overall lifetime of the network. In addition, we have implemented a cluster based architecture which is easily scalable [9]. System implementation of the presented network in this paper is cost effective since it uses in house developed sensors and energy efficient due to low transmission power. Clustering architecture makes the presented network scalable to different sizes. We also analyze the current consumed by the transmitter during data transmission. The field deployment results are discussed in further sections.

2. NETWORK ARCHITECTURE

Our network consists of n nodes, each consisting of soil moisture, soil temperature or humidity sensor which are connected to a TelosB wireless sensor mote. Each node periodically senses the soil moisture or relative humidity or soil temperature, and the data collected from each node need to be transported to a sink node. Here we propose a network architecture which can be used to achieve this goal. To increase the lifetime of the network, the sensing motes follow a periodic sleep and wake up schedule. The various operations occurring in the network are shown in Fig. 1. The nodes are in energy saving sleep mode for most of the time in comparison with other stages (for example, one sleep period = 26 minutes, one wakeup period = 4 minutes and the total time between consecutive “data collection tree formation” phases = 4 days).

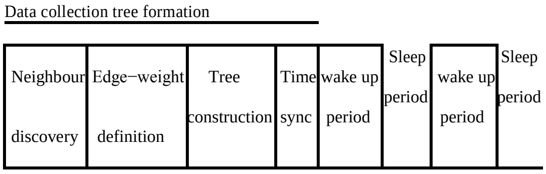


Fig 1. Sequence of operations occurring in the network

During the wake up period, each node senses either soil moisture or soil temperature or relative humidity depending upon the sensor attached to it and transfers the data to the sink node along a shortest path tree (SPT) to minimize the energy consumption as explained in section II-B. The sensor nodes will be in sleep state for most of the time. When a node becomes active (wake up) to initiate the transmission of a message, the destination node should also come into active state from sleep mode. To coordinate this, nodes need to be periodically synchronized as explained in section II-C. Over a period of time the nodes nearer to the sink node will have more battery drainage since it has to relay a lot of data coming from the nodes which are at a lower level in the tree. If proper precautions are not taken, the nodes nearer to the sink node will die at an earlier stage than the nodes in the bottom levels. To make the energy consumption of all the nodes uniform, the SPT has to be recomputed considering the fact that the nodes with more energy will be selected as the parent node with more number of children.

Data collection in a sensor network mainly consists of two steps, tree construction and scheduling of the nodes. Currently we concentrate on various techniques to build an energy efficient data collection tree so that the lifetime of the network can be extended. The basic operations in the tree construction phase are proposed below.

2.1 Neighbor discovery

Since the placement of nodes in the field is considered as random, none of the nodes have any information about any other nodes in the network. Hence as a part of building the wireless sensor network, each node needs to discover its neighbors.

In this phase, each node broadcasts a neighbor discovery message (beacon) and tries to find out its 1-hop neighbors. The message will be transmitted periodically with the maxi-mum transmission power and the nodes receiving the broadcast message update its neighbor list with the source id of the received message. To minimize collisions, the periodic message transmission at each node is initiated after a random delay. At the end of this phase each node will have an updated neighbor list. Neighbor discovery is performed periodically as a part of data collection tree formation as shown in Fig. 1, in order to consider the changes in the neighbors of a particular node which can happen either due to the addition of a new node into the network or due to the removal of a particular node because of various reasons like wireless communication failure, complete drainage of battery etc.

2.2 Edge weight assignment and tree construction

This phase assumes that each node has the knowledge of its 1-hop neighbors. The aim of this phase is to assign an edge weight to all the communication links which exists between any pair of nodes in the network. The edge weight is obtained as a function of node’s current battery level and the transmission power required delivering a packet to the corresponding neighbor.

As shown in Fig. 2, nodes B, C, D are neighbors of node A in the increasing order of distance from node A. To find e_{ab}, A transmits a packet to B with maximum transmission power and B finds the received signal strength indicator (RSSI) which is a measure of strength of the received radio signal of the packet at B. Note that RSSI is a function of the distance between A and B, shadow and multipath fading, and typically decreases as the distance between A and B increases. The above procedure is repeated several times and the average RSSI at B is computed. Node A calculates e_{ab} as a function of the average RSSI and the present battery levels of nodes A and B. In the same fashion node A finds the edge weights to all of its neighbours.[1]

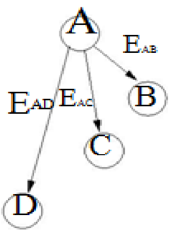


Fig 2. Node A and its neighbors with the respective edge weights. From node A, the distance to node D > the distance to node C > the distance to node B and hence $e_{ad} > e_{ac} > e_{ab}$

At this stage, each node has the details about its neighbouring nodes and corresponding edge weights which needs to be transported to the sink node to build the data collection tree. This is done through simple flooding [11]: each node in the network broadcasts its neighbour-list and corresponding edge weights, each recipient of a broadcast packet, until the sink node receives the packet. At the end of this flooding process, the sink node has complete information about the topology and edge weights in the entire network.

Once the complete information about the network is avail-able with the sink node, it finds the data collection tree using Dijkstra’s algorithm [12]. Dijkstra’s algorithm is used to find the shortest path from every node in the network to the sink node. The union of all the shortest paths results in a shortest path tree rooted at the sink node. If the edge weight equals the energy needed for unit packet transfer, it is shown in that, for raw-data coverage cast, in which the entire sensed data needs to be sent to the sink node without fusion at intermediate nodes, routing the packets through the shortest path tree minimizes the total energy across all nodes, needed to deliver the packets to the sink node. In our algorithm, since the edge weight is also dependent on the residual energy of nodes, the rate of energy consumption at different nodes is more uniform.

2.3 Time synchronization

The sensor nodes, which are in sleep state for most of the time periodically wakes up, senses the parameter and transmits the data to its parent node in the data collection tree. To have a coordinated sleep and wake up schedule between the transmitting and the receiving node, time synchronization between the nodes along the edges of the data collection tree is essential. Since the time accuracy to within fractions of seconds is generally acceptable in sensor networks, we can use any lightweight (in terms of energy) protocol. Lightweight tree-based synchronization (LTS) [14] algorithm is an example.

LTS algorithm implements multi hop time synchronization as an extension of pair-wise synchronization along the edges of the data collection tree which is a spanning tree. Pair-wise synchronization requires an exchange of three message transfers and multi hop synchronization of n nodes require n 1 pair-wise synchronizations. Time synchronization is performed periodically, after every tree construction phase as shown in Fig. 1. This periodic time synchronization compensates the error due to clock drifts which occurred during the last cycle. Once the data collection tree is built and synchronized, each child node sends the data to its parent node. After collecting data from all the children, the parent node forwards the collected data after appending its own data to its parent node. Thus the sink node collects the complete sensed data from all the nodes in the network.

3. IMPLEMENTED NETWORK WITH NODE ARCHITECTURE

In house developed soil temperature sensor (T-type thermo-couple) is inserted in soil to measure the soil temperature. Soil moisture measurement is performed using DPHP technique. The DPHP sensor consists of two probes viz. heater probe and temperature sensor probe. A DC pulse of fixed voltage is applied to the heater probe of DPHP and temperature sensor probe (T-type thermocouple) is used to measure the rise in the soil temperature. This rise in temperature decreases with increase in soil moisture content [15]. A look up table was generated for DPHP sensor which shows different rise in temperature for different moisture values. This look up table is fed in MATLAB which map the rise in temperature value to its corresponding moisture. Humidity is measured using TelosB’s on-board digital sensor, Sensirion SHT11 [16].

IIT Bombay nursery premises is used to perform the in-situ experiments. We have deployed a sensor network consisting of 10 TelosB motes (n = 10) as shown in Fig. 3, to sense soil moisture, soil temperature and humidity. The numbers in Fig. 3 represent the node address. Node 1 acts as the sink node which collects periodic data from all the other nodes. The

---

[1] Edge weight computation can be combined with neighbour discovery for better efficiency.

nodes are grouped as clusters, indicated by the dotted circles. Each cluster has 2 leaf nodes which will be active once in every thirty minutes. When not active, the nodes are in an energy-saving sleep state. During the active period, the leaf nodes transmit the sensed parameters to the respective intermediate nodes. The intermediate node appends its own data with the information collected from its children and forwards to the sink node. Sink node is connected to a base station (laptop), which processes the collected information using MATLAB. The processing part will be explained in section VI.

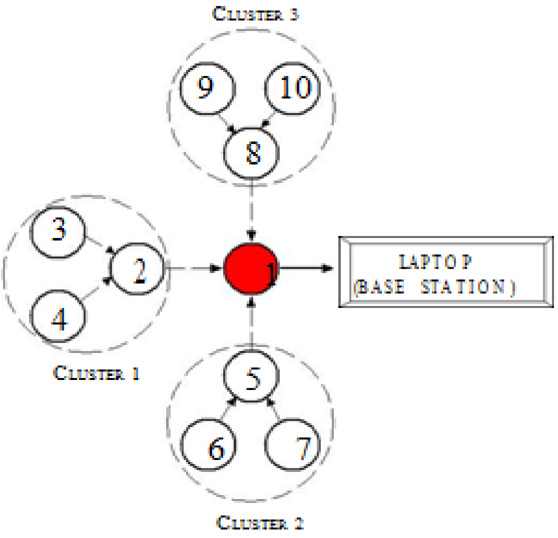


Fig 3. Architecture of the implemented sensor network

The internal structure of a node is shown in Fig. 4. Each node in a cluster is equipped with either DPHP or soil temperature or humidity sensors.

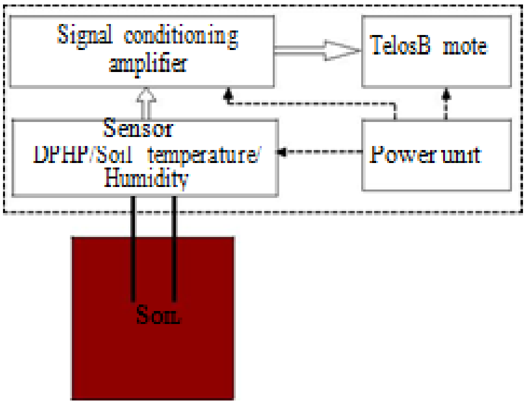


Fig 4. Internal structure of a single node

Crossbow's TelosB mote is an open source platform, extensively used by the research community for low power sensor network applications [16]. We have used TinyOS as the operating system (OS) which is an open-source and energy-efficient OS developed by University of California, Berkeley [17].

TelosB's on-board digital sensor, Sensirion SHT11 is used as the humidity sensor in the default 12-bit resolution mode. The relative humidity is measured using the humidity interface provided by the Tiny OS top level component Sensiri-onSht11C of TelosB platform. The digital data (SO_{RH}) read by the humidity interface can be mapped to relative humidity using (1) [18].

$$RH_{linear} = c_1 + c_2 \quad SO_{RH} + c_3 \quad SO_{RH}^{2} \text{ (\%RH)} \qquad (1)$$

where $c_1 = 2{:}0468$; $c_2 = 0{:}0367$; $c_3 = 1{:}5955\text{x}10^{6}$
Both soil moisture and soil temperature sensor use AD8494 which is a precision instrumentation amplifier with thermo-couple cold junction compensators on an integrated circuit and are used to amplify the output of thermocouple [20]. The relationship between temperature and output voltage of thermocouple is given by (2) [20].

$$T = \frac{\text{Vout1Vref}}{\text{5 mV= C}} \qquad (2)$$

V_{ref} = 0V since V_{ref} is connected to ground.

To power up the system, we have used Lithium-ion batteries of 4000 mAh rated at 3.7 Volts. The system is designed such that all the modules (wireless, instrumentation, sensors) work at 3.3 Volts. TPS 63001 buck boost regulator from Texas instruments has been used to obtain the regulated output [21].

4. POWER MEASUREMENTS

In TelosB, the micro-controller MSP430 controls CC2420 (RF transceiver) through serial peripheral interface (SPI). The output power of CC2420 is programmable so that it can be applied to various energy saving schemes.

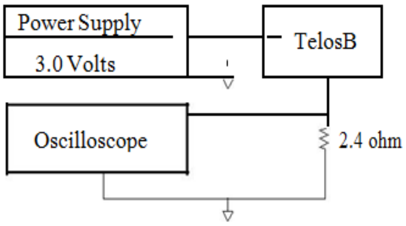


Fig 5. Set up to measure current consumption of TelosB mote

For example, the transmission power of a given transmitter can be set to the smallest value such that its signal is received by the intended receivers. We have used Tiny OS as the platform to perform our experiments [22]. We have measured the maximum communication range between a pair of nodes (one acts as transmitter and another as receiver) by varying transmission power levels of the transmitter in our experimental field. Table I shows the transmission distance with the corresponding power level. The Table also compares the current consumption in each power level with the corresponding values given in the CC2420 datasheet. The transmission power is varied using the ’set Power’ command provided by the CC2420 packet interface in Tiny OS. Current consumption is measured by monitoring the voltage across 2.4 resistors using an oscilloscope (Tektronix, DPO4104B) as shown in Fig. 5.

TABLE I. COMMUNICATION RANGE VS TRANSMISSION POWER

Transmission Power (dBm)	Current consumption (mA)		Communication Range (m)	
	As per datasheet	Measured	LOS	non - LOS
- 25	8.5	10.00	12.0	8.5
- 15	9.9	10.83	15.8	12.0
- 7	12.5	13.33	29.8	21.5
- 3	15.2	15.83	-	49.0
0	17.4	18.33	-	56.5

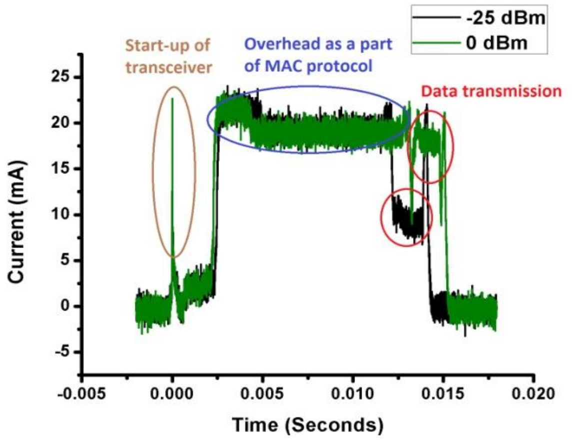


Fig. 6. Current through 2.4 resistors at different transmission power levels

Fig. 6 shows the current consumption by TelosB with transmission power levels as 0 dBm and -25 dBm when a single packet of 15 bytes is transferred. It is clear from the figure that most of the power is consumed as an overhead due to the default medium access control (MAC) protocol implementation of Tiny OS. The initial spike in Fig. 6 is due to the start up of the transceiver. Keeping the RF transceiver in the off state leads to energy savings and thus improves the lifetime of the network.

5. NETWORK LIFETIME CALCULATION

Table II shows the current consumption details of various elements of the sensor node. From the Table it is clear that the DPHP sensor consumes huge current in comparison with the other elements and hence the lifetime of the network is decided by the nodes which have DPHP sensor connected to it. Since DPHP sensors are connected to the leaf nodes in the implemented cluster based network, we will consider its lifetime as the lifetime of the network. A leaf node periodically transmits the sensed data to its parent node during the active period and then goes into sleep state. The energy usage of a leaf node, E can be considered as a combination of energy consumption for sensing the data (E_{sense}), energy used for wireless data transmission (E_{txn}), and energy consumption during the sleep stage (E_{sleep}).

$$E = E_{sense} + E_{txn} + E_{sleep} \qquad (3)$$

Each sensor mote is in active state for a period of 4 minutes in every 30 minutes interval. During the active period, a node sends its sampled data to its parent node once in every one second (the DPHP heater probe remains in the ON stage only for a period of 2 minutes). The rate of sensing, r = (4 60)=(30 60) = 4=30 and the rate of packet trans-mission, g = (60 4)=(30 60) = 4=30. All the energies are considered in milli joules per second and the entire system works on 3.3 Volts.

TABLE II CURRENT CONSUMPTION OF SENSOR MOTE ELEMENTS

Element	**Current consumption (mA)**	**Time (s)**
DPHP sensor (Heater probe - ON)	100	2*60
DPHP sensor (Heater probe - OFF)	Negligible	2*60
Signal conditioning and instrumentation for the sensor output signal	10	4*60
Humidity sensor	1	0.100
Temperature sensor	Negligible	0.080
TelosB during wireless transmission	22	0.020
TelosB in sleep state	0.060	26*60

Therefore from (4), a node with a battery capacity of 4000 mAh has a lifetime of approximately 20 days which can be considered as the network lifetime.

Lifetime = 3:3V 4000mAh 60 60 / E (4)

6. DATA PROCESSING

A laptop has been used as the base station of the network, which receives the data from the sink node through its serial port. The data processing algorithm contains two scripts which run in parallel in MATLAB as shown in Fig. 7. The first one monitors the serial port continuously for any data from the sink node and stores all the incoming data into a single file. The second script reads this file and extracts all the node addresses and their corresponding data. The segregated data is written into separate files with file name same as its node address. We have developed a Graphical User Interface (GUI) as shown in Fig. 8. It provides six different buttons along with the current state. After running the first script, the GUI is launched which provides various options. Start will start the second script i.e segregation of data into respective files while Stop will stop the segregation of data. Reset will create a backup of current data and then clear all the data from the files.

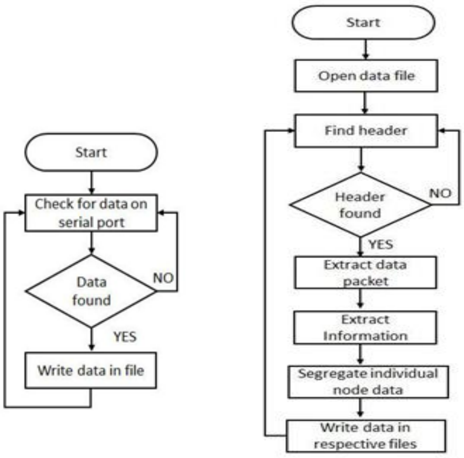


Fig. 7. Data processing algorithms

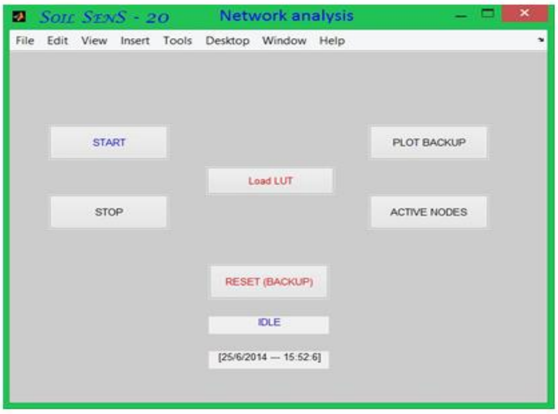


Fig. 8 Soil moisture GUI

Load LUT helps in loading the look up table (LUT) file into MATLAB for the given soil type. Active Nodes will launch another GUI with buttons corresponding to each of the segregated data file with name same as node's address. Clicking on a particular node will plot the data received from that node. Plot backup helps in plotting the backup data files.

7. FIELD MEASUREMENTS AND DATA ACQUISITION

The sensor network described in section III (see Fig. 3) is placed in an area of 400 m^2 as shown in Fig. 9. All the nodes in a cluster are kept at 8m distance apart. Fig. 10a shows the sensor node equipped with DPHP and Fig. 10b shows the node equipped with temperature sensor. Sensors are activated once in every 30 minutes interval. Data obtained from a single cluster is plotted in MATLAB GUI as shown in Fig. 11. DPHP soil moisture sensor data is mapped with LUT to measure the soil moisture. Analog voltage obtained from soil temperature sensor is converted to temperature using (2). Similarly data from humidity sensor is converted into relative humidity (%RH) using (1). Soil moisture sensor is tested in lab for the same soil type. It is able to measure the moisture in the range of 0-40%.

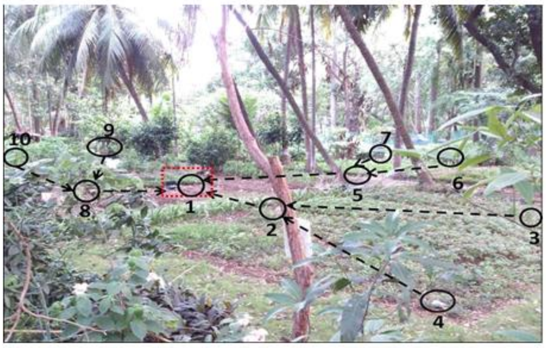
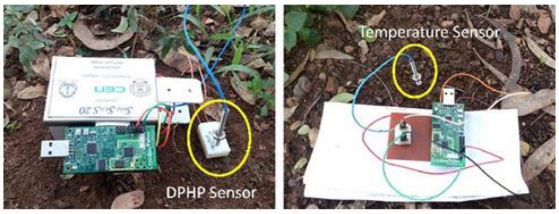


Fig. 9. Field deployment



(a) Node with DPHP sensor *(b) Node with temperature Sensor*

Fig. 10. Placement of nodes

8. CONCLUSION

In this paper we have proposed a shortest path tree based data collection network architecture and demonstrated implementation of a cluster based, low cost and low power wireless sensor network to monitor soil moisture, soil temperature and relative humidity in an area of 400 m^2 using 10 TelosB motes. Cluster based architecture eliminates long distance transmission resulting in energy savings while in house development of sensors makes the system low cost.

REFERENCES

[1] C. -Y. Chong and S. P. Kumar, "Sensor networks: Evolution, opportunities, and challenges", *Proceedings of the IEEE*, vol. 91, no. 8, pp. 1247 - 1256, Aug. 2003.

[2] I. F. Akyildiz et al., "A survey on sensor networks", *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102 - 114, Aug. 2002

[3] J. Zheng and A. Jamalipour, "Wireless Sensor Networks, A Networking Perspective", *IEEE e-book*, pp. 19-67, 2009.

[4] V. S. Palaparthy et al., “Review of Polymer-Based Sensors for Agriculture-Related Applications”, *ICE, Emerging Materials Research*, August 2013.

[5] M. Keshtgary and A. Deljoo, “An Efficient Wireless Sensor Network for recession Agriculture”, *Canadian Journal on Multimedia and Wireless Networks*, vol. 3, no. 1, pp. 1-5, 2012.

[6] OPNET Technologies, Inc. [Online]. Available: http://www.opnet.com

[7] M. Mafuta et al.,“Successful Deployment of a Wireless Sensor Network for Precision Agriculture in Malawi WiPAM”, *IEEE 3rd International Conference on NESEA*, Dec 2012.

[8] Y. Zhou et al.,‘ ‘A Wireless Design of Low Cost Irrigation System Using ZigBee Technology”, 2009 *International Conference on NSWCTC*, pp. 572 - 575, Apr. 2009.

[9] A. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks”, *Computer Communications*, vol. 30 , nos. 14-15, pp. 2826-2841, Oct. 2007.

[10] O. D. Incel, A. Ghosh, B. Krishnamachari, and K. Chintalapudi, “Fast data collection in tree-based wireless sensor networks,” *IEEE Trans. Mob. Comput*. vol. 11, no. 1, pp. 86 - 99, 2012.

[11] B. Williams and T. Camp, “Comparison of broadcasting techniques for mobile ad hoc networks,” *Proceedings of the 3rd ACM international symposium on Mobile ad hoc Networking & computing. ACM*, 2002.

[12] D. Bertsekas and R. Gallager, *Data Networks, 2nd ed. Upper Saddle River*, NJ, USA: Prentice-Hall, 1992.

[13] S. Upadhyayula and S. K. S. Gupta, “Spanning tree based algorithms for low latency and energy efficient data aggregation enhanced converge cast (DAC) in wireless sensor networks,” *Ad Hoc Networks*, vol. 5, no. 5, pp. 626-648, 2007.

[14] J. van Greunen and J. Rabaey, “Lightweight time Synchronization for sensor networks,” In WSNA03: *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, 2003.

[15] V. S. Palaparthy et al., “Soil Moisture Measurement System for DPHP Sensors and In Situ Applications”, *Proceeding of the 5th ISED*, pp. 11-15, Dec 2013.

[16] TelosB, Datasheet of sensor mote [Online]. Available: http://www.willow.co.uk/TelosB Datasheet.pdf

[17] TinyOS [Online]. http://www.tinyos.net/

[18] SensirionSHT11, Datasheet, Humidity sensor [Online]. Available: http://www.sensirion.com/en/products/humidity-temperature/humidity-sensor-sht11/

[19] AD8494, Thermocouple amplifier from analog devices[Online].Available:http://www.analog.com/static/imported-files/data sheets/AD8494 8495 8496 8497.pdf

[20] TPS 63001 : Datasheet of Buck-boost converter [Online]. Available: http://www.ti.com/lit/ds/symlink/tps63001.pdf

[21] CC2420: Datasheet of RF transceiver [Online]. Available: http://www.ti.com/lit/ds/symlink/cc2420.pdf

Authors Biography

Kandasamy Varatharaju is an Associate Professor of Department of Electrical and Electronics Engineering in Kumaraguru College of Technology, Coimbatore.

Divya Ragu is a PG Scholar pursuing Embedded System Technologies in the Department of Electrical and Electronics Engineering in Kumaraguru College of Technology, Coimbatore. I completed my bachelor’s degree in Government College of Engineering, Salem.

Cite this paper:

Kandasamy Varatharajalu, Divya Ragu, “Design and Implementation of a Soil Moisture and Temperature Monitoring using Wireless Sensor Network”, International Journal of Advances in Computer and Electronics Engineering, Vol. 3, No. 3, pp. 1-7, March 2018.