AN ENERGY EFFICIENT MAC PROTOCOL FOR WIRELESS SENSOR NETWORKS IN AUTOMATION ENVIRONMENTS

VINÍCIUS G. GUIMARÃES, ADOLFO BAUCHSPIESS AND RENATO M. DE MORAES.

Depto. de Engenharia Elétrica, Faculdade de Tecnologia, Universidade de Brasília
Caixa Postal 04386, 70910-900, Campus Universitário Darcy Ribeiro, Asa Norte - Brasília, DF, Brasil
E-mails: `vinicius_galvao@msn.com, adolfobs@unb.br, renatomdm@ene.unb.br`

Abstract— This article proposes a new data collision free medium access control protocol for sensor networks in automation environments based on time division multiple access which presents less energy consumption than the S-MAC protocol while maintain high data throughput. The new protocol employs single channel and a carrier sense approach. Simulation results are presented and show how the new proposal outperforms the S-MAC protocol.

Keywords— Sensor and Actuator Networks, Energy Efficiency, MAC Protocol, Wireless Automation.

1 Introduction

The design of an automation system may include the use of remote modules (or nodes) for wireless control and actuation, this can be determined by a lot of parameters, such as real time or relaxed delayed message delivery, necessary payload and battery utilization or not. Utilizing a wireless system dependent on battery may also require energy efficient paths. Previous works suggested improvements to save energy and so extend the module’s operation life. Working with a considerable number of wireless nodes may lead to the concept of sensor network and its issues for automation environments.

Sensor networks are a well-studied topic today and with the advance of technology, sensor systems will be even more present in human life, from the old large integrated circuits to small capsules micrometric ahead reaching the nano computation nowadays. A striking feature in this environment is the presence of batteries and the high dependence of energy.

On the other hand, the energy storage media has not followed this miniaturization at the same speed. Accordingly, even if the components are consuming less energy, in order to get a smaller battery, it is necessary to reduce the total storage capacity. This produces a demand by means of energy saving during the operation of nodes in the network.

There are several ways to save energy, such as the use of the best materials, avoiding unnecessary computation, remaining in economic state during idle times, among others. In addition, the communication task responsible for the medium access control (MAC) and physical layers needs to reduce energy consumption, since there may be large energy expenditures when transmitted packets collide and therefore are not captured by the receivers, or simply when a receiver keep catching unnecessary information until it receives a packet for this node. Therefore, the design of energy aware MAC protocols is an important issue for sensor networks.

This work develops a new MAC protocol designed for energy aware but quasi-real time automation environments for wireless sensor networks. For that, it does not present data collisions and is based on time division multiple access (TDMA) method (Falconer *et al.*, 1995) with low power consumption and high throughput. Simulations with parameters for real automation environments show how the new proposal has better performance than the S-MAC protocol which is a state-of-the-art MAC approach implemented on a real platform (Ye *et al.*, June 2002).

The remainder of this article is structured as follows. Section II presents this paper objectives. Section III explains the definitions for performance metrics. Section IV describes the new protocol and its properties. Section V expounds the parameters and network topology to evaluate the performance of the new and the S-MAC protocols. Section VI presents the simulation results followed by the last section that concludes the work.

2 Objectives

This work set definitions and parameters to evaluate MAC protocols using a specific topology to simulate a sensor network for automation environments. It also explains the new MAC protocol developed, with its most important characteristics that makes it reasonable for automation systems with relaxed delay message delivery, high throughput and energy efficient.

3 Definitions and Related Works

Each system has its characteristics and performance metrics that must be dimensioned to ensure its correct operation and will be used to evaluate these protocols in wireless sensor networks. They are presented as follow:

- *Throughput*: Average amount of data bytes per time transported from source to final destination;
- *Latency*: Average time for a given message to travel from source to final destination;
- *Overhead*: Average amount of control bytes (including headers and acknowledgments)

per time transported from source to final destination;

- *Energy consumption*: Energy consumed from the battery that powers the sensor node.

Among many applications of wireless sensor networks, automation environment is a very important one, which for some of its subsystems, as control of temperature in a room or an agricultural irrigation, has specific parameters.

According to Ferreira *et al*. (2009), it was proposed an intelligent automation environment for thermal comfort with energy saving utilizing wireless network sensor. Although good results were achieved, this work did not evaluated the impact of the MAC protocol utilized and so how changing its parameter would improve their results in terms of modules operation life time. In some cases, choosing the most suitable communication protocol that better fits the system parameters can improve the results, according to the criteria below:

- There is no need for high throughput, because the data generation is not very frequent;
- Possibility of high latency (in some cases, delays of minutes are tolerable);
- Low density of nodes;
- Low connection dynamics, that is, nodes tend to connect and remain there until exhaust their energy.
- High efficiency in energy savings for increased longevity.

Among the protocols currently available that fit relatively well to the above requirements, the Sensor-MAC (Ye *et al.*, June 2002), also known as S-MAC, is a consolidated system in energy-saving which was implemented on a real platform. It has features that contribute for the rationing, as the use of activation and sleep cycles (*duty cycles*), small packets for traffic control, which is similar to the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) (802.11, June 1997) that enable, in case of a collision, low energy loss.

According to Ye *et al*. (2002), the S-MAC protocol was developed to meet the need of energy saving, at the cost of increased latency. In order to address this problem, it was also developed the S-MAC-AL version (*Sensor-MAC with adaptive listen*) (Ye *et al.*, June 2004). However, this version surpasses the limits of activation cycles for latency improvement. This use of off-periods determined by the activation cycle is undesirable in some cases (as will be explained in the following sections) and for this reason it will be considered here only the original S-MAC version for comparison.

The operation of the S-MAC protocol does not prevent the collision of control packets which cause energy loss. Thus, this protocol also uses contention periods, determined by activation cycles (duty cycles), in which all nodes in a group are active to receive or send control packets. These characteristics are not appropriate for systems having message payload of the length of only a few bytes, because the period of contention can be much longer than the necessary to send the payload; hence, spending more energy to control the channel than to transmit data.

Another important protocol with good features for the previous listed requirements is the TDMA. This one does not have collisions and still provides a high throughput to the network. But its main drawback is the constant synchronization mechanism for each node, as well as the determination of the exact number of connected nodes once the network is running. This situation can still be worse when the network is not saturated with nodes transmitting, which produces a waste in channel utilization and reduction of throughput with increase in average latency, because the intervals (slots) of unused time are wasted.

Other protocols suggested in literature use schemes or parameters that do not allow a fair comparison with the S-MAC or the new protocol developed in this article, since they work with flexible periods of activation, as is case of the T-MAC (van Dam *et al.*, November 2003) or additional technologies such as TDMA-W (employing wake-up hardware) (Chen *et al.*, October 2004).

Based on S-MAC and TDMA protocols and their interesting features, such as activation cycles that reduce overhearing, formation of groups synchronized according to a schedule of activation, relaxed latency and no data collision by reserved time allocation for each node transmission, this article develops a new MAC protocol which emphasizes energy savings for wireless sensor networks.

4 Dynamic Timed Energy Efficient Protocol

Similar to S-MAC, the Dynamic Timed Energy Efficient (*DyTEE*) MAC protocol employs single channel and a carrier sense approach, as well as works with groups of nodes (clusters) that are associated to each other by means of an activation schedule and for a time determined by the activation cycle (duty cycle). However, analogous to TDMA, there is a coordinator node responsible for time scheduling and synchronization (by sending small control packets, i.e., beacons).

As reviewed in Huang *et al*. (2013), MAC sensor protocols may be classified by other aspects that help deciding an application. Four aspects are very important. *No synchronization* in which nodes access the channel without any time synchronism; *local synchronization* in which nodes form a group to communicate using one hop synchronization; *global synchronization* in which nodes keep two-hop synchronization and employ a frame-slotted structure; and *multi-channel operation* in which nodes can employ more than one channel for communication. Accordingly, the MAC protocol developed in this paper can be classified as *global synchronization* employing *single-channel operation*.

Another classification of sensor protocols is given by Suriyachai *et al*. (2012), which analyses features for mission-critical applications in which delay is a very strict constraint, using S-MAC as a fixed point of comparison. However, the intended application of the protocol proposed here is not delay aware and so mission-critical requirement is not considered. Nevertheless, considering the aspects described in Suriyachai *et al*. (2012), DyTEE can be classified to be delay decreased and node-to-node guarantee by worst-case delay. Also, DyTEE does not aim reliability, but mostly energy efficiency, which places it into the group of delay-tolerant and loss-tolerant protocols.

To start a network in DyTEE, nodes look for synchronization beacons from a coordinator, in order to join into an existing network, similar to the S-MAC protocol. Once a node listens to a beacon, it will exchange information with the coordinator and it will obtain an identification number for that particular group. If there is no group formed, nodes will try to form their own group, sending a beacon. Nodes already connected to a group keep the search for new groups periodically, in order to maintain connection information available for new groups.

The process of a node to start sending beacon signals (BS) is based on a wait period for previous BS signals, followed by a further random wait interval based on the amount of energy percentage available for that node (it is assumed that the node is able to measure its own energy load available). The higher the energy availability, faster the node will try to become the leader of a group. However, it is important to note that even if the node has 100% load available, the waiting time for sending a beacon will still be random, but most likely it happens before a node with 90% of battery load, for example.

The leadership of a group alternates among nodes according to a percentage use of the battery. That way, when a leader spends this particular amount of load, it announces the end of that group and a new competition among other nodes for the leadership occurs. Thus, avoiding excessive use of battery of a single node; thereby, providing equal energy consumption of batteries among nodes.

Figure 1 presents the simplified timing diagram of the DyTEE protocol. Right after beacons, the system has a moment to synchronize (SyncM) followed by activation cycles. During SyncM, each node try to announce its schedule which informs the destination intended for data packet transmission. The entire period for a node to send data and to receive (acknowledgement, i.e., ACK) is indicated in Figure 1 by PKT. Thus, a node only send data if it succeeds to inform its schedule. Accordingly, there is no collision of data packets. It is important to note that time slots are separated by empty small slots enough to avoid problems with clock drifts, such that according to Ye *et al.* (2004) if the nodes are synchronized in periods of 10 seconds, the difference between the clocks does not exceed 0.2 milliseconds per second (ms/s).

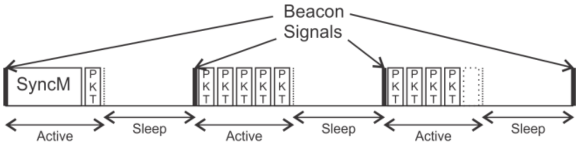

Figure 1. Simplified timing diagram of the DyTEE protocol.

Unlike the S-MAC, the DyTEE does not allow any node within a group to be active for longer than the active period determined by the duty cycle. Thus, if there are other groups nearby, this behavior avoids interference in the functioning of other adjacent groups. Note that adjacent groups will set their active schedule to operate during the sleep period of its neighbors, in this way multiple groups (clusters) can operate simultaneously within a given area.

The maximum time available for the SyncM is given by the maximum time of activation that each period may have, in other words, if the total period (active + sleep) of the group is t seconds, with a cycle of 10% activation, SyncM can last a maximum of $0.1t$ seconds. However, this moment is variable with the number of nodes connected to the group, such that nodes can begin to send their packets immediately after the end of SyncM, respecting the maximum active time.

4.1 *Structure of the synchronization moment*

According to Figure 2, it is observed that the synchronization moment has 4 main sub divisions, explained next.

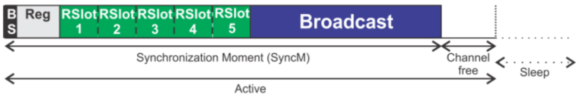

Figure 2. Timing diagram of a synchronization moment (SyncM).

At the beginning of the SyncM, named here as BS, a broadcast packet (or beacon signal or signal of coordination) is sent by the group's leader, which responsible for updating the clocks of all nodes associated with the group. The time is actually counted relative, that is, from this moment and on, nodes will count the time for all the following actions until the occurrence of the next BS signal.

The BS packet contains the identification number (ID) of the group and how many nodes are connected to that group at that time, similar to what is generally used in other protocols that need clock synchronization among other nodes, as TDMA. Following this, a given node that want to connect to the group, after receiving the signal in the BS period, being it already in another group or not, can perform a registration process with the leader node in the REG.

Right after, comes the RSlots, they are always arranged in an ascending numeric sequence ranging from one up to the number of nodes connected at that time. In this way, each node connected to the group (having, therefore, an ID) can request the leader, in their respective RSlot corresponding to its ID, to have an available time in a future moment so it may send its message to another node in this group.

Finishing the SyncM, it comes the broadcast period. The leading node, upon receiving requests from every other node in the group, performs a shuffling in the order in which the nodes will transmit, adds some extra information like possible exchanges of ID of the nodes or commands like the end of the group, etc. This information does not exceed more than 10 bytes, for example.

Following each of these moments inside the SyncM has a clearance space of time to avoid the problems mentioned before with clock drift.

Each node has a simple path of working based on what happens around it. That can be expressed as a flowchart, presented in Figure 3, which represents each event and situation that a node may operate.

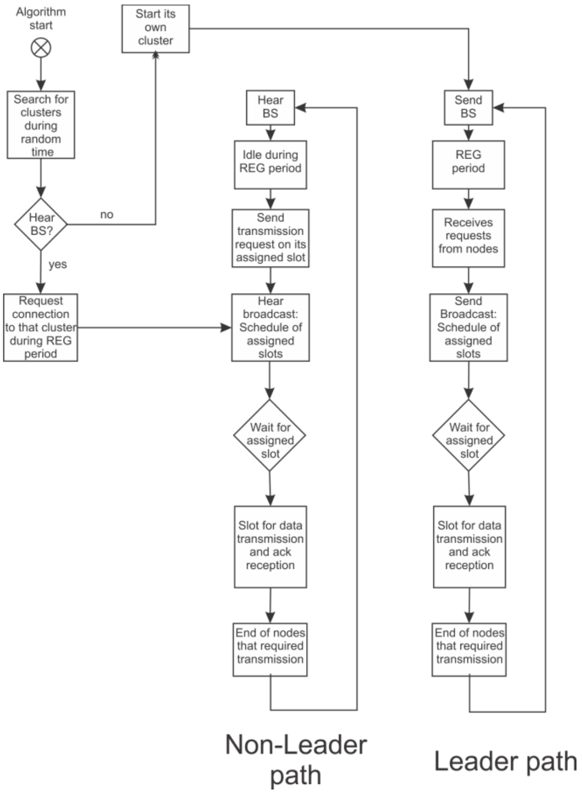

Figure 3. Node point of view of the DyTEE protocol.

4.2 *Sending useful data packets between nodes*

Given that each member of the group receives during the SyncM the total number of connected nodes at that time, it is possible for each node to determine the end of the broadcast transmission and ascertain if there will be enough time for communication between the first pair of nodes determined by the leader, as well as whether there will be enough time for the next transmission sequences until the end of the active period. At the end of the active period, all the members go to sleep and wait for the next cycle, reducing energy consumption, since the energy drained from the battery during sleep time is much smaller than the energy consumed in active period. When this occurs, the nodes follow the schedule again. This happens until the end of the entire sequence has passed. After that a new SyncM starts another period.

Each packet has a simple structure of a small header, message and checksum. After its transmission an ACK is expected from the receiver.

5 Topology and Parameters

To simulate the protocols on an automation environment, it was utilized 14 nodes and one coordinator in a star-like topology shown in Figure 4.

In this topology, all nodes see each other, i.e, each node is within transmission range of one another, and all nodes receive with equal probability new messages at a Poisson arrival rate. This arrival is given by a mean of what would be if 14 nodes receive, each, one message per minute, that is, 60 seconds divided by 14, which leads to approximately 1 message per 4285 seconds, or an arrival rate of 0,233 messages per second. During the cycles of the protocol, each node tries to forward his own message to the destination which acts only as a receiver.

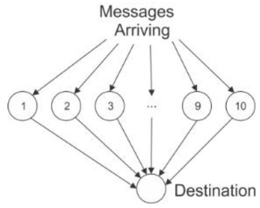

Figure 4. Star-like topology.

To verify the efficiency of DyTEE compared to S-MAC, it was considered the arrival of 2000 messages, each one having a length of 20 bytes with active time of each protocol kept equal to 25 milliseconds and their duty cycles equally varied in a range of 0,04 percent to 2,4 percent that is approximately the total periods of 60 seconds to 1 second, respectively. So it is possible to observe their best operation point with the parameters specified in Table I.

Table I. Parameters used to simulate the protocols.

Parameter	Value	Unit
Headers	5	Byte
Control packs	10	Byte
Payload	20	Byte
Channel Bandwidth	250	Kbps
Average arrival rate	0.233	Msg/Sec
Maximum active time	25	ms
Number of nodes (not counting destination)	14	node
Power consumption: Reception	75.9	mW
Transmission	165	
Sleep	0.015	

To obtain the results, it was used the MATLAB© software (MathWorks, 2007) with scripts that simulates the events that occur in each protocol (S-MAC and DyTEE), counting energy costs and other variables used for the measurements.

6 Results

Figure 5 presents the average energy consumption used by one node per useful byte sent. The DyTEE protocol has a better performance in energy saving than the S-MAC, because for almost all the time, DyTEE has fewer nodes turned on, unlike the S-MAC in which all nodes of the group are turned on during active periods. Furthermore, even if the leader node has a higher consumption, which is given to the expense of the realization of synchronization, all simulations resulted consumption for the leader around 7.3% more than the other nodes.

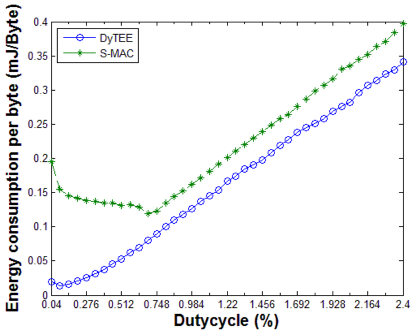

Figure 5. Average energy consumed, by one node per useful byte, per duty cycle percentage, to send 2000 packages of 20 bytes each.

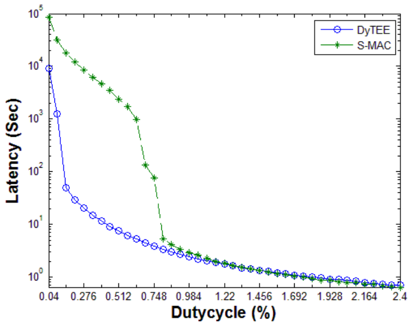

Figure 6. Average latency of packets per duty cycle percentage.

Figure 6 shows the result of the latency given by the difference between the time that the package appears in the node and the time at which it arrives at the destination. It is very important to notice that as the simulation has finite messages to deliver, the average latency will always have a finite time, as shown in Figure 6. Although, in a continuous operating system, what would mean infinite simulation time, this high latency time points, above 1000 seconds, refers to situations of message accumulation on nodes, causing them to probably have message losses due to memory limitation.

Figure 7 presents the excess of packet header (overhead) used in each protocol. Therefore, the difference between the values observed in these figures is consequence of the fact that while in the S-MAC there is an active period (with all nodes turned on) to send a single message, in DyTEE occurs one active period to synchronize all nodes and for the others active periods only the assigned pairs of communicating nodes remain active. This generates efficiency gains in proportion to the number of nodes that wish to transmit. The smaller is the duty cycle percentage, the greater is the gain. In the worst case, where no node wants to transmit (more common on higher duty cycles), the DyTEE shows less consumption than the S-MAC, because fewer nodes will be connected throughout the active period. This gain is also reflected in the throughput, shown in Figure 8.

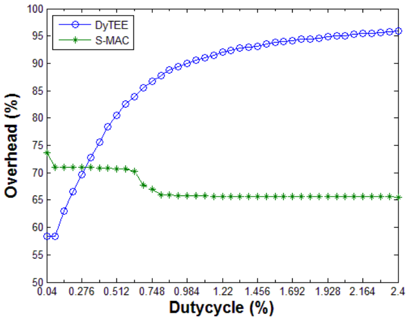

Figure 7. Overhead per duty cycle percentage, to send 2000 packets of 20 bytes each.

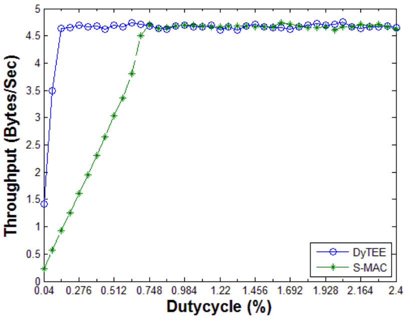

Figure 8. Throughput of the protocols per duty cycle percentage, to send 2000 packages of 20 bytes each.

Given all these results, it is expected that DyTEE has better average energy consumption per time in comparison with the S-MAC, which is presented in Figure 9 as an estimated average power to operate each protocol under the assumed parameters of Table I.

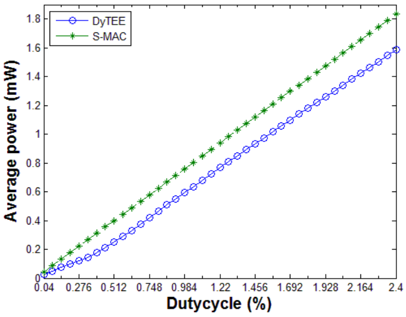

Figure 9. Average power per duty cycle percentage to operate during the simulation.

However, energy saving is the main goal for the DyTEE protocol rather than latency and throughput. Accordingly, Figure 10 present the relation between these three metrics combined, by a new metric of throughput per energy consumption times latency.

As the designed automation environment creates a mean of 14 messages per minute, it is expected that DyTEE handle this system when its period of operation becomes at least three times faster, that is, the duty cycle is equal or greater than 0,125 percent, which means a total period of 20 seconds or less. For S-MAC, as it delivers only one message per activation period, it needs, on average, an operation duty cycle of at least the same rate as the message arrivals, that is, a total period of 4,2 seconds or less, or duty cycles of at least 0,58 percent.

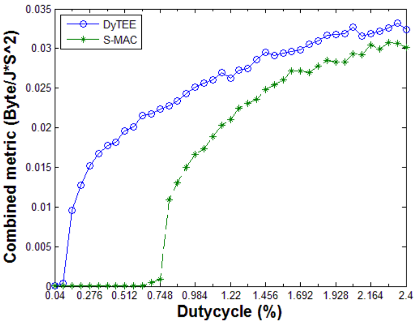

Figure 10. Throughput per energy consumption times latency per duty cycle percentage.

By the estimated values of duty cycles (greater than 0,125 percent for DyTEE and 0,58 percent for S-MAC, which are reasonable values according to Figures 6, 8 and 10) that would make each protocol to attend to the arrival rate of the system, it is possible to determine how long a node could work and how many messages it could deliver based on Figures 8 and 9.

Assuming that the node spends energy with parameters in Table I and is powered by a battery of 3,3 volts, 1200 milli-ampere hour (mAh), by the Figure 10 and points of 0,158 percent of duty cycle for DyTEE and 0,748 percent of duty cycle for S-MAC (known to be able to attend the system and proved by simulations), it result in an average power of 0,0762 milli-Watts (DyTTE) and 0,5798 milli-Watts (S-MAC).

With those values, it is estimated that DyTEE would work for about 5,93 years, while S-MAC would work for about only 284,59 days. Besides that, with only one battery, DyTEE would send about 827,93 Megabytes, while S-MAC would only send about 110,38 Megabytes.

Considering an automation environment that measures humidity and temperature with a sensor SHT71 (SENSIRION, 2011), each sensor would, in average, spend 90 microwatts more as specified in datasheet of SHT71. In this situation DyTEE would work for about 2,72 years, while S-MAC would work for about only 246,35 days and DyTEE would send about 379,72 Megabytes, while S-MAC would only send about 95,55 Megabytes.

7 Conclusions

This paper evaluates a new protocol, named Dynamic Timed Energy Efficient (DyTEE), with data collision free feature, presenting low energy consumption compared to the S-MAC protocol in a simulated automation environment. Moreover, the DyTEE also outperformed other metrics, like latency, power and throughput.

For future work, new researches are already being carried out, which will be tested on a real platform like sensor motes, in order to assess its performance in practical automation systems.

References

D. Falconer, F. Adachi, B. Gudmundson (1995). Time Division Multiple Access Methods for Wireless Personal Communications. IEEE Communications Maganize, vol. 33, no. 1, pp. 50-57.

W. Ye, J. Heidemann, D. Estrin, June 2002. An energy-efficient MAC protocol for wireless sensor networks. Proc. IEEE INFOCOM, pp. 1567-1576.

P.A. Ferreira Jr., A. Bauchspiess. November 2009. Energy Saving in Building Automation using ZigBee Wireless Network and Fuzzy Control. 20th International Congress of Mechanical Engineering.

802.11, June 1997. I. Wireless LAN MAC and Physical Layer Specifications. Editors of IEEE.

W. Ye, J. Heidemann, D. Estrin, June 2004. Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks. IEEE/ACM Transactions on Networking, v. 12, pp. 493-506.

T. van Dam and K. Langendoen, November 2003. An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. ACM Intl. Conference on Embedded Networked Sensor Systems (SenSys), pp. 171-180.

Z. Chen and A. Khokhar, October 2004. Self Organization and Energy Efficient TDMA MAC Protocol by Wake Up For Wireless Sensor Networks. IEEE SECON, pp. 335-341.

P. Huang, L. Xiao, S. Soltani, M. W. Mutka, N. Xi. First Quarter 2013. The Evolution of MAC Protocols in Wireless Sensor Networks: A Survey. IEEE Communications Surveys & Tutorials, vol. 15, no. 1, pp. 101-120.

P. Suriyachai, U. Roedig, A. Scott. Second Quarter 2012. A Survey of MAC Protocols for Mission-Critical Applications in Wireless Sensor Networks. IEEE Communications Surveys & Tutorials, vol. 14, no. 2, pp. 240-264.

MathWorks. (2007) MATLAB Version R2007b.

SENSIRION. (2011) Datasheet SHT7x (SHT71, SHT75) Humidity and Temperature Sensor IC, Version 5.