C-MAC: An Energy-Efficient MAC Scheme Using Chinese-Remainder-Theorem for Wireless Sensor Networks

YUH-SHYAN CHEN[1] AND YUN-WEI LIN[2]

[1]*Department of Computer Science and Information Engineering*
National Taipei University
Taipei, Taiwan, R.O.C.
E-mail: yschen@csie.ntpu.edu.tw

[2]*Department of Computer Science and Information Engineering*
National Chung Cheng University
Chia-Yi, Taiwan, R.O.C.
E-mail: jyneda@giam.dynu.com

This paper investigates a new TDMA-based MAC, called C-MAC (MAC using Chinese remainder theorem), protocol with low power consumption while maintaining low packet latency for large-scale cluster-based WSNs. To offer low power consumption, each node and coordinator to be active and sleep states based on the time-slot schedule of Chinese remainder theorem. To provide low packet latency, our C-MAC protocol provides an adaptive time-slot scheme to distributively and dynamically wake up time slots for the burst data transmission in a duty cycle. Finally, simulation results illustrate our performance achievements to verify that C-MAC performs better than existing TDMA-based MAC protocols, BMA and EMAC, in terms of power consumption and average packet latency.

Keywords: Wireless sensor networks, MAC, Latency, Energy efficient operation, energy-efficient scheduling

1. INTRODUCTION

Energy efficiency is the most important design issue for WSNs. To reduce energy consumption of the sensor nodes, many researches investigate the research issues of energy efficiency to design low-power communication protocols for WSNs. There are many researches about media access control (MAC) protocol designed for wireless sensor network. Existing results can be divided into contention-based MAC protocols [3] [8]and TDMA-based MAC protocols [7] [10]. High cost of idle listening and hidden terminal are the main design problems for the contention-based MAC protocols. This is mainly due to the idle listening [11]. In addition, TDMA-based MAC protocols is another main design stream for WSNs and is often used in cluster-based approach to reduce the cost of idle listening.

We investigate the TDMA-based MAC protocols [7] [10] as follows. Clustering is a common distributed techniques used in large-scale WSNs. In general, energy consumption in TDMA-based MAC protocols is more efficient than contention-based MAC protocols. Li *et al.* proposed BMA (Bit-Map-Assisted) MAC protocol [7] for a cluster-based WSNs. If any data arrives after the contention period, BMA protocol must waits for data transmissions in the next duty cycle. Therefore, it results a long packet latency. Recently, Hoesel *et al*. developed EMAC (EYES MAC) protocol [10]. In a duty cycle, only one cluster can work and all other clusters enter idle mode. EMAC incurs long packet latency than that of BMA protocol. Efforts will be made to develop a new TDMA-based MAC protocol with

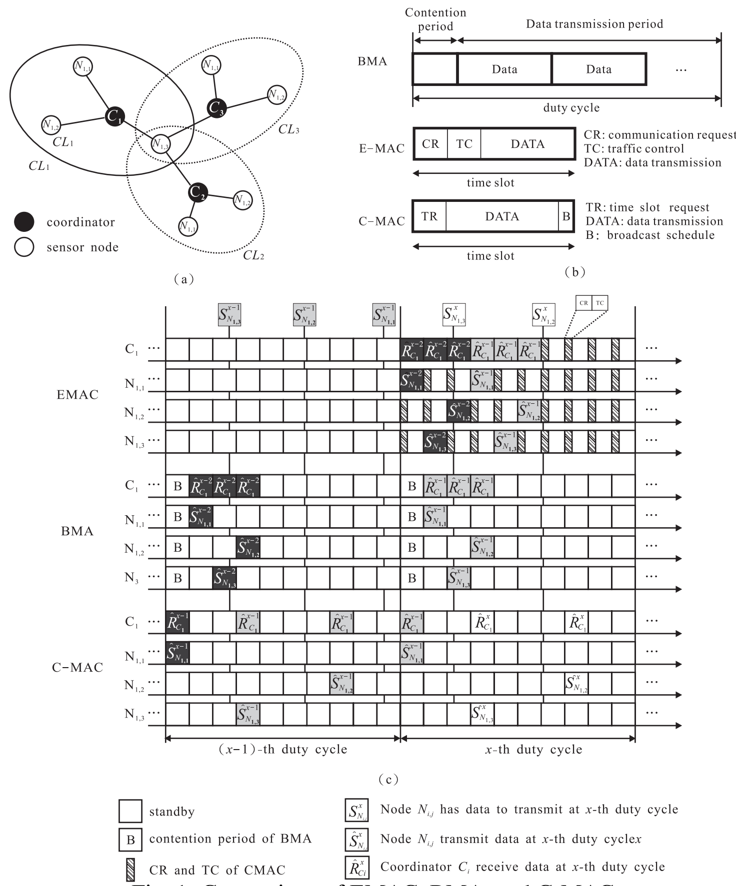

Fig. 1. Comparison of EMAC, BMA, and C-MAC

less power consumption and low packet latency. In this paper, we propose a new TDMA-based medium access control (MAC) protocol, called C-MAC, using Chinese remainder theorem for wireless sensor networks (WSNs). To achieve low power consumption, each node and coordinator obey the time-slot schedule of C-MAC to keep active and sleep states. Time-slots are based on the Chinese remainder theorem to avoid the time slot collision between different clusters. To provide low packet latency, our C-MAC protocol provides an adaptive time-slot scheme to wake up distributively and dynamically for the burst data transmission in a duty cycle. Finally, simulation results illustrate our performance achievements to verify that C-MAC performs better than existing TDMA-based MAC protocols, BMA and EMAC, in terms of power consumption and average packet latency. This paper is organized as follows. Section 2 discusses the basic idea of C-MAC protocol. Section 3 presents C-MAC protocol. Section 4 discusses the performance analysis. Section 5 gives a conclusion.

2. PRELIMINARY AND BASIC IDEA

Our work investigates the TDMA-based MAC protocol in the cluster-based WSNs. Example of a cluster-based WSN is IEEE 802.15.4 standard [5], which is a low-rate wireless personal area network (LR-WPAN). The star network defined in IEEE 802.15.4 standard [5] is also as an example of cluster-based WSNs. Let CL_i denote i-th clutser in the WSNs. Let C_i denote the coordinator node of CL_i. Coordinator node C_i discovers two hop away adjacency coordinator node to create the neighbor cluster table. Let $B_{CL_i} = \{CL_{i+1}, ..., CL_{i+j}, ..., CL_{i+k}\}$ denote the neighbor cluster table of CL_i. Let α_i is the number of neighbor clusters in B_{CL_i}. Example of a cluster-based WSN is given in Fig. 1(a), three cluster CL_1, CL_2, and CL_3 is existed. B_{CL_1}, B_{CL_2}, and B_{CL_3} are $B_{CL_1} = \{CL_2, CL_3\}$, $B_{CL_2} = \{CL_1, CL_3\}$, and $B_{CL_3} = \{CL_1, CL_2\}$ respectively, and $\alpha_1 = \alpha_2 = \alpha_3 = 2$. Our work mainly compares with two other existing TDMA-based MAC protocols, BMA [7] and EMAC protocols [10]. The model used in this work is same as the models defined in [2] [4] [7] [9] [10]. A cluster CL_i is composed of FFD (cluster coordinator) and RFD (cluster member) [5]. In the following, we point out the performance improvements of our scheme both in power consumption and packet latency.

Some notations are defined for the up-link communication. Let N_i denote the total number of sensor node in cluster CL_i. Let $S^x_{N_{i,j}}$ be an event request that sensor node $N_{i,j}$ intends to send data to coordinator node C_i occurred at x-th duty cycle in cluster CL_i, where $1 \leq j \leq N_i$. Let $\hat{S}^x_{N_{i,j}}$ denote a time slot, where sensor node $N_{i,j}$ sends data to coordinator for an event request $S^x_{N_{i,j}}$. Let $\hat{R}^x_{N_{i,j}}$ denote a time slot, where coordinator C_i receives data from sensor node $N_{i,j}$ for request $S^x_{N_{i,j}}$. The key idea of C-MAC protocol allows that coordinator node C_i can enter into the PS mode, a novel wake-up scheme is scheduled is this work for purpose of the low power consumption and low packet latency.

Before describing our scheme, we describe the main works of BMA and EMAC protocols [7] [10] as follows. Example is shown in Fig. 1(a) and $N_{1,3} = N_{2,3} = N_{3,3}$. The time slot structures of BMA and EMAC are given in Fig. 1(b). In EMAC protocol, radios of coordinator and sensor nodes are turned on in CR (communication request) period and in TC (traffic control) period to transmit/receive control messages. To prevent collision, only one cluster CL_i can active in a duty cycle. For an active duty cycle, all sensor nodes periodically wake up to listen control message from coordinator during TC period for every time slot. Not all sensor nodes have data to send or receive. Idle listening is occurred and useless control packet is increased. Example is given in Fig. 1(c), using EMAC protocol, nodes $N_{1,1}$, $N_{1,2}$, and $N_{1,3}$ send data to coordinator C_1 on time slots $\hat{S}^{x-2}_{N_{1,1}}$, $\hat{S}^{x-2}_{N_{1,2}}$, and $\hat{S}^{x-2}_{N_{1,3}}$ at x-th duty cycle for requests $S^{x-2}_{N_{1,1}}$, $S^{x-2}_{N_{1,2}}$, and $S^{x-2}_{N_{1,3}}$. The operation of BMA includes *cluster set-up* and *steady-state* phases, where steady-state phase contains contention, data transmission, and idle periods. If a sensor node intends to send data, it sends a short control to coordinator in the contention period. Coordinator broadcasts the schedule for sensor nodes after contention period. If any data arrives after the contention period, a sensor nodes must waits and transmits data until next duty cycle. That is, if any node detects request of data transmission, it buffers the data and transmits it until next duty cycle. Therefore, BMA provides less packet latency than EMAC. Example is given in Fig. 1(c), using BMA protocol, nodes $N_{1,1}$, $N_{1,2}$, and $N_{1,3}$ send data to coordinator C_1 at time slots $\hat{S}^{x-2}_{N_{1,1}}$, $\hat{S}^{x-2}_{N_{1,2}}$, and $\hat{S}^{x-2}_{N_{1,3}}$ at $(x-1)$-th duty cycle.

Our C-MAC protocol is developed to allow coordinator to some pre-scheduled time

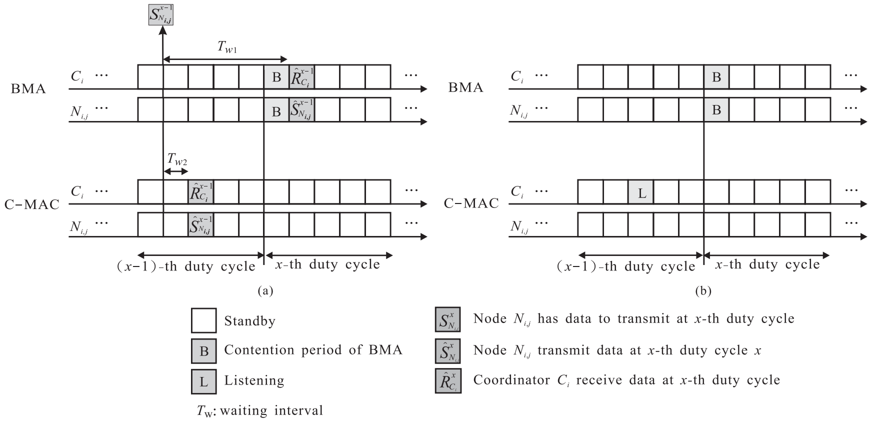

Fig. 2. The improvement of the packet latency of C-MAC protocol

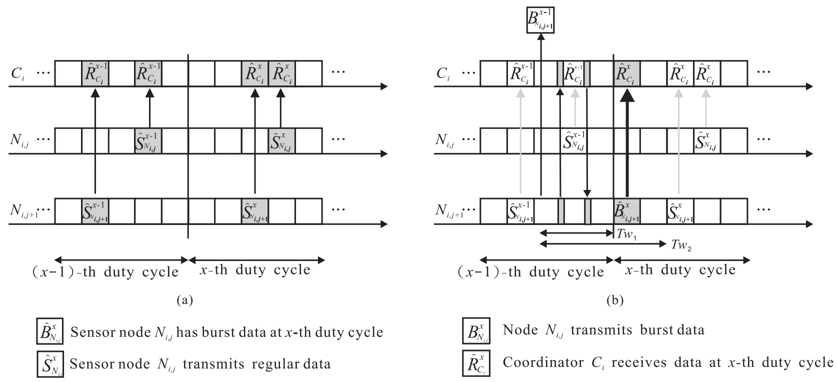

Fig. 3. (a) Regular data transmission and (b) Burst data transmission

slots by Chinese Remainder Theorem are in wake-up mode to reduce the power consumption and improve the packet latency. Example is given in Fig. 1(c), C-MAC allows that nodes $N_{1,2}$ and $N_{1,3}$ send data to coordinator C_1 at time slots $\hat{S}^{x-1}_{N_{1,2}}$ and $\hat{S}^{x-1}_{N_{1,3}}$ at $(x-1)$-th duty cycle for requests $S^{x-1}_{N_{1,2}}$ and $S^{x-1}_{N_{1,3}}$. Fig. 2 indicates the main feature how C-MAC can improve the packet latency then BMA protocol. Using the polling sequence produced by Chinese Remainder Theorem significantly reduces the collision probability for the up-link transmission in the fully distributed wireless environment.

3. C-MAC PROTOCOL

To effectively reduce the power consumption and packet latency, we present a new medium access control protocol using Chinese Remainder Theorem (CRT), called C-MAC protocol, for wireless sensor networks.

In C-MAC protocol, each sensor node $N_{i,j}$ is responsible for periodically wake up to transmit sensed data to the coordinator node C_i in a cluster CL_i, where $1 \leq j \leq N_i$. Let $WTP_{i,j}$ denote the wake-up time period of sensor node $N_{i,j}$, where $WTP_{i,j}$ =

$\hat{S}^{x}_{N_{i,j}} - \hat{S}^{x-1}_{N_{i,j}}$. Each sensor node $N_{i,j}$ has different wake-up time period $WTP_{i,j}$ to avoid the collision of data transmission. C-MAC protocol performs a wake-up time period schedule to generate distinct wake-up time period $WTP_{i,j}$ for each sensor node $N_{i,j}$ in a cluster CL_i. This wake-up time period schedule is executed by a coordinator node C_i based on a *prime* and *remainder sequences* (defined later) from the Chinese Remainder Theorem [1]. Each coordinator node C_i has distinct *prime* and *remainder sequences,* each sensor node $N_{i,j}$ can wake up in different time slot $\hat{S}^{x}_{N_{i,j}}$. Sensor node $N_{i,j}$ can choose another remainder sequences to change the $WTP_{i,j}$ for different sensing task.This result can effectively avoid the time slot collision for data transmission. The main goal of the wake-up time period schedule is to generate $WTP_{i,j}$ for each sensor node $N_{i,j}$. Sensor node $N_{i,j}$ wake up and transmit data to the coordinator node C_i if the sensor node $N_{i,j}$ has data transmission event before the wake-up time slot $\hat{S}^{x}_{N_{i,j}}$.

C-MAC provides two transmission capability, regular data transmission and burst data transmission mechanism. The regular data transmission mechanism is used for periodically transmitting sensed data. In the regular data transmission, sensor node $N_{i,j}$ transmits sensed data at time-slot $\hat{S}^{x}_{N_{i,j}}$ which is determined by *prime* and *remainder sequences*. As shown in Fig. 3(a), sensor node $N_{i,j}$ transmits sensed data to coordinator node C_i at a specific time-slot $\hat{S}^{x}_{N_{i,j}}$. The burst data transmission mechanism is used for transmitting the burst data. In the burst data transmission, sensor node $N_{i,j}$ transmits burst data at a specific time-slot which is determined by coordinator node C_i. Let $B^{x}_{N_{i,j}}$ be an burst data request that sensor node $N_{i,j}$ occurred at x-th duty cycle in cluster CL_i. Let $\hat{B}^{x}_{N_{i,j}}$ denote a time slot, where sensor node $N_{i,j}$ sends data to coordinator for a burst request $B^{x}_{N_{i,j}}$. As shown in Fig. 3(b), a burst data request is happened at $B^{x-1}_{N_{i,j+1}}$, and the burst data is transmitted at $\hat{B}^{x}_{N_{i,j+1}}$. Using burst data transmission mechanism, the time delay of this burst data transmission is T_1, and using regular data transmission to transmit the burst data, the time delay of burst data transmission is T_2, where $T_1 < T_2$, as shown in Fig. 3(b). It is worth to develop the burst data transmission mechanism for the burst data transmission. The regular data transmission and burst data transmission mechanisms of C-MAC are described as follows.

3.1 Regular Data Transmission Mechanism

In the regular data transmission, C-MAC protocol mainly calculates a duty cycle length L_i for each coordinator node C_i in the cluster CL_i, where L_i is the duty cycle length for cluster CL_i. Based on calculated L_i, the regular data transmission mechanism is executed as follows.

Sensor node $N_{i,j}$ has different wake-up time period $WTP_{i,j}$ for different sensing task. Due to different sensing task, each sensor nodes $N_{i,j}$ waits for different wake-up time period $WTP_{i,j}$ and wake up at different wake-up time slot $\hat{S}^{x}_{N_{i,j}}$ to transmit data to coordinator node C_i if sensor nodes $N_{i,j}$ has buffered data. The wake-up time slot $\hat{S}^{x}_{N_{i,j}}$ is assigned by coordinator node C_i for existing TDMA-based protocols. Only sensor node $N_{i,j}$ knows its required wake-up time period $WTP_{i,j}$ due to different sensing task. To assign the wake-up time slot $\hat{S}^{x}_{N_{i,j}}$, the assignment of required wake-up time period $WTP_{i,j}$ is not achieved by the coordinator node C_i. Therefore, in our C-MAC protocol, each sensor node $N_{i,j}$ determines its own the wake-up time slot $\hat{S}^{x}_{N_{i,j}}$. The wake-up time slot $\hat{S}^{x}_{N_{i,j}}$ is determined by prime and remainder sequences from Chinese Remainder Theorem. When a sensor node $N_{i,j}$ registers to the coordinator node C_i, the coordinator node C_i informs the sensor node $N_{i,j}$ prime and remainder sequences. Sensor node $N_{i,j}$ then chooses a remainder sequences to determine the wake-up time slot $\hat{S}^x_{N_{i,j}}$. Before describing how to determine the $\hat{S}^x_{N_{i,j}}$ from prime and remainder sequences, the Chinese Remainder Theorem [1] is formally described below.

Theorem 1 *[1] Let prime sequence* $p = \{p_1, p_2, ..., p_t, ...p_v\}$ *denote a pairwise relatively prime integer sequence, where* $1 \leqq t \leqq v$. *Let a remainder sequence* $r = \{r_1, r_2, ..., r_t, ..., r_v\}$ *denote the remainder integer sequence form* p*, where* $0 \leqq r_t < p_t$. *Then*

- *There exists an integer* I *such* $I \equiv r_t (\bmod p_t)$*, and*
- *If* $I' \equiv r_t (\bmod p_t)$*, then* $I' \equiv I (\bmod p_1 p_2 ... p_v)$

For example, there exists an integer $I = 23$, and the prime sequence $p = \{3, 5, 7\}$ and the remainder sequence $r = \{2, 3, 2\}$, since $23 \equiv 2 (\bmod 3)$, $23 \equiv 3 (\bmod 5)$. If another integer $I' = 128$ exists, then $128 \equiv 2 (\bmod 3)$, $128 \equiv 3 (\bmod 5)$, $128 \equiv 2 (\bmod 7)$ and $128 \equiv 23 (\bmod p_1 p_2 p_3) = 128 \equiv 23 (\bmod 105)$.

Let $p_i = \{p_{i,1}, p_{i,2}, ..., p_{i,s}, ...p_{i,k}\}$ denote the prime sequence of CL_i, where $p_{i,s} \in p$, $1 \leq s \leq k$. Let $r\_set_i = \{\{i-1, 0, ..., 0\}, ..., \{i-1, r_2, ..., r_s, ..., r_k\}, ..., \{i-1, p_{i,2} - 1, ..., p_{i.s} - 1, ...p_{i,k} - 1\}\}$ denote the available set of remainder sequence of CL_i, where $r_s \in r$, $0 \leq r_s < p_{i,s}$. Let $r_{i,j} = \{i-1, r_2, ..., r_s, ..., r_k\}$ denote the chose remainder sequence of sensor node $N_{i,j}$ from r_set_i, where $r_{i,j} \in r\_set_i$. For instance, $r\_set_1 = \{\{0,0\}, \{0,1\}, \{0,2\}, \{0,3\}, \{0,4\}\}$, $r\_set_2 = \{\{1,0\}, \{1,1\}, \{1,2\}, \{1,3\}, \{1,4\}\}$, and $r\_set_3 = \{\{2,0\}, \{2,1\}, \{2,2\}, \{2,3\}, \{2,4\}\}$ respectively if prime sequence $p_1 = \{3,5\}$, $p_2 = \{3,5\}$, $p_3 = \{3,5\}$.

The prime sequence $p_i = \{p_{i,1}, p_{i,2}, ..., p_{i,s}, ...p_{i,k}\}$ and remainder sequence $r_{i,j} = \{r_1, r_2, ..., r_s, ..., r_k\}$ are used to calculate the duty cycle length L_i and wake-up time slot $\hat{S}^x_{N_{i,j}}$, respectively. Let $B^j_{CL_i}$ denote the j-th neighbor cluster CL_j in B_{CL_i}. Let $N(B^j_{CL_i}) = N_j$ denote the number of sensor nodes in neighbor cluster CL_j.

Given a N_i for a cluster CL_i, coordinator node C_i can determine the duty cycle length L_i. Duty cycle is composed of many time-slots, and sensor node $N_{i,j}$ wakes up at wake-up time-slot $\hat{S}^x_{N_{i,j}}$. Given a sensor network with several cluster CL_i. A coordinator node C_i and sensor node $N_{i,j}$ in a cluster CL_i perform the regular data transmission mechanism as follows.

Step 1. For a cluster CL_i, each sensor node $N_{i,j}$ initially registers to the coordinator node C_i, such that coordinator node C_i knows the number of sensor nodes N_i in the cluster CL_i. Each sensor node $N_{i,j}$ registers to coordinator node C_i.

Step 2. Coordinator node C_i communicates with other neighbor coordinator nodes to obtain the total number of sensor nodes among one-hop neighbor clusters B_{CL_i}. Let β_i denote the number of sensor nodes in neighbor clusters B_{CL_i}. The β_i is computed by

$$\beta_i = \sum_{j=1}^{\alpha_i} N(B^j_{CL_i})$$

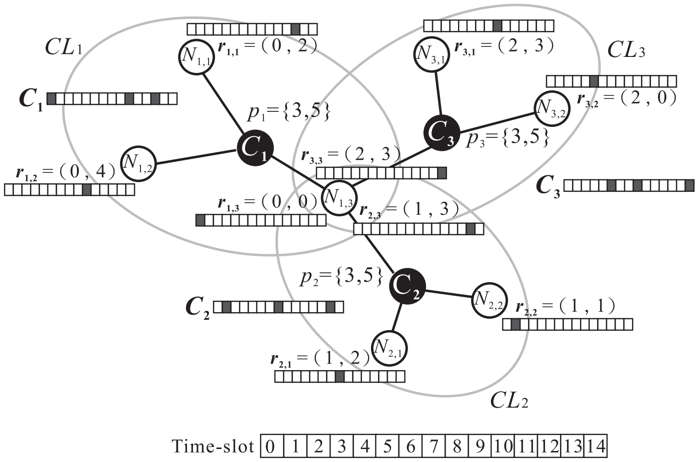

Fig. 4. Example of C-MAC

Step 3. The coordinator node C_i generates a prime sequence $p_i = \{p_{i,1}, p_{i,2}, ..., p_{i,s}, ...p_{i,k}\}$ to calculate the duty cycle length L_i, where $p_{i,1}$ is a smallest prime integer to satisfy $p_{i,1} > \alpha_i$. The duty cycle length L_i is calculated by

$$L_i = \min(\prod_{s=1}^{k} p_{i,s}), \text{ where } L_i \geq (\beta_i + N_i).$$

The coordinator node C_i announces the duty cycle length L_i, the prime sequence $p_i = \{p_{i,1}, p_{i,2}, ..., p_{i,s}, ...p_{i,k}\}$, and the available set of remainder sequence r_set_i, to all sensor nodes $N_{i,j}$, where $1 \leq j \leq N_i$. Each sensor node $N_{i,j}$ receives the duty cycle length L_i, the prime sequence $p_i = \{p_{i,1}, p_{i,2}, ..., p_{i,s}, ...p_{i,k}\}$, and the available set of remainder sequence r_set_i from coordinator node C_i.

Step 4. A sensor node $N_{i,j}$ chooses a remainder sequence $r_{i,j} = \{i-1, r_2, ..., r_s, ..., r_k\}$ from r_set_i and replies the selected remainder sequence $r_{i,j}$ to the coordinator node C_i. The coordinator node C_i removes $r_{i,j}$ from r_set_i. Next, coordinator node C_i announces the new r_set_i and reveices the remainder sequence $r_{i,k} = \{i - 1, r_2, ..., r_s, ..., r_k\}$ from the next sensor node $N_{i,k}$ untill coordinator node C_i reveices remainder sequences from all sensor nodes in CL_i.

Step 5. Each sensor node $N_{i,j}$ determines the wake-up time-slot $\hat{S}^x_{N_{i,j}}$, where $\hat{S}^x_{N_{i,j}} = (\sum_{s=1}^{k} t_{i,s} P_{i,s} r^{s-\text{th}}_{i,j}) \bmod L_i$, and $r^{s-\text{th}}_{i,j}$ is the s-th prime number in the $r_{i,j}$. Denote $P_i = \{P_{i,1}, P_{i,2}, ..., P_{i,s}, ...P_{i,k}\}$ as a *combine number sequence* of CL_i , and $P_{i,s} = \prod_{u=1,u\neq s}^{k} p_{i,u}$, $1 \leq s \leq k$. Denote $t_i = \{t_{i,1}, t_{i,2}, ..., t_{i,s}, ..., t_{i,k}\}$ as a *base number sequence* of CL_i, and $t_{i,s}$ is a smallest positive integer to satisfy $t_{i,s} P_{i,s} \equiv 1 (\bmod\, p_{i,s})$. Each sensor node $N_{i,j}$ uses the selected remainder sequence

$r_{i,j}$ to determine the wake-up time-slot $\hat{S}^x_{N_{i,j}}$, where

$$\hat{S}^x_{N_{i,j}} = (\sum_{s=1}^{k} t_{i,s} P_{i,s} r_{i,j}^{s-\text{th}}) \bmod L_i.$$

The wake-up time period $WTP_{i,j}$ of sensor node $N_{i,j}$ is $\hat{S}^x_{N_{i,j}} - \hat{S}^{x-1}_{N_{i,j}}$. To save energy, sensor node $N_{i,j}$ keeps in sleep mode if the time-slot is not $\hat{S}^x_{N_{i,j}}$. Each sensor node $N_{i,j}$ obeys the transmission schedule to send regular data to coordinator node C_i.

Step 6. All wake-up time slots $\hat{R}^x_{C_i}$ of coordinator node C_i are determined by $\hat{S}^x_{N_{i,j}} \cup \hat{S}^x_{N_{i,j+1}} \cup ... \cup \hat{S}^x_{N_{i,N_i}}$.

A sensor network can cluster using MINPOW [6]. As shown in Fig. 4, B_{CL_1} is $\{CL_2, CL_3\}$ and α_1 is 2. The smallest prime integer greater than α_1 is 3. The number of sensor nodes in neighbor clusters B_{CL_1} is $\beta_1 = N(B^1_{CL_i}) + N(B^2_{CL_i}) = N_2 + N_3 = 6$. The minimum product of prime sequence to satisfy $\beta_1 + N_1 = 6 + 3 = 9$ is $3 \times 5 = 15$. Therefore, the prime sequence of coordinator node C_1 is $p_1 = \{3, 5\}$. By the same method, the prime sequence of C_2 and C_3 are $p_2 = \{3, 5\}$ and $p_3 = \{3, 5\}$ respectively. The available set of remainder sequence in CL_1, CL_2, and CL_3 are $r\_set_1 = \{\{0, 0\}, \{0, 1\}, \{0, 2\}, \{0, 3\}, \{0, 4\}\}$, $r\_set_2 = \{\{1, 0\}, \{1, 1\}, \{1, 2\}, \{1, 3\}, \{1, 4\}\}$, and $r\_set_3 = \{\{2, 0\}, \{2, 1\}, \{2, 2\}, \{2, 3\}, \{2, 4\}\}$ respectively. Sensor node $N_{1,1}$ gets a remainder sequence $r_{1,1} = \{0, 2\}$, and $\hat{S}^x_{N_{1,1}} = ((t_{1,1} \times P_{1,1} \times r_{1,1}^{1-\text{th}}) + (t_{1,2} \times P_{1,2} \times r_{1,1}^{2-\text{th}})) \bmod 15 = ((2 \times 5 \times 0) + (2 \times 3 \times 2)) \bmod 15 = 12$. Sensor node $N_{1,2}$ gets a remainder sequence $r_{1,2} = \{0, 4\}$ and $\hat{S}^x_{N_{1,2}} = 9$. Sensor node $N_{1,3}$ gets a remainder sequence $r_{1,3} = \{0, 0\}$ and $\hat{S}^x_{N_{1,3}} = 0$. To confirm each $\hat{S}^x_{N_{i,j}}$ is unique time slot for preventing collision, we explain the proof as follows.

Theorem 2 *Assume p_s and p_u are co-prime and let r_s and r_u are arbitrary, where p_s, $p_u \in p$, r_s, $r_u \in r$, and $1 \leq r_s \leq p_s$, $1 \leqq r_u \leqq p_u$. The pair of equations $\hat{S}^x_{N_{i,j}} = r_s (mod\, p_s)$ and $\hat{S}^x_{N_{i,j}} = r_u (mod\, p_u)$ have an unique solution for $\hat{S}^x_{N_{i,j}}$ $(mod\, p_s \cdot p_u)$.*

Proof. Assume that $\hat{S}^x_{N_{i,j}}$ and $\hat{S}^{x\prime}_{N_{i,j}}$ are two solutions. From the first equation it follows that

$$\hat{S}^x_{N_{i,j}} - \hat{S}^{x\prime}_{N_{i,j}} = K_1 p_s$$

That is the difference between the solutions must be a multiple of p_s. From the second equation it follows that

$$\hat{S}^x_{N_{i,j}} - \hat{S}^{x\prime}_{N_{i,j}} = K_2 p_u$$

From above equations, it must be that $\hat{S}^x_{N_{i,j}} - \hat{S}^{x\prime}_{N_{i,j}}$ is a multiple of both p_u and p_s. By assumption, p_s and p_u are co-prime that means gcd(p_s, p_u) = 1. The least common multiple of p_s and p_u is $p_s . p_u$, which means that $\hat{S}^x_{N_{i,j}} = \hat{S}^{x\prime}_{N_{i,j}} \cdot (mod\, p_s \cdot p_u)$. This proves uniqueness. ■

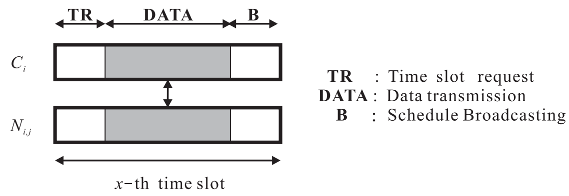

Fig. 5. Time slot structure of C-MAC

3.2 Burst Data Transmission Mechanism

The main objective in designing the burst data transmission mechanism is to improve the latency of transmitting the burst data. Let r_set_{used} denote a remainder sequence set which is those remainder sequences $r_{i,j}$ chose by sensor node $N_{i,j}$. Based on r_set_{used}, sensor node $N_{i,k}$ can inform a burst data request $B^x_{N_{i,k}}$ to coordinator node C_i via the designed time slot $\hat{R}^x_{N_{i,j}}$, where $N_{i,k} \neq N_{i,j}$. This structure of $\hat{R}^x_{N_{i,j}}$ consists of three sections, ***TR***, *Data* and *B section* as illustrated in Fig. 5. TR section is used for coordinator node C_i to receive the burst data request $B^x_{N_{i,k}}$ from sensor node $N_{i,k}$. DATA section is used for coordinator node C_i to receive data from sensor node $N_{i,j}$ in arranged schedule. B section is used for coordinator node C_i to announce temporary time slots. The burst data transmission mechanism operates as follows. Multiple sensor nodes $N_{i,k}, N_{i,k+1}, ..., N_{i,k+n}$ send burst data requests $B^x_{N_{i,k}}, B^x_{N_{i,k+1}}, ..., B^x_{N_{i,k+n}}$ to coordinator node C_i at TR section, where $0 \leq n < N_i$. When the coordinator node C_i detects a collision,the coordinator node C_i allocates two temporary time slots. Let $\hat{B}^{\text{1st}}_{C_i}$ and $\hat{B}^{\text{2nd}}_{C_i}$ denote the first and second allocated temporary time slots respectively by coordinator node C_i. At the temporary time slot $\hat{B}^{\text{1st}}_{C_i}$ and $\hat{B}^{\text{2nd}}_{C_i}$, each sensor node $N_{i,k}$ can send burst data after a random backoff time $R_{i,k}$ if the physical media is free.

Given a coordinator node C_i and sensor nodes $N_{i,k}, N_{i,k+1}, ..., N_{i,k+n}$ in a cluster CL_i, the burst data transmission mechanism is performed as follows.

Step 1: Sensor nodes $N_{i,k}, N_{i,k+1}, ..., N_{i,k+n}$ send burst data requests $B^x_{N_{i,k}}, B^x_{N_{i,k+1}}, ..., B^x_{N_{i,k+n}}$ to coordinator node C_i at TR section of time slot $\hat{R}^x_{N_{i,j}}$. Coordinator node C_i detects the collision condition. Coordinator node C_i judges that multiple sensor nodes $N_{i,k}, N_{i,k+1}, ..., N_{i,k+n}$ send burst data requests $B^x_{N_{i,k}}, B^x_{N_{i,k+1}}, ..., B^x_{N_{i,k+n}}$ due to the collision condition.

Step 2: Coordinator node C_i announces to allocate two temporary time slots $\hat{B}^{\text{1st}}_{C_i}$ and $\hat{B}^{\text{2nd}}_{C_i}$ for sensor nodes $N_{i,k}, N_{i,k+1}, ..., N_{i,k+n}$ at B section of time slot $R^x_{N_{i,j}}$. The two temporary time slots $\hat{B}^{\text{1st}}_{C_i}$ and $\hat{B}^{\text{2nd}}_{C_i}$ are

$$\hat{B}^{\text{1st}}_{C_i} = (\hat{R}^x_{N_{i,j}} + \min(q)) \bmod L_i \neq (\sum_{s=1}^{k} t_s P_s r\_set_{used}) \bmod L_i$$

$$\hat{B}^{\text{2nd}}_{C_i} = (\hat{B}^{BS}_{C_i} + \min(q)) \bmod L_i \neq (\sum_{s=1}^{k} t_s P_s r\_set_{used}) \bmod L_i$$

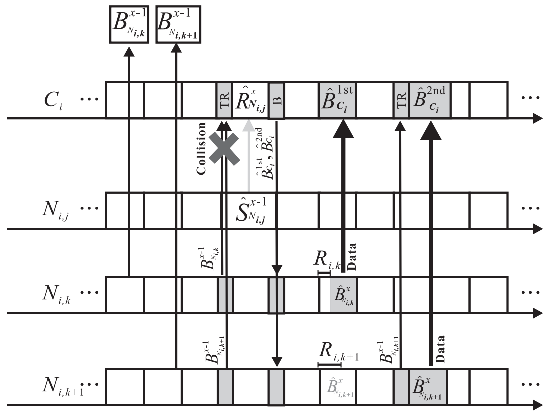

Fig. 6. Burst data transmission mechanism

The entire temporary time slot $\hat{B}_{C_i}^{1st}$ is used for coordinator node C_i to receive burst data from sensor node $N_{i,k}$. The temporary time slot $\hat{B}_{C_i}^{1st}$ does not contain TR and B section. At the temporary time slot $\hat{B}_{C_i}^{1st}$, sensor nodes $N_{i,k}, N_{i,k+1}, ..., N_{i,k+n}$ can send burst data after a random backoff time $R_{i,k}, R_{i,k+1}, ..., R_{i,k+n}$ respectively if the physical media is free.

Step 3: The temporary time slot $\hat{B}_{C_i}^{2nd}$ contains TR and B section. Sensor nodes $N_{i,k+1}, ...,$ $N_{i,k+n}$ send burst data requests $B_{N_{i,k+1}}^{x}, ..., B_{N_{i,k+n}}^{x}$ to coordinator node C_i at TR section of time slot $\hat{B}_{C_i}^{2nd}$. Sensor nodes $N_{i,k+1}, ..., N_{i,k+n}$ can send burst data after a random backoff time $R_{i,k+1}, ..., R_{i,k+n}$ respectively if the physical media is free. If coordinator node C_i detects collision condition again at TR section of time slot $\hat{B}_{C_i}^{2nd}$, goto step 2.

In step 1, if coordinator node C_i exactly receives a burst data requests $B_{N_{i,k}}^{x}$ and does not detect any collision, coordinator node C_i only allocates $\hat{B}_{C_i}^{1st}$ and sensor node $N_{i,k}$ sets the $R_{i,k} = 0$. Example is illustrated in Fig. 6, sensor nodes $N_{i,k}$ and $N_{i,k+1}$ use r_set_{used} to computer the next wake-up time slot $7\hat{R}_{N_{i,j}}^{x}$of coordinator node C_i. Sensor nodes $N_{i,k}$ and $N_{i,k+1}$ send burst data requests $B_{N_{i,k}}^{x}$ and $B_{N_{i,k+1}}^{x}$ respectively at TR section of time slot $\hat{R}_{N_{i,j}}^{x}$. Coordinator node C_i detects a collision at TR section of time slot $\hat{R}_{N_{i,j}}^{x}$. Coordinator node C_i announces two temporary time slots $\hat{B}_{C_i}^{1st}$ and $\hat{B}_{C_i}^{2nd}$ at B section of time slot $\hat{R}_{N_{i,j}}^{x}$. Sensor node $N_{i,k}$ sends burst data first at temporary time slot $\hat{B}_{C_i}^{1st}$ due to random backoff time $R_{i,k} < R_{i,k+1}$. Sensor node $N_{i,k+1}$ sends a burst data request $B_{N_{i,k+1}}^{x}$ to coordinator node C_i at TR section of temporary time slot $\hat{B}_{C_i}^{2nd}$. Coordinator node C_i exactly receives the burst data request $B_{N_{i,k+1}}^{x}$. Sensor node $N_{i,k+1}$ transmits burst data at DATA section of time slot $\hat{B}_{C_i}^{2nd}$.

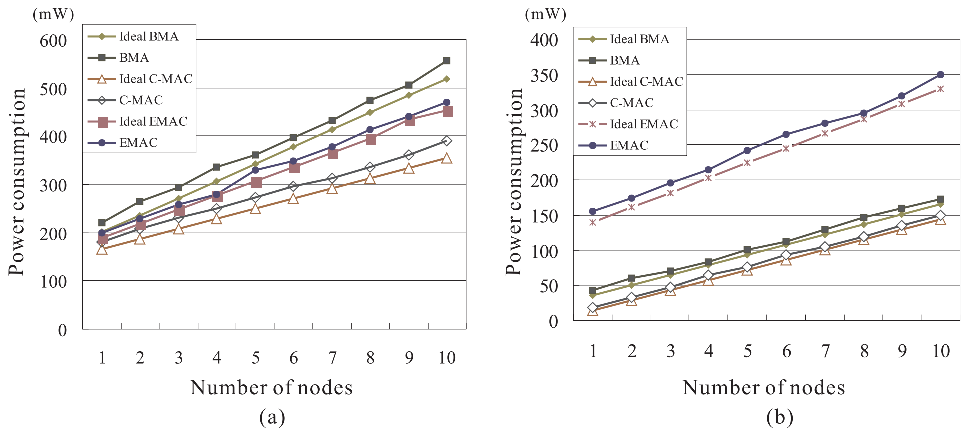

Fig. 7. Power consumption for (a) US_{data} and (b) US_{none}

4. PERFORMANCE ANALYSIS

To evaluate the performance of BMA [7], EMAC [10], and our C-MAC for power consumption and packet latency, we consider simulation scenarios under with upstream/no-upstream data transmission as follows. The system parameters are given below. To discuss the effect of the number of nodes in WSNs, the number of sensor nodes is assumed from 1 to 10 nodes. The simulation environment in this study is a cluster-based network, the coordinator is in the simulation area and all sensor nodes are randomly deployed. The topology of network is fixed and unchangeable. The other system parameters, refereed from [9] [10]. The radio propagation range is fixed at 10 meters. Adopting the same power consumption model [10], the power consumption of sensor node in transmitting, receiving, and standby states are 21mW, 14.4mW, and 15 μW, respectively. In the following, the simulated results in our simulation are represented as "Ideal EMAC", "Ideal BMA" and "Ideal C-MAC" for EMA, BMAC, and C-MAC protocols, respectively. When a sensor node is idle listing, the sensor node stay in standby mode. The performance metrics to be observed are:

- *Power consumption:* The power consumption of both coordinator nodes and sensor nodes are consumed in the WSNs. Coordinator nodes and sensor nodes consume variable energy in different radio operational mode.
- *Packet latency:* The time is spent between event generated and packet transmitted. The packet latency time includes upstream and downstream.

An efficient MAC protocol in a WSN is achieved with a low power consumption and packet latency. To illustrate the performance achievements, power consumption and packet latency for EMA, BMAC, and C-MAC protocols are compared as follows.

4.1 Power consumption

Fig. 7 shows the simulation results of upstream data transmission for EMA, BMAC, and C-MAC protocols. Fig. 7(a) gives the average simulation result of US_{data}. In BMA,

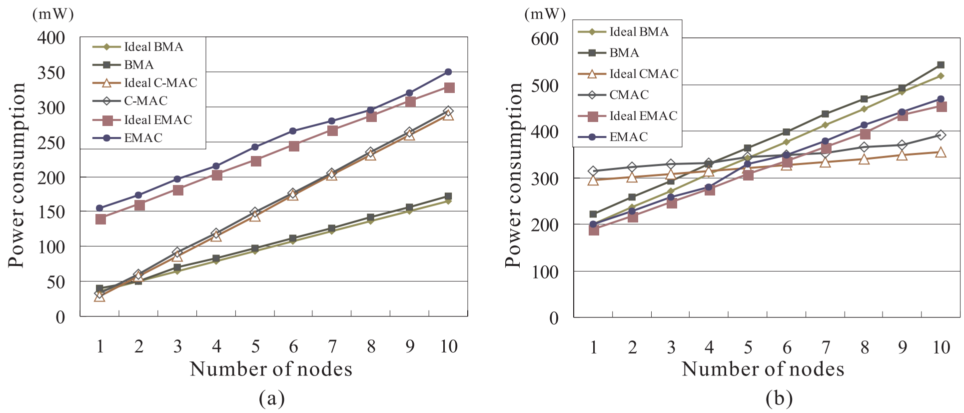

Fig. 8. Power consumption for (a) DS_{data} and (b) DS_{none}

coordinator node C_i receives the requests from each sensor node and broadcasts schedule in the contention period. In EMAC, sensor nodes send requests to contend the transmission time slot. After the contention, cluster node broadcasts the transmission schedule to each sensor node. Both BMA and EMAC consume more energy due to its heavy control overhead. In C-AMC, transmission schedule is chosen by sensor nodes when sensor nodes register to coordinator node C_i. Therefore, the minimum control overhead is used. Fig. 7(b) shows the average simulation result of US_{none}. EMAC consumes more power than BMA and C-MAC. In EMAC, even sensor node has no data transmission, sensor node still need to receive the control message from coordinator node C_i. The energy is wasted on this operation. In BMA, sensor nodes keep sleep in contention period because of no data transmission. In our C-MAC, only coordinator node C_i wake up in the assigned time slot, sensor nodes keep sleep for power saving purpose.

Fig. 8 illustrates the simulation results of downstream data transmission for EMA, BMAC, and C-MAC protocols. Fig. 8(a) is the average simulation result of DS_{data}. EMAC consumes the most energy due to using the greater number of control overhead. C-MAC consumes more power than BMA because of switch operational state. Fig. 8(b) is the average simulation result of DS_{none}. The result of C-MAC is almost horizontal because both coordinator node C_i and sensor nodes stay in receiving mode. In BMA, coordinator node C_i wakes up to transmit control message and all sensor nodes wake up to receive control message in the contention period. In EMAC protocol, coordinator node C_i wakes up to listen communication requests and transmits traffic control messages in TC section. All sensor nodes wake up to listen control messages in TC section. In C-MAC protocol, sensor nodes wake up to listen the downstream data from coordinator node C_i. If coordinator node C_i has no data to transmit, it still wakes up to prepare to receive data from sensor nodes. In our simulation result, C-MAC consumes less energy during upstream. But C-MAC consumes more energy during the downstream. That is caused by insufficient information for sensor nodes to know that coordinator node C_i has control messages to send or not.

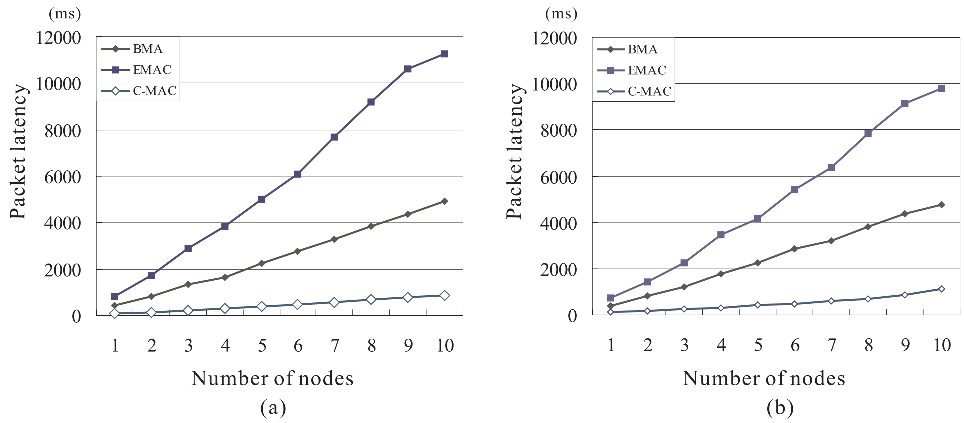

Fig. 9. Packet latecy for (a) upstream and (b) downstream

4.2 Packet latency

Fig. 9 shows the simulation results of packet latency for EMA, BMAC, and C-MAC protocols. Fig. 9(a) and (b) are the average simulation results of upstream and downstream data transmissions. To prevent the collision in EMAC protocol, adjacent clusters active by turns. When some events are occurred, data packets must be stored in the buffer until turn to active duty cycle. This causes the higher latency. In BMA protocol, if burst data detected after the contention period, sensor node buffered the data and transmit the data until the next duty cycle. The worst case of BMA is event occur after the contention period, the packet latency is almost a duty cycle. In our C-MAC MAC protocol, sensor node can request additional time slot when coordinator is wake up to serve other sensor nodes. The result is given in Fig. 9. As a summary, our C-MAC has the better packet latency, compared to all other protocols.

5. CONCLUSION

This paper investigates a new TDMA-based MAC protocol in cluster-based wireless WSNs with low power consumption and low packet latency. We propose an energy-efficient MAC scheme using Chinese Remainder Theorem. To provide low packet latency, our C-MAC protocol provides an adaptive time-slot scheme to distributively and dynamically wake up time slots for the burst data transmission in a duty cycle, where the wake-up time slots are based on the Chinese remainder theorem to avoid the time slot collision between different clusters. Finally, simulation result illustrates our performance achievements to verify that C-MAC performs better than existing TDMA-based MAC protocols, BMA and EMAC, in terms of power consumption and average packet latency.

6. ACKNOWLEDGMENTS

This research was supported by the National Science Council of the R.O.C. under grant NSC-95-2221-E-305-008. This paper is an extension of results presented at IEEE International Conference on Communications, (ICC 2007), Glasgow, Scotland, 2007. The authors would like to thank the anonymous referees for carefully reading an earlier version of this paper and giving many helpful suggestions.

REFERENCES

1. J.-H. Hong C.-H. Wu and C.-W. Wu. "RSA Cryptosystem Based on the Chinese Remainder Theorem,". In *Proceeding of Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 391–395, Jan. 2001.

2. T. Dam and K. Langendoen. "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks,". In *Proceeding of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 171 – 180, Nov. 2003.

3. A. El-Hoiydi, J. D. Decotignie, and J. Hernandez. "Low Power MAC Protocols for Infrastructure Wireless Sensor Networks". In *Proc. of European Wireless (EW'04)*, pages 563–569, Feb. 2004.

4. L.F.W. Hoesel and P.J.M. Havinga. "A Lightweight Medium Access Protocol(LMAC) for Wireless Sensor Networkd: Reducing Preamble Transmissions and Transceiver State Switches". In *Proceeding of International Workshop on Networked Sensing Systems (INSS)*, pages 1481–1486, Mar. 16-20 2004.

5. Jr. J. A. Gutierrez, E. H. Callaway and Jr. R. L. Barrett. *"Low-Rate Wireless Personal Area Networks"*. Standards Information Network, IEEE Press, 2003.

6. V. Kawadia and P. R. Kumar. "Power Control and Clustering in Ad Hoc Networks". In *Proceeding of INFOCOM 2003*, pages 459–469, April 2003.

7. J. Li and G. Y. Lazarou. "A Bit-Map-Assisted Energy-Efficient MAC Scheme for Wireless Sensor Networks". In *Proceeding of Information Processing in Sensor Networks (IPSN)*, pages 55–60, 26-27 Apr. 2004.

8. J. Polastre, J. Hill, and D. Culler. "Versatile Low Power Media Access for Wireless Sensor Networks". In *Proceeding of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 244–251, 3-5 Nov. 2004.

9. L.F.W. van Hoesel, S. Chatterjea, and P.J.M. Havinga. "An Energy Efficient Medium Access Protocol for Wireless Sensor Networks". In *Proceeding of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pages 244–251, 3-5 Nov. 2004.

10. L.F.W. van Hoesel, T. Nieberg, H.J. Kip, and P.J.M. Havinga. "Advantage of a TDMA based, energy-efficient, self-organizing MAC protocol for WSNs". In *Proceeding of IEEE Vehicular Technology Conference (VTC)*, pages 1598–1602, Spring 2004.

11. W. Ye, J. Heidemann, and D. Estrin. "Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks". *IEEE/ACM Transactions on networking*, (3):493–506, Jun. 2004.