Feasibility Study of Zero Excess Node MAC (ZEN-MAC) for Ultra-Low-Power Communications

Tatsuhiro Kawaguchi and Koji Ishibashi
Advanced Wireless and Communication Research Center, The University of Electro-Communications,
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan.
Emails: kawaguchi@awcc.uec.ac.jp, koji@ieee.org

***Abstract*—The zero excess node MAC (ZEN-MAC) protocol was recently proposed as an approach that leverages clear channel assessment (CCA) and multistage beacons to achieve substantially lower power consumption compared with conventional receiver-initiated methods. However, its implementation requires precise CCA control, and the feasibility of real-world deployment, along with the extent of achievable power savings, has not yet been clarified. In this paper, we implement ZEN-MAC on a commercially available TI CC1312R SimpleLink High-Performance Sub-1 GHz Wireless MCU and refine the algorithm to accommodate practical hardware constraints. Through direct power consumption measurements conducted in real communication scenarios, we demonstrate the energy efficiency of the improved ZEN-MAC protocol.**

***Index Terms*—Wireless sensor networks, clear channel assessment, receiver-initiated MAC, asynchronous, implementation, energy harvesting.**

I. INTRODUCTION

In recent years, increasing attention has been directed toward large-scale infrastructure monitoring systems that employ digital twin technology to analyze natural phenomena based on the long-term collection of spatial environmental data. As a means of acquiring such data, considerable efforts have focused on constructing sustainable and virtually perpetual energy harvesting wireless sensor networks (EHWSNs), which operate using small amounts of energy harvested from the environment [1]–[3]. However, because the harvested energy is limited, many studies have examined methods for optimizing energy usage in multihop communication to improve scalability [4], [5]. These methods activate the radio frequency (RF) modules only during transmission periods while keeping the nodes in sleep mode at all other times.

The intermittent receiver-driven data transmission (IRDT) protocol [6], which is an asynchronous receiver-initiated medium access control (MAC) protocol, enables communication by having sensor nodes (SNs) periodically wake from sleep and transmit ready to receive (RTR) messages. Data-holding SNs continuously perform idle listening to detect these RTR messages and then respond by transmitting their data packets (DATAs) to the RTR senders. However, IRDT still requires prolonged idle listening, and the frequent activation of the RF modules increases power consumption in SNs that hold data. Increasing the transmission frequency of RTR messages can shorten the idle listening duration, but doing so raises the power consumption of each SN. This trade-off motivates the need for a method that reduces idle listening without increasing the load on the transmitting nodes. To address this challenge, the zero excess node MAC (ZEN-MAC) protocol was proposed in [7]. It incorporates a mechanism that uses CCA and multistage beacons to mitigate the power consumption caused by idle listening.

Although the paper [7] demonstrated the potential of the proposed ZEN-MAC protocol for reducing power consumption through computer simulations, its feasibility has not yet been examined using actual hardware. As a result, the implementation challenges and practical effectiveness of ZEN-MAC remain unclear. In particular, the ZEN-MAC protocol relies on the capability to rapidly control the RF circuit through a low-cost microcontroller unit (MCU) in order to realize intermittent CCA and sleep operations.

In this paper, we implement the protocol in software on a commercially available evaluation board (TI CC1312R SimpleLink High-Performance Sub-1 GHz Wireless MCU [8]) and evaluate its feasibility and implementation challenges. During the implementation of ZEN-MAC, we observe that, for commercially available devices, the short CCA execution intervals that contribute to power savings depend on the response speed that can be controlled by the hardware. As a result, the CCA execution interval must be sufficiently long to allow the RF module to enter a sleep state. This paper examines whether significant power savings can still be achieved under this constraint. Moreover, to reliably realize the functionality of the ZEN-MAC protocol, enhancements to the beacon observation mechanism are required to account for delays in RF control. Specifically, this functionality is achieved by introducing a time-window mechanism in which CCA is performed over a fixed period and the number of detected beacons is counted. Finally, the reduction in power consumption achieved by the proposed method is demonstrated through a comparison with conventional approaches in a simple multihop network.

II. SYSTEM MODEL

We consider a wireless sensor network (WSN) composed of multiple SNs and a single gateway (GW). Each SN employs an off-the-shelf TI CC1312R SimpleLink High-Performance Sub-1 GHz Wireless MCU (LAUNCHXL-CC1312R1), as illustrated in Fig. 1. A regulated 3.3 V power supply is provided from a stabilized external source. The network is organized into 4 clusters, and the cluster that contains SN i corresponds

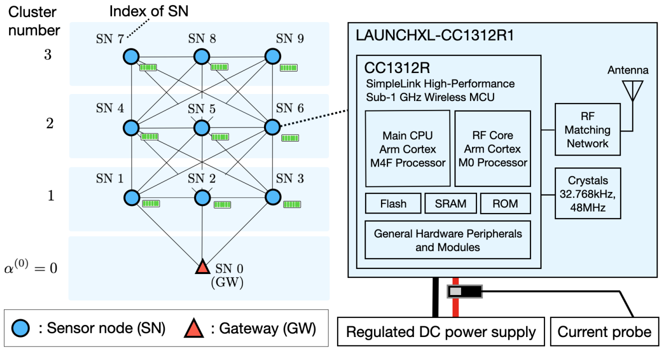

Fig. 1. System model. An example of a simple multihop network with an adequate power supply.

to the $\alpha^{(i)}$-th cluster, where $\alpha^{(i)} \in \{x \in \mathbb{Z} | 0 \leq x \leq 3\}$. In this study, we assume that the cluster number $\alpha^{(i)}$ is assigned according to the distance between the i-th SN and the GW and according to its feasible communication range at the time of installation. These cluster numbers are used solely to determine the direction of data transmission. Each SN operates according to the MAC protocol, which is implemented in software and described in detail in the following sections. The software implementation is based on the SIMPLELINK-LOWPOWER-F2-SDK (version 8.30.01.01). The proprietary RF software package provided by Texas Instruments is used with the following configuration: symbol rate of 100 kbps, frequency deviation of 50 kHz, and binary GFSK modulation. The receiver bandwidth is configured as 195.9 kHz. Data whitening is enabled using a dynamic IEEE 802.15.4g-compatible whitener with a 32-bit CRC. The preamble length is 4 bytes, the sync word is 24 bits, the transmission power is 0 dBm, and the carrier frequency is set to 923 MHz.

III. IMPLEMENTATION OF MAC PROTOCOLS

This section describes the challenges encountered in implementing the ZEN-MAC protocol and the improvements made to the algorithm to address hardware constraints in practical deployments. As a basis for evaluating the implementation, we first provide an overview of the IRDT protocol.

A. *Conventional IRDT Protocol*

Fig. 2 illustrates the basic operational procedure of the IRDT protocol. In this protocol, nodes alternate between sleep and wake states according to a predefined intermittent interval T_R s. The behavior of each node depends on whether it holds a DATA. If no DATA is stored, the SN operates as a receiver. At each interval, receiver nodes broadcast an RTR message. Meanwhile, a transmitter continues idle listening until it receives an RTR message, thereby identifying a node that is ready to initiate communication.

If an SN holds sensing DATA, it continuously receives and listens for RTR messages. Each RTR contains a cluster number $\alpha^{(i)}$, which is used to determine the direction of communication. When the i-th SN receives an RTR message with a smaller cluster number, a communication link is established, enabling asynchronous multihop communication without maintaining any routing table. Moreover, in the IRDT protocol, the transmitter sends an send-request (SREQ) message to the receiver instead of transmitting DATA immediately after receiving an RTR, as illustrated in Fig. 2. Because SREQ packets are shorter than DATA packets, this method reduces both power consumption and air traffic. It is particularly effective when multiple SNs holding DATA respond to the same RTR message. Upon receiving an SREQ, the receiver responds with an receive-acknowledgement (RACK). After transmitting DATA, the transmitter sends a data-acknowledgement (DACK). Once a node receives DATA from another SN, it returns to sleep to prepare for subsequent relay communication. A timeout is configured for the waiting period associated with each message, such as the reception of an RACK. If a required message is not received within the specified time, the corresponding DATA is discarded.

A key consideration in implementing the IRDT protocol is the need to configure a sufficiently long reception timeout to prevent the messages from being missed. Thus, because the basic operation of IRDT does not rely on rapid RF control, its implementation remains relatively straightforward. The IRDT protocol nevertheless incurs substantial power consumption during the idle listening required to detect potential communication partners. The ZEN-MAC protocol [7], introduced in the following section, is designed to address this limitation.

B. *ZEN-MAC Protocol and Its Implementation Challenges*

The ZEN-MAC protocol consists of two primary strategies: 1) an low power listening (LPL) method that employs multistage beacons, and 2) a load control function optimized for energy-harvesting (EH) power supply. In this paper, we focus on evaluating whether the functionality in 1) can be realized through RF control on actual hardware and whether it provides meaningful reductions in power consumption. We present an implementation-based investigation of these aspects. The ZEN-MAC protocol introduces two types of beacons, named initial beacon (IB) and cluster beacon (CB), prior to RTR transmission. The DATA receiver repeatedly performs CCA and transitions to continuous reception only when these beacons are detected, enabling it to observe the RTR. However, unnecessary RTR reception increases power consumption, and therefore the ZEN-MAC protocol determines whether the receiving cluster is the intended destination by varying the interval between the IB and the CB.

Several concerns arise when considering the implementation of the proposed method. Commercial MCUs suitable for mass production may not always incorporate high-end or high-performance CPUs, and therefore the achievable RF control speed is expected to be limited. Moreover, because the RF circuit repeatedly switches between active and sleep states, it is necessary to experimentally verify whether power consumption can be effectively reduced even when the transition periods are taken into account. Jitter in the timing of RF state transitions also raises concerns regarding the reliability of achieving the required high-precision operation.

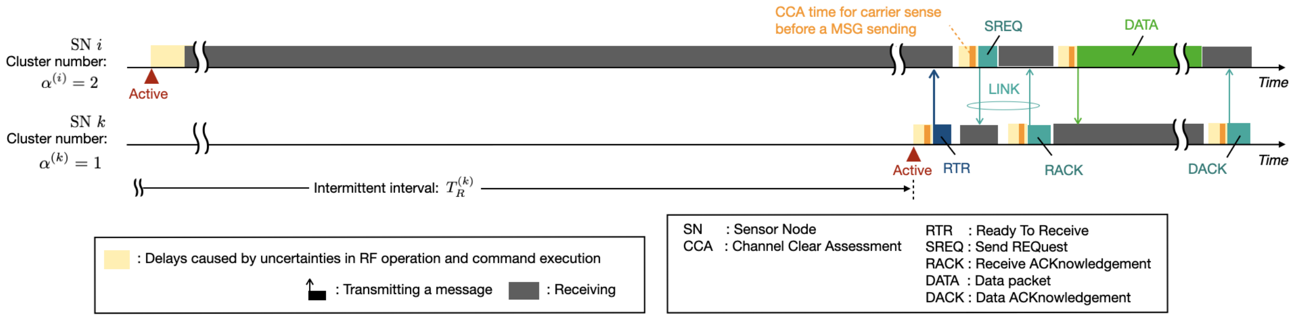

Fig. 2. Basic procedure of the IRDT protocol. Prolonged listening is required to receive an RTR message.

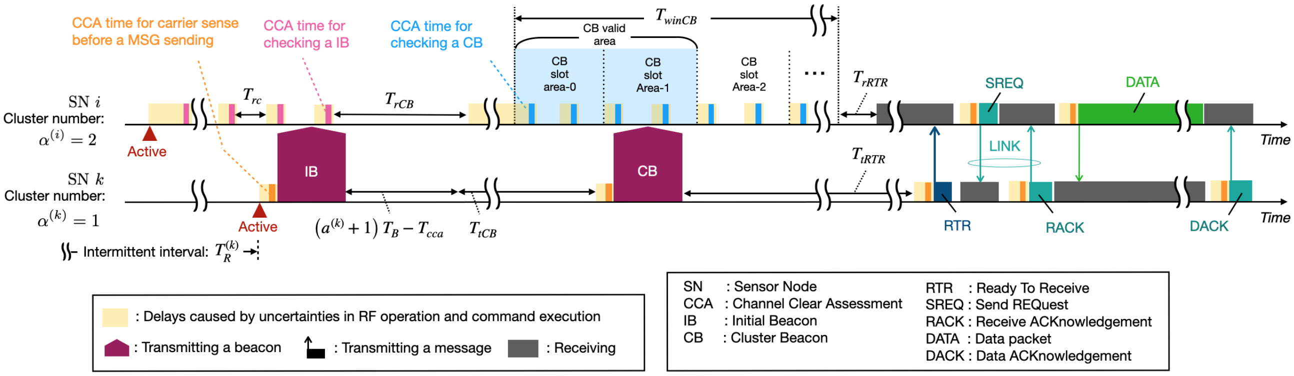

Fig. 3. Basic procedure of the improved ZEN-MAC protocol with a time window for CB detection, modified to better suit practical hardware operation.

C. *ZEN-MAC's Modification Using A Time Window for Hardware Implementation*

Regarding the first issue related to CCA control, our implementation-based investigation confirmed that repeated CCA functionality can be realized. However, if the CCA interval is not sufficiently long, the transition to RF sleep cannot be completed in time. Alternatively, using RF standby provides some reduction in power consumption, but achieving the level of power savings expected from the proposed method requires a full transition to sleep. The actual power consumption results are summarized and discussed in the subsequent section IV.

Additionally, in the conventional ZEN-MAC protocol, the receiver attempts to capture a CB by monitoring a predetermined time interval after receiving an IB, with the goal of detecting the CB from the target cluster. In [7], this interval was precisely controlled by defining time slots, and the method relied on determining the presence or absence of a single reception. However, in practical RF implementations and in command-processing processors that are not optimized for real-time communication tasks, non-negligible and irregular delays frequently occur. These delays make it difficult to reliably capture the CB at the intended timing. To address this issue, the proposed method repeatedly performs CCA over a fixed time window T_{winCB} s designated for CB detection. Fig. 3 illustrates the basic procedure of our proposed MAC protocol. The CPU records the timing of the first channel busy event detected by CCA and counts the number of consecutive busy occurrences within the window. Based on this information, it estimates whether a CB exists within the target interval, thereby improving the reliability of CB acquisition.

Although beacons in ZEN-MAC do not necessarily need to have a packet structure, in our implementation they include the same header format and identification structure as other packet types. This design allows the receiver to identify and discard any erroneously received beacons during consecutive RTR listening. This approach enables energy-efficient ZEN-MAC communication.

IV. EVALUATION AND DISCUSSION OF POWER CONSUMPTION

In this section, we evaluate the power consumption of each protocol. The measurement setup is shown in Fig. 4, which illustrates the environment used to measure the current drawn by the TI CC1312R1 module [8]. A regulated power supply provided a constant voltage of 3.3 V to the evaluation board, and the current was measured using a current probe. The current drawn by only the CPU during sleep was found to be on the order of a few microamperes, which is negligible. Therefore, in this analysis, we assume the power consumption during sleep to be 0 W. Although the energy consumption within an SN is influenced by various factors [9], we estimate the total required energy based on the measured current for each fundamental communication operation. The following

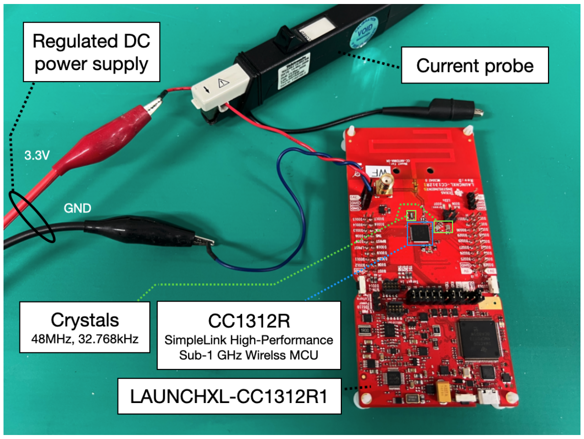

Fig. 4. Evaluation environment for measuring the current of the off-the-shelf TI CC1312R SimpleLink High-Performance Sub-1 GHz Wireless MCU (LAUNCHXL-CC1312R1) [8]. During measurement, the jumper pins are disconnected to prevent current from flowing into an XDS110 debugger.

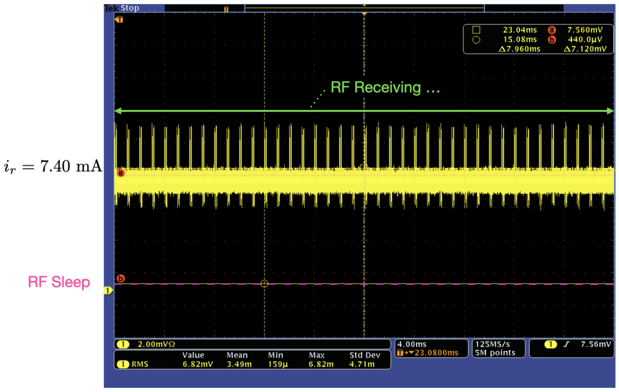

Fig. 5. Measured current consumption during continuous RF receiving reception

section presents the measurement results and provides a detailed analysis of the power consumption for each protocol.

A. IRDT: Continuous RF Reception

Fig. 5 shows the measured current consumption during continuous reception while waiting for RTR in the IRDT protocol. Therefore, the power consumption P_I W during continuous RF reception is given as follows:

$$P_I = i_r v, \tag{1}$$

where v is the voltage applied from the regulated power supply. Table I summarizes each current consumption and time.

B. ZEN-MAC: CCA Operation

Next, Fig. 6 shows the measured current consumption for repeated CCA operations to receive RTR in the ZEN-MAC protocol. By carefully designing the interval duration, it is possible to enable RF sleep between CCA cycles, further reducing power consumption. This adjustment process needs to be performed in advance through empirical measurements.

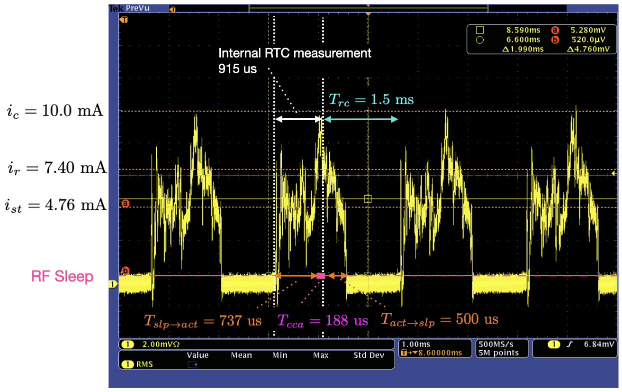

Fig. 6. Measured current consumption during CCA with RF sleep execution.

The power consumption of CCA cycles with RF sleep, P_{ZL} W, is calculated as follows:

$$P_{ZL} \approx \frac{i_{st} v T_{slp\to act} + i_c v T_{cca} + i_{st} v T_{act\to slp}}{T_{slp\to act} + T_{cca} + T_{rc}}. \tag{2}$$

Similarly, when the CCA execution interval is short and RF sleep cannot be used, the power consumption P_{ZH} W when using the standby state instead can be calculated as follows:

$$P_{ZH} \approx \frac{i_r v T_{stb\to act} + i_c v T_{cca} + i_r v T_{act\to stb} + i_{st} v (T_{rc} - T_{act\to stb})}{T_{stb\to act} + T_{cca} + T_{rc}}. \tag{3}$$

C. Comparison of Power Consumption for RTR Reception

To discuss the power consumption of the conventional continuous reception method versus CCA method, we'll now summarize the CCA patterns described. P_Z W can be expressed in the following three types:

$$P_Z = \begin{cases} P_{ZH} & \text{for CCA without RF sleep,} \\ P_{ZL} & \text{for CCA with RF sleep,} \\ P_{ZI} & \text{for ideal CCA without execution delay.} \end{cases} \tag{4}$$

For comparison, let's assume the power consumption P_{ZI} is $P_{ZI} \approx i_c v T_{cca} / (T_{cca} + T_{rc})$ when only CCA can ideally be executed without delay.

To consider power saving for RTR reception, Fig. 7 shows a comparison of the power consumption required before RTR reception when the CCA sleep interval is varied. From Fig. 7, it is evident that without using a method that incorporates RF sleep, the power consumption can only be reduced to the level of RF standby power. To achieve sufficient power savings, it is necessary to extend the CCA sleep interval to a certain degree and to use a method that enables RF sleep. The threshold values of 1.0 ms and 1.5 ms in Fig. 7 represent time settings that were adjusted and verified on an actual device to ensure proper operation.

However, extending the CCA sleep interval T_{rc} may require longer beacon transmission durations prior to repeated RTR transmissions, which can potentially increase power consumption. Taking this into account, the next section evaluates

TABLE I
MEASURED POWER CONSUMPTION PARAMETERS OF THE MODULE

Symbol	Detail	Value
i_t	Average current during RF transmitting. (Transmission power : 0 dBm)	7.96 mA
i_r	Average current during RF receiving	7.40 mA
i_c	Average current during RF wake-up preparation and during CCA	10.0 mA
i_{st}	Average current consumption during RF standby	4.76 mA
v	Operation voltage	3.3 V
$T_{stb\to act}$	Delay time (RF standby → RF active)	157 us
$T_{act\to stb}$	Delay time (RF active → RF standby)	240 us
$T_{slp\to act}$	Delay time (RF sleep → RF active)	737 us
$T_{act\to slp}$	Delay time (RF active → RF sleep)	500 us
T_{cca}	CCA is set to 128 µs, and an extra 60 µs is added as a stabilization period through configuration.	188 us

s = second, V = volt, A = ampere

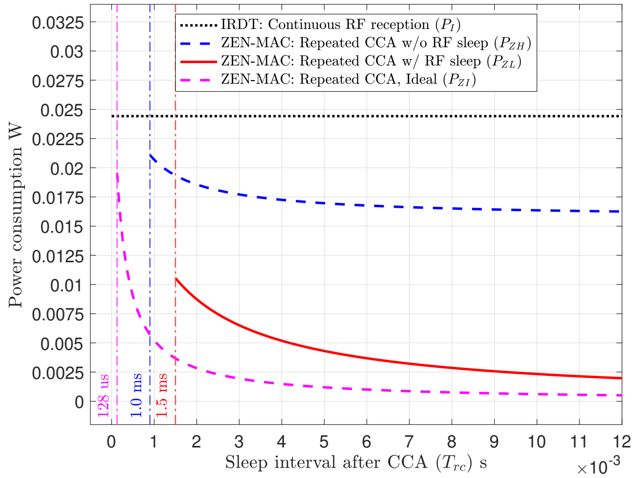

Fig. 7. Comparison of power consumption required before RTR reception using different methods.

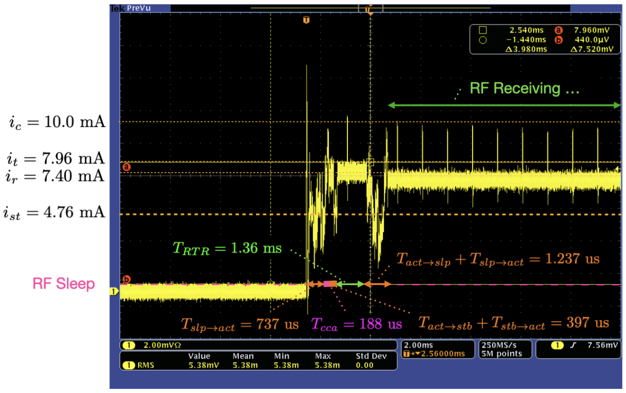

Fig. 8. Measured current consumption during RTR sending operation

whether power savings can still be achieved when the entire communication sequence is considered.

V. POWER CONSUMPTION EVALUATION UNDER PRACTICAL COMMUNICATIONS SCENARIOS

In the previous section, we discussed only the power consumption before the process of RTR reception. Next, we evaluate whether the power saving of ZEN-MAC can be achieved under practical communication scenarios. To clarify the power consumption required for each communication, Fig. 8 presents the measurement results of the current consumption associated with RTR transmission.

In actual operation, the transmission of each signal includes the transition time associated with each RF control. The energy consumption that takes into account the packet transition operations during the transmission of each signal can be calculated by

$$\begin{aligned} e_t(T_M) \triangleq{}& i_{st} v T_{slp\to act} + i_c v T_{cca} \\ &+ i_{st} v (T_{act\to stb} + T_{stb\to act}) + i_t v T_M + i_{st} v T_{act\to slp}, \end{aligned} \tag{5}$$

where T_M represents the communication time of each packet. Also, the energy consumption during the reception of a packet can be given by

$$e_r(T_M) \triangleq i_{st} v T_{slp\to act} + i_r v T_M + i_{st} v T_{act\to slp}. \tag{6}$$

Then, from Fig. 8, the energy consumption E_{tRTR} J for a single RTR transmission until timeout can be given by

$$E_{tRTR} \approx e_t(T_{RTR}) + e_r(T_{ws}). \tag{7}$$

Hereinafter, power consumption is evaluated based on the measurements from the previous section and the theoretical results derived in [7]. Note that some errors in [7] have been corrected, and therefore several equations differ from those presented previously. The power consumption of SN i, denoted as $E^{(i)}$ W, is expressed as

$$\begin{aligned} E^{(i)} ={}& \frac{E_{Rx}}{T_R^{(i)}} \left\{ 1 - \frac{\rho^{(i)}}{T_U} \left(\tau^{(i)} + T_{Dt} + T_{Dr} \right) \right\} \\ &+ \frac{\rho^{(i)}}{T_U} \left(\tau^{(i)} i_r v + E_{Dt} + E_{Dr} \right), \end{aligned} \tag{8}$$

where $\rho^{(i)}$ is the packet possession probability of SN i per unit time T_U, E_{Rx} represents the energy consumption required to wake up from the sleep state and transmit an RTR, E_{Dt} and E_{Dr} respectively represent the total energy consumption associated with the transmission and reception of each DATA message, and $\tau^{(i)}$ denotes the time from the start of communication partner discovery until the arrival of an RTR.

Specifically, E_{Rx} J is given by

$$E_{Rx} = \begin{cases} E_{tRTR}, & \text{for IRDT,} \\ E_{tRTR} + 2e_t(T_B), & \text{for ZEN-MAC.} \end{cases} \tag{9}$$

For the transmission of a single DATA packet, the total energy consumption is given by

$$\begin{aligned} E_{Dt} ={}& e_r(T_{RTR}) + e_t(T_{SREQ}) + e_r(T_{RACK}) \\ &+ e_t(T_{DATA}) + e_r(T_{DACK}). \end{aligned} \tag{10}$$

Similarly, the energy required to receive a single DATA packet is expressed as

$$E_{Dr} = e_r(T_{SREQ}) + e_t(T_{RACK}) + e_r(T_{DATA}) + e_t(T_{DACK}), \tag{11}$$

TABLE II

MAC PROTOCOL IMPLEMENTATION PARAMETERS

Symbol	Detail	Value
T_B	Transmission time of an IB and CB packet	5.04 ms
T_{RTR}	Transmission time of an RTR packet	1.36 ms
T_{SREQ}	Transmission time of an SREQ packet	1.44 ms
T_{DATA}	Transmission time of a DATA	10.4 ms
T_{RACK}	Transmission time of a RACK packet	1.44 ms
T_{DACK}	Transmission time of a DACK packet	1.44 ms
T_{ws}	Timeout counter (SREQ receiving)	4.0 ms
T_{rc}	Sleep interval after CCA in Fig. 3	$4,000$ us
T_{rCB}	Delay to CB Reception Stage in Fig. 3	5.0 ms
T_{rRTR}	Delay to RTR Transmission in Fig. 3	10.0 ms
T_{winCB}	Observation Window for CB in Fig. 3	50.0 ms
T_{tCB}	Delay to CB Transmission in Fig. 3	10.0 ms
T_{tRTR}	Delay to RTR Reception Stage in Fig. 3	95.0 ms

where T_{Dt} and T_{Dr} represent the total communication times corresponding to E_{Dt} and E_{Dr}, respectively. Table II summarizes the parameters required for these calculations.

The time $\tau^{(i)}$ from the start of communication partner discovery until the arrival of RTR is written by

$$\tau^{(i)} = \begin{cases} \tau_I^{(i)} & \text{for IRDT,} \\ \tau_{ZL}^{(i)} & \text{for ZEN-MAC: CCA with RF sleep.} \end{cases} \tag{12}$$

Then, $\tau_I^{(i)}$ is given by

$$\tau_I^{(i)} \approx \\ T_U \sum_{t=1}^{T^{(i)}_{R\,\min}/T_U} \left[\sum_{n=1}^{|\mathcal{R}(i)|} \frac{tT_U}{T_R^{(n)}} \left\{ \prod_{l \in \mathcal{R}(i) \setminus \{n\}} \left(1 - \frac{tT_U}{T_R^{(l)}} \right) \right\} \right], \tag{13}$$

where $T^{(i)}_{R\,\min}$ indicates the shortest intermittent interval among the SNs in $\mathcal{R}(i)$, $\mathcal{R}(i)$ denotes the set of SNs in the upper cluster with which SN i can communicate, and l and n respectively denote the indices of the SNs. Considering the power consumption ratio during RF operation, we obtain $\tau_{ZL}^{(i)} = \tau_I^{(i)} P_{ZL}/P_I$. Based on these values, the average power consumption $\bar{E}$ W of the entire network, excluding the GW, can then be calculated.

Fig. 9 illustrates the average power consumption $\bar{E}$ W, theoretically analyzed based on the measured values, when the intermittent interval of all nodes is uniformly extended. The results demonstrate that ZEN-MAC achieves lower power consumption than IRDT. Furthermore, ZEN-MAC provides substantial reductions in power consumption as the intermittent interval becomes longer. It is noted that for both protocols, the end-to-end packet loss rate remains below 0.1 when T_R is between 1.0 and 24.5 s.

VI. CONCLUSIONS

In this paper, we implement the energy-efficient ZEN-MAC protocol on the off-the-shelf TI CC1312R SimpleLink high-performance Sub-1GHz wireless MCU to evaluate its power-saving performance and practical feasibility. To address constraints in RF control during implementation, we propose an improved version of the ZEN-MAC protocol that introduces a fixed time window to prevent the omission of CB observations and ensure reliable operation. Experimental power measurements confirm that the proposed CCA-based neighbor discovery approach achieves significantly lower power consumption compared to the continuous reception method employed in the conventional IRDT protocol. Future work will explore the realization of highly reliable communication in large-scale networks and examine the behavior of dynamic intermittent interval control under energy harvesting (EH) power sources.

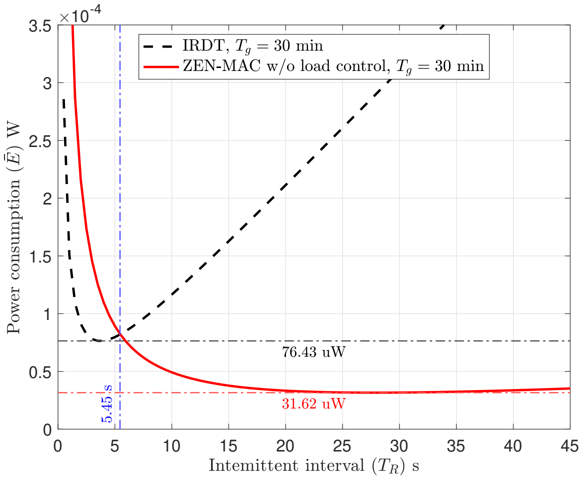

Fig. 9. Power consumption comparison through theoretical analysis evaluated based on measured values. T_g is DATA generation interval.

ACKNOWLEDGMENT

This work was supported by JST, CRONOS, Japan Grant Number JPMJCS24N1.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, 2002.

[2] L. D. Xu, W. He, and S. Li, "Internet of Things in Industries: A survey," *IEEE Trans. Ind. Informat.*, vol. 10, no. 4, pp. 2233–2243, 2014.

[3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications," *IEEE Commun. Surveys Tut.*, vol. 17, no. 4, pp. 2347–2376, 2015.

[4] S. Ergen and P. Varaiya, "On multi-hop routing for energy efficiency," *IEEE Commun. Lett.*, vol. 9, no. 10, pp. 880–881, 2005.

[5] H. Kawabata, K. Ishibashi, S. Vuppala, and G. T. F. de Abreu, "Robust Relay Selection for Large-Scale Energy-Harvesting IoT Networks," *IEEE Internet of Things J.*, vol. 4, no. 2, pp. 384–392, 2017.

[6] D. Kominami, M. Sugano, M. Murata, and T. Hatauchi, "Energy-Efficient Receiver-Driven Wireless Mesh Sensor Networks," *Sensors*, vol. 11, no. 1, pp. 111–137, 2011.

[7] T. Kawaguchi and K. Ishibashi, "ZEN-MAC: Zero Excess Node MAC for Multihop Sensor Networks With Energy Harvesting," *IEEE Internet of Things J.*, vol. 12, no. 14, pp. 27 818–27 834, 2025.

[8] "LAUNCHXL-CC1312R1 Launchpad (Simplelink CC1312R Wireless MCU Launchpad for 868MHz/915MHz frequencybands) development kit," Texas Instruments Inc., 2018.

[9] N. Thiyagarajan and N. Shanmugasundaram, "An Investigation on Energy Consumption in Wireless Sensor Network," in *2022 8th Int. Conf. on Adv. Comput. and Commun. Syst. (ICACCS)*, vol. 1, 2022, pp. 1359–1364.