2021 IEEE 46th Conference on Local Computer Networks (LCN) | 978-1-6654-1886-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/LCN52139.2021.9524937

Enabling Robust Wireless Communication for BMS on Electric Vehicles

Fabian A. Rincon Vija[1,2], Samuel Cregut[1], Georgios Z. Papadopoulos[2], Nicolas Montavont[2]
[1] Renault Group, Guyancourt, France
{fabian-antonio.rincon-vija, samuel.cregut}@renault.com
[2] IMT Atlantique, IRISA, France
{georgios.papadopoulos, nicolas.montavont}@imt-atlantique.fr

***Abstract*—Battery Management System (BMS) is a critical part of Electric Vehicles (EVs). The introduction of a wireless communication and networking inside the BMS in order to replace the traditional wired bus brings multiple benefits. As it is a critical application, the network has stringent requirements such as high reliability, low energy consumption, and bounded latency. In this paper, we propose a network architecture based on an enhanced version of the IEEE Std 802.15.4 Time Slotted Channel Hopping (TSCH) Medium Access Control (MAC) mode running over the physical layer of Bluetooth Low Energy (BLE). To orchestrate the data transmissions, we present a reliable and predictable schedule based on the Low Latency Deterministic Network (LLDN) Group Acknowledgement (GACK) method which dynamically manages the retransmissions. We implement a WBMS for the Renault Zoe battery pack to demonstrate that our proposed architecture achieves 100% of network reliability with bounded latency, and low energy consumption.**

***Index Terms*—Electric Vehicles, EV, Battery Management System, BMS, Wireless BMS, IoT, Industrial IoT, IEEE Std 802.15.4, TSCH, LLDN, GACK.**

I. INTRODUCTION

Electric Vehicles (EVs) are considered as the present and future of transportation thanks to their several advantages such as low CO_2 emission, energy efficiency and low operating cost. In an EV, the battery is one of the most important and critical components. Typically, a battery is divided into modules that are composed of cells. The Battery Management System (BMS) is a critical system in charge of monitoring the battery, and ensuring its smooth operation. To do this, the BMS has a Central Processing Unit (CPU) called master BMS which receives all the battery cell voltages and temperatures from the BMS slaves. Then, it computes the State of Charge (SOC), the State of Health (SOH), and generates cell balancing orders. In a traditional system, the slaves send this data every 100 ms to the master over a wired connection, commonly a Universal Asynchronous Receiver Transmitter (UART) daisy chain, as you can see in Fig. 1.

Lately, the industry has seen the interest of removing the wired bus and replacing by a wireless link between the BMS master and the slaves. Wireless BMS (WBMS) can bring multiple advantages, but replacing a wired bus with a wireless link is a challenging task since it has to guarantee deterministic latency, high reliability and low energy consumption. Towards this aim, in 2012, the IEEE Std 802.15.4 introduced Time Slotted Channel Hopping (TSCH) and Low Latency Deterministic Network (LLDN) Medium Access Control (MAC) protocol modes. Therefore, in this paper, we propose a highly reliable and low power TSCH-based wireless network architecture that can be applied to critical applications such as the WBMS. The contribution of this paper is threefold: *i)* we propose a dynamic retransmission schedule for a TSCH star topology network, which uses the LLDN Group Acknowledgement (GACK) mechanism to achieve bounded latency and high network reliability. *ii)* we present hardware and software considerations to take into account when implementing our proposed architecture on different platforms. *iii)* finally, we demonstrate how our WBMS architecture can achieve a reliability performance of 100% when running in a Renault Zoe battery pack.

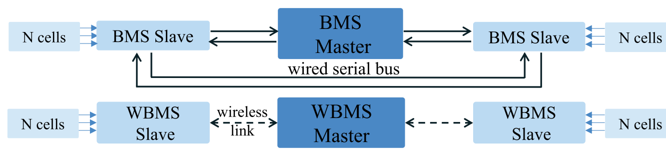

Fig. 1: From BMS to Wireless BMS.

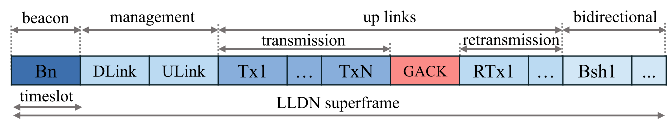

Fig. 2: LLDN superframe with dedicated slot for GACK [1].

II. TECHNICAL BACKGROUD & RELATED WORK

In this section, we provide an overview of the LLDN and TSCH MAC protocols.

A. Low Latency Deterministic Networks (LLDN)

The LLDN MAC scheme only supports star network topology and relies on the Time Division Multiple Access (TDMA) mechanism to achieve high reliability and low energy consumption. In TDMA the continuous time is divided into short intervals called timeslots. At each timeslot, a node can transmit or receive a data, or go to "sleep" mode to save energy [1]. The standard defines four types of timeslots, see Fig. 2. LLDN groups a set of timeslots into a larger structure called superframe, which is repeated periodically. The standard indicates that the number of retransmission timeslots can not be higher than half of the transmission ones, and that each node can have at most one retransmission timeslot per superframe [1].

LLDN proposes to use the Group Acknowledgement (GACK) mechanism to synchronize the distribution of the retransmission timeslots. The principle of GACK is that the root receives the data from the nodes without responding with individual ACK at the end of each timeslot. Then, the root sends a group ACK in broadcast to inform which messages were correctly received. Based on this GACK, the nodes know if they can go to sleep or if they have to retransmit at a specific timeslot [1]. LLDN uses the same radio channel for all the message exchange between the nodes. The standard does not propose a timeslot length because it depends on the payload required by the application [1], [2].

In the literature, several works have been proposed to overcome the deficiencies of LLDN protocol. Dariz et al. [3] proposed to use a different length for each timeslot to adapt the network to different payload sizes. Using relays to retransmit data on LLDN is proposed by Berger et al. [4]. Patti et al. [5] proposed to dynamically allocate timeslots according to the traffic. Finally, Takamori et al. [6] proposed to use GACK on a star topology network, with orthogonal spreading code, to receive many messages at the same time.

B. Time Slotted Channel Hopping (TSCH)

The TSCH MAC mode supports both star and mesh topologies. This method uses TDMA and Frequency Division Multiple Access (FDMA) to achieve high reliability and low power wireless communication [7]. Similar to LLDN, TSCH splits the time into small intervals called timeslots of $10ms$, and the group of timeslots which repeats periodically is called slotframe. During at each timeslot, the nodes are able to transmit a packet and to receive an acknowledgement [8]. In TSCH, a transmitting node uses different radio channels for two subsequent packet transmissions, i.e., channel hopping [7]. Using different channels for each timeslot makes the network more robust to interference and multipath problems [9].

Contrary to LLDN, TSCH does not propose how the network should schedule the data exchange among the nodes, they leave this task to the upper layers. The scheduling function allocates a certain number of cells to each transmitter and receiver, which indicates when the nodes must transmit, receive a packet, or remain idle, and which radio channel they have to use [7]. Several scheduling algorithms have been proposed so far, they can be classified into centralized, distributed, and hybrid approaches [10]. In a centralized approach, a unique entity (typically the root) creates a schedule based on the network architecture [11]. For the distributed approach, the nodes negotiate the schedule with their neighbors [12]. Compared to the distributed approach, centralized method has proven to be more adequate for fairly static networks with deterministic traffic [13].

III. Wireless Communication for Critical Systems

In this section, we present our TSCH-based architecture which uses the LLDN GACK mechanism to manage retransmissions to achieve highly reliable, bounded latency and low power wireless networking for critical applications. First, we present an overview of the proposed schedule, then we detail the retransmission management mechanism and, finally, we present the innovative physical layer approach as well as the hardware considerations.

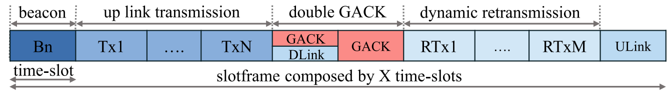

Fig. 3: The proposed dynamic retransmission schedule.

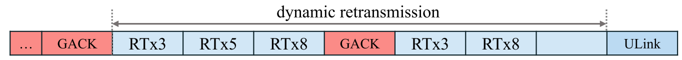

Fig. 4: An example of the dynamic retransmission procedure.

A. Proposed Architecture

Typically, Wireless Sensor Networks (WSNs) are composed of N nodes that send their measurements to the root every T milliseconds. Our proposed network consists of a star topology, employs a centralized scheduling function, and the LLDN GACK mechanism to synchronize the dynamic retransmission. The TSCH MAC protocol is used at MAC layer, to take advantage of time division for collision avoidance, and radio channel hopping to mitigate the external interference. Fig. 3 presents our proposed dynamic retransmission schedule for a network of N nodes. As it can be observed, the first timeslot is reserved for the beacon frame. Then, we have N dedicated transmission timeslots for the N slave nodes where the acknowledgement field (from the root to the slave nodes) is removed. These timeslots are distributed according to the node ID, which the root assigns to each node during the joining stage.

To efficiently manage the retransmission timeslots, the root has two dedicated timeslots to send the GACK frame to inform the slave nodes about the messages that were not received by the root. Then, the slave nodes define which retransmission timeslots they can use, or, if the message was successfully delivered. The network requires two management timeslots, Up Link (ULink) and Down Link (DLink) that are used to exchange the joining request and response. The timeslot N+1 is used for dual purposes, as a DLink management timeslot, or when there is no joining response message from the root on the queue, it is used to send a GACK.

B. Dynamic Retransmission Algorithm

In this study, we focus on a dynamic retransmission method rather than a classic static method, for the following two reasons. First of all, this mechanism provides several transmission opportunities to the nodes to send their data, within the same slotframe. In a static method, the nodes would have at most one retransmission opportunity within a slotframe. The second reason is the low energy consumption operation. Indeed, the dynamic retransmission approach allows the nodes to send their data only when is required and, thus, they can go to "sleep" mode when their task is completed.

Let us assume a network of 10 nodes and 7 retransmission timeslots. In Fig. 4, an example of our proposed dynamic retransmission method is depicted. In this case, the root sends two times the GACK indicating that it did not receive the messages from nodes 3, 5 and 8. The three slave nodes retransmit their message using the first three timeslots. The node with the lowest ID should uses the first retransmission timeslot, the next node ID

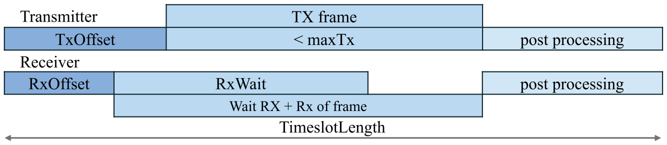

Fig. 5: The proposed TSCH timeslot structure.

uses the second one and so on. In some cases, one retransmission is not enough and, thus, after the first three retransmission timeslots, the root sends another GACK indicating which messages are still missing, in this example messages from nodes 3 and 8. The process can continue until the end of the slotframe, or until the root has received all the messages. The maximum number of retransmission opportunities is a configurable parameter, and it depends on several factors such as the number of timeslots that is possible to have in a slotframe, or the interference level. Note that there is a trade-off between network reliability and power consumption, the higher the number of retransmission opportunities, the higher the energy consumption.

C. Physical Layer & Hardware Considerations

The IEEE Std 802.14.5 2020 standard proposes to use an OQPSK modulation at $250kb/s$ with 16 radio channels in the $2.4GHz$ band and a timeslot length of $10ms$ [8]. However, a timeslot of $10ms$ leads to very long slotframe sizes which in critical applications are unacceptable. Moreover, 16 radio channels may not be sufficient for certain industrial environments. To overcome these issues, we propose to employ the TSCH MAC layer over the Bluetooth Low Energy (BLE) physical layer, controlled by our proposed dynamic retransmission schedule. The BLE PHY operates in the $2.4GHz$ band as well, it comes with 40 radio channels, and GFSK modulation with $2Mb/s$ data rate, which allow us to exchange one frame in a shorter time [14].

To reduce the $10ms$ timeslot length proposed by the IEEE Std 802.15.4 [8], there are two points to consider, the maximum payload to be sent over the wireless medium, and the necessary time for post processing that the hardware requires at the end of each timeslot. In our schedule, we do not have to execute the Clear Channel Assessment (CCA) mechanism at the beginning, because all the timeslots are reserved for dedicated nodes. Thus, the Tx and Rx offset values can be reduced to the minimum. These minimum values depend on the time required by the hardware to wake up and initialize all the necessary peripherals. In addition, thanks to the GACK mechanism in our schedule, the nodes do not need to transmit the ACK frames. Therefore, we can delete the ACK field at the end of the timeslots which further reduces its length. In Fig. 5, the proposed TSCH timeslot structure is depicted.

IV. WBMS Network Application

Wireless Battery Management System (WBMS) is the target application for our proposed TSCH-based schedule with dynamic retransmission. In this section, we present how we designed and implemented a robust WSN for the battery pack of a Renault Zoe. The Renault Zoe battery pack has 12 modules each containing 8 cells, thus, our network should be composed by a root, the WBMS master, and 12 nodes, the WBMS slaves. For this application,

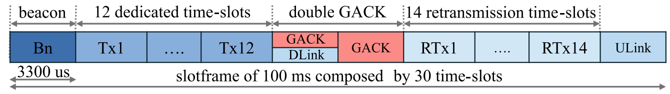

Fig. 6: Renault Zoe WBMS schedule.

the root must receive all the 96 voltage measures every 100 milliseconds, the amount of data packets successfully delivered to the root must be higher than 99.999%, and the average current consumption of each node (WBMS Slaves) must be less than 1 mA. Additionally, the nodes should receive every second a broadcast message from the root with the cell balancing commands.

A. Experimental Setup: Hardware & Software Considerations

The selected hardware for the nodes and the root is the Texas Instrument (TI) cc26x2 microcontroller. The network was developed using 13 TI Simplelink cc26x2 Launchpad (1 root and 12 slaves). For the experiments on the Renault Zoe battery pack, the launchpad nodes were replaced by a custom board composed by a cc2642 for the network management and a BMS ASIC (TI bq79616) in charge of the battery cell measurements. Regarding the software, we were allowed to use and modify the proprietary implementation of the IEEE Std 802.15.4 provided by Texas Instrument.

The first step of the implementation was to define the timeslot length for this platform. Toward this aim, we set a simple schedule with a slotframe composed of 13 timeslots, the first one for the beacon and the following 12 dedicated for each node. Then, we executed several experiments changing the total length of the timeslot and Tx/Rx offset value. The objective was to observe in which cases the nodes were able to maintain synchronization, and have sufficient time to finish the necessary tasks, for the timeslot operation, before the start of the next timeslot. Finally, we obtained a timeslot lenght of $3.3ms$, with a Rx Offset of $300\mu s$ and a guard time of $300\mu s$, see Fig. 5.

B. Dynamic Schedule Definition

Considering a timeslot length of $3.3ms$, and a slotframe of $100ms$, we can have 30 timeslots. In our proposed solution, we have the first timeslot allocated for the beacon frame, and 12 dedicated timeslots for the slaves, one for each node. Then, we have 2 timeslots for the GACK, one of which is used also for the joining responses from the root. Next, we have 14 timeslots for the dynamic retransmission mechanism and the last timeslot for the joining requests from the nodes, as it is illustrated in Fig. 6. For this application we did not set the maximum number of retransmission that a frame can have within a slotframe. As it can be observed, even in the worst case scenario where all the message transmissions fail in the first 12 dedicated timeslots, the proposed schedule can support a retransmission round for all nodes within the same slotframe. Moreover, if any message transmission fails, it can be reattempted more than once to ensure high reliability. Regarding the downlink traffic, we have chosen to send the cell balancing orders in the same GACK frames.

C. WBMS Network Performance Results

To validate our proposed framework, we executed several experiments to demonstrate that it meets the BMS requirements

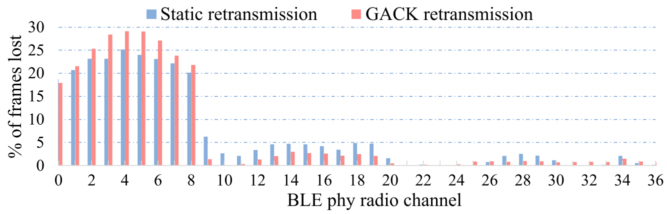

Fig. 7: Percentage of messages lost per channel for the static and dynamic retransmission tests.

TABLE I: Reliability results.

	No battery cover	With battery cover
Total msg	60 086 640	58 870 164
Lost before Rtx	200 200	21 193
Lost with dynamic Rtx	0	0

of the Renault Zoe. First, we compare our proposed schedule against a static retransmission approach. The difference is that, in the first one, we employed the GACK mechanism to achieve dynamic retransmission, as presented in section III, while in the second case, the schedule does not change at each slotframe, so the nodes have only one dedicated timeslot to retransmit their message. For both cases, we set up a network of 1 root and 12 nodes placed on a desk, using 13 TI cc26x2 launchpads, where each node sends one frame to the root every 100 ms during two hours. We intentionally produce electromagnetic interference by generating constant traffic over the WiFi radio channels 1 and 6. In Fig. 7 is depicted the percentage of messages lost per channel during the tests. After two hours the nodes sent almost 800 000 frames to the root. For both cases, there were around 55 000 lost messages prior to retransmission. In the static retransmission case, the network definitely lost 7 277 messages after the second try. On the other hand, using our proposed framework with the GACK mechanism, the network lost only 11 messages.

We conducted a second experiment during a week on an actual battery pack of the Renault Zoe. In this experiment, we have 1 root (TI cc26x2 launchpad) and 12 slaves (WBMS node custom board). Each WBMS slave should measure the 8 cell voltages of a module, group them in a frame, and transmit it to the root every 100 ms. If a frame does not reach the root within a slotframe (100ms), it is considered as a lost message. The test was executed twice, without the metal battery pack cover, and with the cover in place. Table I shows the results obtained from the experiments. For both cases the total messages exchanged were almost 60 million. By employing our proposed dynamic retransmission schedule, the total of frames lost were zero, which can be translated as network reliability of 100%.

Finally, using the TI EnergyTrace Technology [15], we measured the current consumption of the root, and the nodes in a network of 13 TI cc26x2 Launchpads. The measured average current consumption of each node was $0.42mA$. Considering this results, we demonstrate that with our proposed dynamic retransmission schedule, we are able to achieve a highly reliable and low energy WBMS that meets all the requirements set by the application.

V. Conclusions

Introducing Wireless BMS in EV brings multiple advantages, but at the same time several challenges. To fulfill the requirements of the WBMS network, we proposed to use a modified version of the IEEE Std 802.15.4 TSCH mode over the BLE PHY layer, and to reduce the timeslot length presented in the standard. In addition, we proposed to use the LLDN Group Acknowledgement (GACK) method to develop a highly reliable, low latency and energy efficient architecture. We validated our proposed framework by implementing a WBMS for a Renault Zoe battery pack. After conducting series of real-world experiments, the results show that our proposal can achieve a wireless network reliability of 100% with bounded latency and low energy consumption. Future work consists on EV driving experiments to evaluate the WBMS in multiple scenarios.

Acknowledgment

All the research results presented in this paper have been achieved in a joint work between the Renault Group, Texas Instrument and the IMT Atlantique.

References

[1] M. Anwar and Y. Xia, "IEEE 802.15. 4e LLDN: Superframe Configuration for Networked Control Systems," in *Proc. IEEE 33rd CCC*, 2014.

[2] "IEEE Standard for Local and Metropolitan Area Networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer," *IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011)*, 2012.

[3] L. Dariz, G. Malaguti, and M. Ruggeri, "Performance Analysis of IEEE 802.15.4 Real Time Enhancement," in *Proc. IEEE 23rd ISIE*, 2014.

[4] A. Berger, A. Entinger, A. Pötsch, and A. Springer, "Improving IEEE 802.15. 4e LLDN Performance by Relaying and Extension of Combinatorial Testing," in *Proc. IEEE (ETFA)*, 2014.

[5] G. Patti and L. L. Bello, "A Priority-aware Multichannel Adaptive Framework for the IEEE 802.15. 4e-LLDN," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 10, pp. 6360–6370, 2016.

[6] M. Takamori and Y. Yamao, "Enhancing Throughput of Star Topology Sensor Network by Group Acknowledgement Method and MCR SS-CSMA/CA," in *Proc. IEEE APCC*, 2015.

[7] X. Vilajosana, T. Watteyne, T. Chang, M. Vučinić, S. Duquennoy, and P. Thubert, "IETF 6TiSCH: A Tutorial," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 595–615, 2019.

[8] "IEEE Standard for Low-Rate Wireless Networks," *IEEE Std 802.15.4-2020 (Revision of IEEE Std 802.15.4-2015)*, pp. 1–800, 2020.

[9] T. Watteyne, A. Mehta, and K. Pister, "Reliability Through Frequency Diversity: Why Channel Hopping Makes Sense," in *Proceedings of the 6th ACM symposium on Performance evaluation of wireless ad hoc, sensor, and ubiquitous networks*, 2009.

[10] A. G. Rodrigo Teles Hermeto and F. Theoleyre, "Scheduling for IEEE 802.15.4-TSCH and Slow Channel Hopping MAC in Low Power Industrial Wireless Networks: A Survey," *Elsevier Computer Communications*, vol. 114, pp. 84–105, 2017.

[11] G. L. Gall, N. Montavont, and G. Z. Papadopoulos, "IoT Network Management within the Electric Vehicle Battery Management System," *Springer, Journal of Signal Processing Systems*, 2021.

[12] V. Kotsiou, G. Z. Papadopoulos, P. Chatzimisios, and F. Theoleyre, "LDSF: Low-latency Distributed Scheduling Function for Industrial Internet of Things," *IEEE Internet of Things Journal*, vol. 7, pp. 8688–8699, 2020.

[13] M. R. Palattella, N. Accettura, M. Dohler, L. A. Grieco, and G. Boggia, "Traffic Aware Scheduling Algorithm for Reliable Low-power Multi-hop IEEE 802.15. 4e Networks," in *Proc. IEEE PIMRC*, 2012.

[14] C. S. W. Group, "Bluetooth Core Specification v5.2," *Core Specification Working Group*, 2019.

[15] J. Lindh, C. Lee, M. Hernes, and S. Johnsrud, "Measuring CC13xx and CC26xx current consumption," Texas Instrument, Tech. Rep., 2019. [Online]. Available: https://www.ti.com/lit/an/swra478d/swra478d.pdf