



Acoustic Backscatter Network for Vehicle Body-in-White

Weiguo Wang
NIO & Tsinghua University
wangwg.wwg@gmail.com

Yuan He*
Tsinghua University
heyuan@tsinghua.edu.cn

Yadong Xie
Tsinghua University
ydxie@tsinghua.edu.cn

Chuyue Xie
Tsinghua University
xiexue24@mails.tsinghua.edu.cn

Yi Kai
NIO
yi.kai@nio.com

Chengchen Hu
NIO
huc@ieee.org

ABSTRACT

We present a novel approach to monitor the Body in White (BiW), the fundamental metallic structure of a vehicle. Existing monitoring methods, including both wired and wireless sensor systems, face significant challenges due to integration complexity, weight considerations, material costs, and signal blockage within the metallic environment. To overcome these limitations, we introduce ARACHNET, an acoustic backscatter network that leverages the conductive properties of the BiW to propagate vibration signals for energy transfer and data communication. This system comprises battery-free tags that harvest energy from BiW vibrations and utilize a backscatter technique for efficient communication, thereby eliminating the need for external power sources and reducing the power consumption. We address key challenges such as power sufficiency for tag activation and sustained operation, and collision reduction in network communication, by designing an ultra-low power backscatter tag and a distributed slot allocation protocol. We implement ARACHNET, and deploy 12 tags onto the BiW of an electric SUV car. The evaluation results show that the power consumption of the tag is $51.0 \mu\text{W}$ for uplink packet transmission, and $24.8 \mu\text{W}$ for downlink packet reception. With our network protocol, the slot utilization can be up to 81.2%.

CCS CONCEPTS

• **Networks** → **Network protocols**; • **Computer systems organization** → **Sensor networks**; • **Hardware** → **Wireless devices**;

KEYWORDS

Acoustic Backscatter; Battery-free Sensing; Structural Health Monitoring; Body-in-White; Distributed Slot Allocation

ACM Reference Format:

Weiguo Wang, Yuan He, Yadong Xie, Chuyue Xie, Yi Kai, and Chengchen Hu. 2025. Acoustic Backscatter Network for Vehicle Body-in-White. In *ACM SIGCOMM 2025 Conference (SIGCOMM '25)*, September 8–11, 2025, Coimbra, Portugal. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3718958.3750486>

*Yuan He is the corresponding author.



This work is licensed under a Creative Commons Attribution 4.0 International License. *SIGCOMM '25, Coimbra, Portugal*
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1524-2/25/09.
<https://doi.org/10.1145/3718958.3750486>

1 INTRODUCTION

Body in White (BiW) is the foundational metallic structure of a vehicle, providing structural integrity and ensuring vehicular safety. It requires high-strength materials capable of resisting bending and torsional forces. Moreover, it is crucial that BiW offers comprehensive crash protection, ensuring the safety of both the driver and passengers.

However, over time, the BiW's structural integrity may be compromised due to corrosion and the natural aging process. Such degradation can subtly undermine vehicular safety and performance, raising the need for regular structural health monitoring [10, 18, 31]. Despite its necessity, the monitoring of BiW's health often presents as impractical or infeasible.

Apparently, deploying sensors onto the BiW is the initial step towards structural health monitoring. Nevertheless, significant hurdles arise in this endeavor, presenting difficulties for both wired and wireless sensors.

Adopting wired sensors is impractical for several reasons:

- **Integration Complexity.** Modern vehicles already have complex wiring systems, with some containing over 3,000 wires totaling about 4,000 m [5]. Introducing wired sensors entails careful design to avoid physical and electromagnetic interference with existing systems [24].
- **Weight Addition.** Additional wiring increases the overall weight of the vehicle, negatively impacting both performance and efficiency. This is especially critical for electric vehicles, where excess weight directly reduces driving range and impairs acceleration.
- **Material and Manufacturing Cost.** The requirement for durable *automotive-grade* cables that can withstand the harsh conditions inside a vehicle's body significantly raises expenses. Moreover, the wire installation process itself is labor-intensive, which also inflates the manufacturing cost.

Despite the clear safety benefits of BiW monitoring, the industry's push to cut costs and weight makes using wired sensors both economically and practically questionable.¹

Seemingly, wireless sensors that use WiFi, BLE, and ZigBee technologies for communication could be considered more advantageous. Unfortunately, they are also infeasible in practice: (1) The

¹Tesla exemplifies the automotive industry's strong commitment to reducing wire usage in vehicles. The Tesla Model S, launched in 2012, contained approximately 3,000 m of wiring. This amount was halved to 1,500 m in the Model 3, launched in 2017. Furthermore, the Tesla Cybertruck continues this trend, with Tesla reportedly achieving a 68% reduction in total wiring compared to the Model 3. [9, 16, 41, 43].

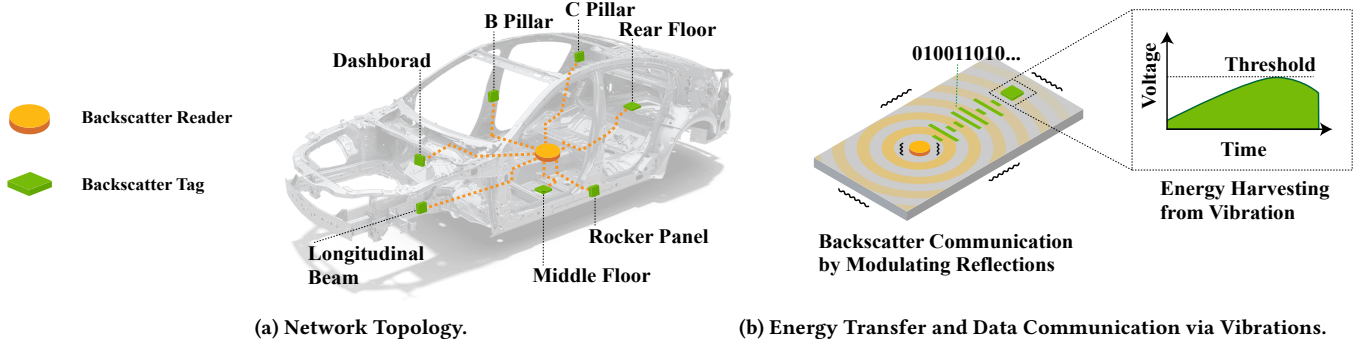


Figure 1: ARACHNET Illustration. (a) The reader and multiple battery-free tags are attached to the vehicle's BiW, which serves as a shared acoustic medium. (b) The reader emits acoustic vibrations through the BiW. Tags harvest energy from these vibrations and modulate the reflected signals via backscatter to transmit data. In this way, ARACHNET supports large-scale sensor deployment for structural health monitoring, without the constraints of wiring harnesses, battery maintenance, or signal blockage.

metallic nature of the BiW poses significant challenges for Radio-Frequency (RF) signal transmission, as the metal can severely impede RF signal propagation. (2) These wireless nodes are typically battery-powered, which necessitates regular battery replacements or recharging. The maintenance of these power sources becomes a cumbersome task, particularly due to the difficult access to sensors once they are embedded within the BiW.

We propose a novel perspective that *treating the BiW as a whole medium capable of propagating vibration signals*. Through the lens, we propose a wireless, battery-free network system via vibrations, ARACHNET. Rather than replacing existing wired or RF systems, ARACHNET offers a lightweight, scalable alternative for scenarios where conventional solutions are impractical. We believe this could redefine the BiW not merely as a passive structure, but as a power conduit and communication channel.

Interestingly, the trend towards Single-Piece Mega-Casting in modern BiW manufacturing aligns well with our proposal [4, 13]. This technique, involving replacing multiple smaller components with a single large casting, simplifies production and reduces joints and seams in the BiW, providing a more uniform medium for vibration propagation.

Fig. 1(a) shows that ARACHNET contains two components: multiple tags and a single reader, all directly attached to the BiW. The vibration within the BiW serves dual purposes, shown in Fig. 1(b): (1) **Energy Transfer**. The tags are designed to be battery-free and harvest energy from the BiW vibrations. The reader actively generates vibrations that propagate through the BiW. As these vibrations traverse the metallic structure, the tags directly harness this vibration energy, converting it into usable power for their operations. (2) **Data Communication**. The tags use a backscatter technique for communication. Instead of generating and transmitting their own vibration waves, which require significant energy, the tags modulate and reflect the incoming vibration waves sent by the reader. By altering the amplitude of reflections, the tags can transmit data back to the reader at a low power.

Designing ARACHNET presents three challenges.

Challenge 1: *How to activate tags deployed across BiW where reader power is restricted by electrical safety regulations?*

In human-accessible environments (i.e., BiW), the reader power should be intentionally restricted to meet electrical safety requirement. This limitation results in lower applied voltages, reducing the energy available to power tags. Thus, tags away from the reader may not receive sufficient energy to reach the activation threshold, leaving them dormant.

To address this issue, our solution focuses on two folds: firstly, increasing the voltage harvested from acoustic waves using multi-stage voltage amplifiers, and secondly, lowering the operational voltage thresholds of the tags. This ensures that even tags located at considerable distances from the reader can gather enough voltage to activate.

Challenge 2: *How to ensure continuous operation of tags with limited energy availability and inefficient energy conversion?*

Maintaining the long-term operation of activated tags within the BiW is crucial for tag to participate in the backscatter network. However, it poses substantial challenges, primarily due to two critical factors. First, the inherently limited power supply from the reader, mentioned above, restricts the energy available to the tags. Second, while multi-stage voltage amplifiers are effective at increasing the voltage necessary for activation, they also introduce inefficiencies in energy conversion, challenging long-term tag operation.

To address this, we develop a low-power software architecture for tags, where all CPU activities—including modulation, demodulation, and network operations—are driven by interrupts. This keeps the CPU in low-power mode as much as possible, and awake it only when necessary, allowing sustained tag operation despite the limited power available.

Challenge 3: *How to coordinate tags with diverse transmission requirements under high collision rates, while maintaining limited communication overhead?*

Tag coordination in BiW faces three fundamental constraints. First, different BiW monitoring tasks require varying reporting frequencies. For example, tags monitoring structural aging need less frequent updates, while those tracking critical battery pack conditions require more frequent communication.² Second, the inherently low throughput of acoustic backscatter communication leads to longer uplink (UL) packet durations, which increases the

²This is crucial as battery damage can pose safety risks, including fires.

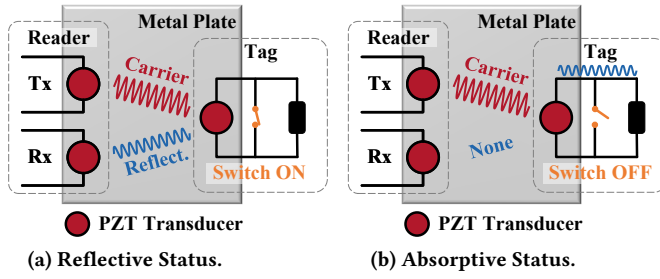


Figure 2: Basic Principle of PZT Backscattering. (a) When the PZT is short-circuited, incoming vibrations are reflected. (b) When open-circuited, the PZT absorbs the vibrations. Toggling between these two states enables low-power On-Off Keying (OOK) modulation.

likelihood of packet collisions. Most importantly, the strict power constraints of tags make it infeasible for them to receive detailed downlink (DL) feedback from the reader. Since coordination requires frequent DL packets, longer DL packet would quickly deplete the limited energy available to tags during reception, compromising their continuous operation.

To address this, we propose a novel network protocol specifically tailored to the unique conditions of the BiW. Our protocol leverages the predictable traffic patterns of the tags by implementing a distributed slot allocation. With a low communication overhead, this approach minimizes collisions and allows for the customization of tag transmission rates.

While ARACHNET is tailored to vehicle BiW, we believe our hardware-software co-design might provide insights for energy-sensitive applications in general. Meanwhile, our distributed slot allocation mechanism can be applied to coordinate tags with predictable but diverse transmission rates in other IoT systems. Our contributions are as follows:

- We propose ARACHNET, a first-of-its-kind complete acoustic backscatter network system that enables deploying battery-free and wireless sensors onto BiW. Comparison with existing solutions is provided in Appendix D.
- We implement an ultra-low-power tag that can be activated and sustain continuous operation, with $51.0 \mu\text{W}$ for transmission and $24.8 \mu\text{W}$ for reception.
- We propose a distributed backscatter protocol that reduces packet collisions and supports different transmission rates for different tags, achieving up to 81.2% slot utilization. The convergence of the protocol is also proved.

2 BACKGROUND

2.1 Body in White

The term BiW refers to the core framework of a vehicle's structure. It originally represents the stage in automotive manufacturing where the vehicle's frame is assembled before the addition of major components like the engine and exterior trim. In contemporary contexts, this term has expanded to include the entire structural framework, made predominantly from sheet metal, which comprises doors, roofs, and the frame. The BiW is fundamental not only for defining the vehicle's dimensions and structural integrity but also for ensuring its performance and safety throughout its

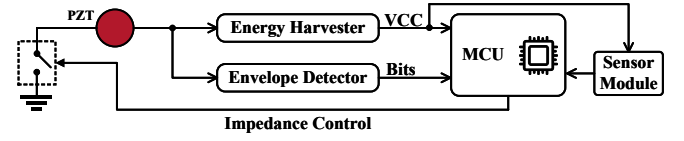


Figure 3: Design Overview of Tag Hardware. The PZT enables energy harvesting and backscatter; the MCU handles communication and supports external sensing.

life-cycle. Apparently, continuous monitoring of the BiW is crucial for early detection of fatigue, corrosion, or impact damage.

For electric vehicles, the BiW is also designed to protect battery pack. Any damage to the battery could expose or short-circuit the battery cells, thus increasing the risk of fire. BiW monitoring allows for the early detection of damage to the battery pack, enabling timely preventative maintenance.

2.2 Principle of Acoustic Backscatter

Piezoelectric Transducer, commonly known as PZT, exhibits piezoelectric effect. That is it can convert mechanical energy into electrical energy and vice versa. When a sinusoidal voltage is applied to a PZT, it generates vibrations at the same frequency as the voltage. Conversely, when the transducer is exposed to mechanical vibrations, it produces a corresponding electrical voltage [19, 27, 34, 37]. This bidirectional functionality allows PZT to transmit and receive acoustic signals—such as vibrations of a vehicle BiW—similarly to how antennas transceive RF signals in the air.

PZT transducers have two fundamental states: *reflective* and *absorptive*. These states are pivotal in modulating backscattered signals, as illustrated in Fig. 2. Fig. 2(a) shows the reflective mode, where the short-circuited transducer reflects incoming carrier waves back. In contrast, Fig. 2(b) demonstrates the absorptive mode, where the open-circuited transducer absorbs the carrier wave and converts it into electrical energy, thus reducing the energy reflected.

Leveraging these properties, PZT transducers enable ultra-low power On-Off Keying (OOK). A typical backscattering tag features a PZT transducer with a transistor switch that shorts or opens the circuit. By toggling this switch, the tag modulates reflection strength for OOK with minimal power, which is ideal for devices with limited power budget.

■ **Discussion: What's the impact of external vibration on communication?** While the vehicle itself generates significant vibrations during operation, these vibrations have negligible impact on our communication. This is because their frequency is below 0.1 kHz [20, 21], while our communication operates at 90 kHz (the resonant frequency of the system). This frequency separation ensures robust communication even under typical vehicle operating conditions.

These self-vibrations can, however, serve as an auxiliary energy source. While our current design relies on reader-transmitted vibrations to ensure predictable energy delivery and communication, harvesting ambient vibrations remains a promising enhancement for future work.

3 ENERGY MANAGEMENT

This section introduces the hardware design of our backscatter tag, focusing on harvesting energy to activate tags.

3.1 Hardware Design Overview

Our design follows established principles from both acoustic [2, 3, 8, 19, 27] and RF backscatter [7, 35, 39, 44] literature. Fig. 3 illustrates the overview of tag hardware.

For the downlink (DL, from reader to tag), the tag features an envelope detector, paired with a comparator, essential for converting the DL carrier signal from the reader into binary voltage levels. These levels are then processed by the MCU for decoding. For the uplink (UL, from tag to reader), the tag MCU uses a transistor switch to control the PZT impedance, thus modulating the reflection strength. To monitor the BiW, each tag includes a strain measurement sensor module with a Wheatstone bridge that detects small differential voltages from the strain gauge. These voltages are then fed into the MCU's analog input, and are digitized by the onboard ADC. The digital value serves as the payload in the UL packets.

For energy harvesting, the tag uses a PZT-connected harvester to convert reader vibrations into electrical energy, storing it in a supercapacitor until sufficient voltage is reached for tag activation. In our case, the power budget for the tags is inherently limited. (1) **Power Supply.** The reader power is intentionally restricted to ensure safety within environments accessible to humans, i.e., BiW. Consequently, the applied power must remain within safe limits, reducing the energy available for backscatter tags. (2) **Propagation Attenuation.** With only one reader, some tags might be far away from the reader, and receive weakened energy due to the attenuation.

Under such conditions, the voltage of the supercapacitor may not reach the threshold required to activate the MCU.

3.2 Multi-Stage Voltage Amplifier

One straightforward solution is to use a multi-stage voltage multiplier (i.e., charging pump), as depicted in Fig. 4. The primary function of this multi-stage voltage multiplier is to amplify the voltage harvested from the carrier waves transmitted by the reader. Each stage of the voltage multiplier consists of a voltage doubler circuit, which effectively doubles the input voltage through a combination of diodes and capacitors. By cascading multiple stages, the circuit cumulatively increases the voltage level. Formally, for a N -stages voltage multiplier, the output voltage V_{DD} is given by $V_{DD} = 2N(V_P - V_{ON})$, where V_P is the peak voltage of the PZT output voltage, and V_{ON} the voltage drop introduced by a diode. In our design, we employ an 8-stage voltage multiplier to increase the voltage to a usable level.

Clearly, one critical issue is the voltage drop V_{ON} across the diodes. Traditional diodes have a forward voltage drop around 0.7 V, thus significantly reducing the efficiency of the voltage multiplier, especially when dealing with low input voltages. To mitigate this, we employ Schottky diodes CDBU0130L [11], featuring a lower voltage drop, potentially less than 0.15 V when the current is below 1 mA. However, as the number of stages increases, the overall charging efficiency tends to decrease due to cumulative voltage drops and power losses. We address this problem in the next section.

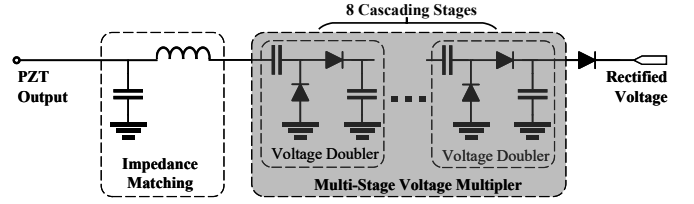


Figure 4: Multi-stage Voltage Multiplier. Cascaded voltage doublers amplify the rectified PZT output to power the tag's MCU.

On the other hand, to reduce voltage requirement, we select an ultra low-power MCU, MSP430G2 [26], and operate it at 1.8 V instead of the standard 3.3 V. This considerable reduction in operating voltage directly translates to lower voltage requirements for tag activation and operation. By reducing the voltage needed to power up the MCU, we ensure that tags can become operational more quickly and reliably.³

Also, we use a low-frequency clock operating at 12 kHz, which reduces MCU power consumption while maintaining a balance between processing capability and energy efficiency.

3.3 Low-Voltage Cutoff

In our network, detailed in Sec. 5, tags are always-on and receive packets reactively, necessitating efficient energy management. Our hardware design incorporates a low-voltage cutoff circuit to minimize energy leakage. This circuit intelligently controls power delivery from a supercapacitor to the MCU using two voltage thresholds: a high threshold (HTH) of 2.3V for activation and a low threshold (LTH) of 1.95V for deactivation, managing voltage fluctuations and leakage. The complete circuit design is detailed in Appendix A.

With this low-voltage cutoff mechanism, tags can efficiently resume operation from the LTH voltage rather than starting from zero, improving their energy efficiency and response time. Additionally, the harvested energy is stored in a supercapacitor. To minimize power leakage due to the capacitor's leakage current, we opt to use a 1 mF tantalum capacitor KEMET T491X108K006AT [12], with leakage current less than 0.01 CV (μ A) at rated voltage after 5 minutes.

4 INTERRUPT-DRIVEN DESIGN

Tags must maintain continuous operation under limited charging power to participate in the network. We first cover the channel modulation and packet structure for UL and DL in Sec. 4.1 and 4.2. These are essential in supporting the low-power design described in Sec. 4.3.

4.1 Channel Modulation

UL Modulation. We employ FM0 encoding for UL, a widely used scheme in backscatter systems. Specifically, The raw bit pairs 10 or 01 correspond to an FM0 bit 0, while the raw bit pairs 00 or 11 correspond to an FM0 bit 1.

³It is noteworthy that while adopting a 0.9 V MCU, such as the MSP430L092, is feasible, the primary challenge lies in the incompatibility with current commercial flash memory devices, which do not support 0.9 V operation.

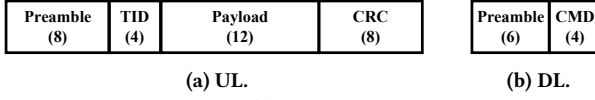


Figure 5: Packet Structures. (a) The UL packet carries tag ID, sensor data, and CRC. (b) The DL format is kept minimal to reduce tag power consumption, as each bit broadcast from the reader wakes up all tags via interrupts.

DL Modulation. We adopt pulse-interval encoding (PIE) in DL. Here, a PIE bit 0 is represented by two raw bits 10, while a PIE bit 1 is represented by three raw bits 110.

To mitigate the ring effect in DL, where the reader's PZT continues vibrating after voltage cutoff, we adopt the approach from [19]. We transmit high amplitudes at the BiW's resonant frequency and low amplitudes at non-resonant frequencies, reducing the vibration tail that interferes with demodulation. This 'FSK in OOK out' scheme also provides the benefit of reduced audible sound leakage compared to conventional amplitude-based transmission methods.

The default raw bit rates for UL and DL are 375 bps and 250 bps, respectively.

■ Discussion: Why Lower Throughput than Existing Works?

Existing backscatter systems [19, 27, 34] achieve over kbps throughput, while ours operates at lower rates due to three reasons: (1) Energy: Most works use high-power amplifiers for readers (450W for PAB [27], 1120W for EcoCapsule [19]) while we only use an 18W amplifier for safety (one exception is [34], but it operates on steel strips of few meters, versus our full BiW). (2) Hardware: We use a low-frequency 12 kHz MCU clock to minimize power consumption, reducing timer accuracy needed for precise packet decoding. (3) Design Choice: We deliberately set conservative rates, ensuring reliable communication to help network convergence.

4.2 Compact Packet Structure

The design of packet structures, especially for DL, must carefully balance multiple constraints. First, due to the inherently low throughput of acoustic backscatter in BiW, shorter packets are crucial to maintain acceptable latency and channel efficiency. Second, since tags are always-on and each DL bit triggers an interrupt that wakes up all tags, minimizing DL packet size is essential to reduce the tags' standby power consumption. With these considerations in mind, Fig. 5(a) and (b) show the packet design for UL and DL, respectively.

UL Packet: It is designed to transmit sensor readings from the tag to the reader. It comprises the following parts: (1) Preamble (8 bits): Marks the beginning of the packet. (2) TID (4 bits): Identifies the tag node (Tag ID); the 4-bit field supports up to 16 tags, but can be extended to support more if needed. (3) Payload (12 bits): Carries sensor data. (4) CRC (8 bits): Ensures data integrity via cyclic redundancy check.

DL Packet: The DL packet (i.e., beacon), transmitted by the reader, is designed to be short to reduce transmission time, thereby conserving channel resources, and to decrease the energy required for tag decoding. Therefore, it has only two short parts: (1) Preamble (6 bits): marking the arrival of the beacon. (2) CMD (4 bits): contains commands, including ACK/NACK (acknowledgment, Sec. 5.3), EMPTY (slot status, Sec. 5.5), RESET (state reset), and RESERVED (for future use).

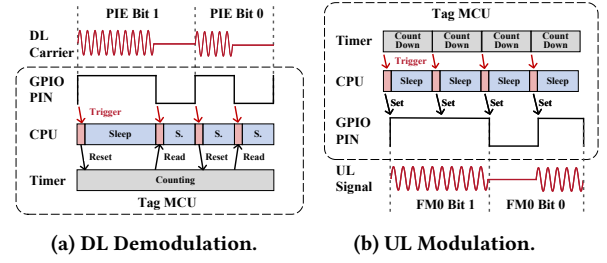


Figure 6: Trigger-based Communication Design. (a) DL demodulation uses edge-triggered interrupts to detect PIE symbols. (b) UL modulation uses timer interrupts to modulate FM0 symbols.

Notably, the DL packet does not include a CRC. This omission is not problematic because the primary function of the DL packet is to facilitate slot timing rather than to transmit data. Moreover, our network design incorporates tolerances for errors in DL reception, accommodating occasional inaccuracies. Meanwhile, the DL packet has an ACK without specifying Tag ID, introducing the need for tags to determine applicability of the ACK message independently (see Sec. 5).

■ Discussion: Why no Tag ID in DL packet? Including Tag ID in DL packet would necessitate additional CRC fields to ensure reliable ID transmission [17, 36], significantly increasing packet overhead. For example, with 4 bits for Tag ID and 8 bits for CRC, the DL packet length would increase by 12 bits, doubling our compact 10-bit design.

4.3 Interrupts and Handling

Building upon the modulation techniques and packet structures outlined, our tags employ an interrupt-driven software architecture to minimize energy consumption. This approach is critical for tags to remain operational within the distributed slot allocation protocol (see Sec. 5), given the constrained power budget of our tags.

Our design principle is that *all tag CPU behaviors are driven by interrupts*. This ensures that the CPU remains in a low-power state whenever possible. Generally, our tags are designed to perform three key tasks: (1) **DL Demodulation**, (2) **UL Modulation**, and (3) **Network Operation**. each task is dependent on specific interrupts: **DL Demodulation.** In this task, the CPU is driven by the interrupts triggered by transitions in binary signals that converted from the DL signal. Fig. 6(a) illustrates the concept of demodulating PIE symbol using interrupts. The PIE signal generated by the reader is initially processed through an envelope detector and comparator, converting the analog signal into digital bits [33]. This digital output is then directed to an input pin on the MCU. Transitions (both positive and negative edges) on this input pin trigger interrupts that wake the CPU from its low-power mode. Specifically:

- Upon detecting a positive edge, an interrupt is triggered, waking the CPU to reset the timer's counter to zero.
- When a negative edge is detected, another interrupt wakes the CPU to read the timer's counter value. This value corresponds to the duration of the pulse interval.

The CPU decodes the DL PIE bit based on the measured pulse interval. If the interval exceeds a predefined threshold, the CPU decodes it as bit 1; otherwise, it decodes the bit as 0.

UL Modulation. As shown in Fig. 6(b), the MCU is driven by interrupts generated by the MCU's timer. The timer is configured to generate interrupts at fixed intervals corresponding to the duration of one raw bit interval ($=1 / \text{bit rate}$). Each interrupt serves to wake the CPU from its low-power mode. Upon waking, the CPU sets the state of an output pin (ON/OFF) according to the encoded data packet. This operation is used to control the reflective or absorptive state of the tag's PZT by toggling a MOSFET switch. This in turn modulates the reflection signal strength.

Network Operation. Network tasks are managed through software interrupts, which are triggered by the successful decoding of reader beacons. This approach is rooted in our *reader-talks-first* communication principle, meaning that tags only act after receiving beacons from the reader. In detail, as each DL bit is decoded, the MCU matches the DL preamble to accurately detect the beacons. Once a complete beacon is decoded, it triggers the network state machine (to be introduced in Sec. 5), which then determines the tag's transmission behaviors. If a response is required, the tag will modulate and transmit the packet.

Our design keeps the CPU in low-power mode most of the time, only activating it when necessary. For reference, in active mode, the MCU MSP430 draws about 40-50 μA at 2 V. Our interrupt-driven architecture reduces the current to just 6.4 μA when receiving packets and 4.7 μA when transmitting—over 80% less than continuous active mode.

5 DISTRIBUTED SLOT ALLOCATION

We present a novel MAC protocol for ARACHNET that coordinates transmissions among multiple tags in a distributed manner.

5.1 Challenges

The coordination of backscatter tags in BiW presents unique challenges that make traditional MAC protocols unsuitable:

Diverse Transmission Rate: Different components of BiW require different monitoring frequencies—tags monitoring battery pack structure need frequent updates while those tracking structural aging can transmit less often. This necessity for heterogeneous transmission rates complicates collision avoidance.

High Collision Rate: The inherently low throughput of acoustic backscatter in BiW results in long packet durations (~ 200 ms for an UL packet). Additionally, our hardware design enables more tags to be activated within the BiW, creating a dense channel. The combination of lengthy packets and high tag density introduce frequent packet overlaps.

Limited Feedback Information: The limited power budget of tags makes it impractical for them to receive detailed DL feedback from the reader. Since the reader must frequently broadcast DL packets to coordinate tags, and each DL bit triggers tags to wake up for demodulation (see Sec. 4.3), larger DL packet sizes would significantly increase the tags' standby power consumption. Meanwhile, the available charging power only marginally exceeds the reception cost, the energy budget for sustained tag operation is limited. (see Sec. 6.2). This constraint necessitates compact DL packets, which provide only basic feedback without including tag IDs (see Sec. 4.2).

Table 1: Illustrative Slot Allocation for Four Tags with Different Transmission Periods.

Tag/Slot	0	1	2	3	4	5	6	7	Allocation
t_A	T		T		T		T		$p_A = 2, a_A = 0$
t_B		T				T			$p_B = 4, a_B = 1$
t_C								T	$p_C = 8, a_C = 7$
t_D				T					$p_D = 8, a_D = 3$

"T" marks the slots each tag transmits.

Therefore, this limitation makes coordinating multiple tags particularly challenging since the reader cannot directly address or control individual tags using traditional POLL or SELECT commands.

In summary, these constraints form a unique coordination problem: *In a high-collision network, how to orchestrate tags with diverse transmission rates using only limited feedback information?* We propose a distributed protocol to address these challenges. We begin with a vanilla design.

5.2 Vanilla Slot Allocation

This design orchestrates the tag transmissions by fully exploiting the predetermined transmission periods of different tags and by assigning dedicated initial slot offsets to them.

Permissible Transmission Period. Let us first define a discrete set of transmission periods that are powers of two, which can facilitates systematic and predictable slot allocation. Mathematically, the set of permissible transmission periods, \mathbf{P} , is defined as: $\mathbf{P} = \{2^k | k \in \mathbb{N}_0\}$, where k denotes non-negative integers. Each tag t_i in the network selects a period p_i from \mathbf{P} . The period p_i determines the frequency of transmissions for tag t_i and is selected based on its transmission requirements. Basically, the larger transmission period a tag selects, the lower transmission rate it can transmit. Note that the combined transmission rate of all tags, slot utilization U , should not exceed the channel capacity:

$$U = \sum_i \frac{1}{p_i} \leq 1. \quad (1)$$

Initial Slot Offset. Given the transmission periods of tags, it is feasible to orchestrate a non-overlapping transmission schedule by carefully determining the initial slot offset for each tag within its respective transmission period.

Let a_i denote the initial slot for tag t_i . This offset, an integer between 0 and $p_i - 1$, specifies the local slot offset for tag t_i to transmit packet within its period p_i . Each tag maintains a local counter s_i , incrementing by 1 upon receiving a Beacon from the reader. The tag transmits its packet if the remainder of s_i divided by p_i equals a_i , and enters a sleep state otherwise. Formally, the behavior of tag t_i is defined as follows:

$$\begin{cases} \text{Transmit,} & \text{if } s_i \pmod{p_i} = a_i \\ \text{Sleep,} & \text{if } s_i \pmod{p_i} \neq a_i \end{cases} \quad (2)$$

Illustrative Example. Consider a network comprising four tags: t_A , t_B , t_C , and t_D , with diverse transmission period requirements: $p_A = 2$, $p_B = 4$, $p_C = 8$, and $p_D = 8$. Table 1 illustrates the initial slot offsets assigned to these tags. This configuration ensures that during each of the slots from 0 to 7, only one tag transmits at a time,

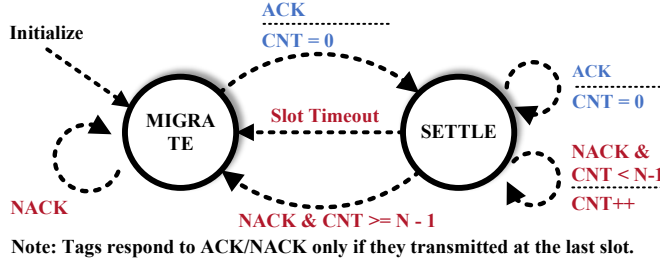


Figure 7: Tag State Machine. Tags begin in the MIGRATE state, randomly selecting a slot offset. Upon receiving an ACK, they enter the SETTLE state; upon receiving repeated NACKs, they revert to MIGRATE. This mechanism enables tags to adaptively find collision-free slots without centralized control.

thereby ensuring a non-overlapping transmission scheme. Clearly, this setup achieves the maximum slot utilization.

Comment. The effectiveness of the above method heavily relies on the assumption that *each tag maintaining a synchronized slot index*. This assumption is often invalid since inherent uncertainties in practice.

- **Beacon Loss:** Synchronization issues may arise if a tag fails to receive a Beacon packet from the reader, which is crucial for updating its local slot index s_i . This failure prevents the tag from correctly aligning its transmission schedule with others, leading to potential collisions.
- **Late-Arriving Tags:** Owing to variable charging durations, ranging from 4.5 s to 56.2 s in our case, tags within the network may be activated at different times. Thus, some tags may begin recording the slot indexes while others have already been recording their indexes for a while. This asynchrony among the tags can lead to incorrect slot counting, seriously disrupting the transmission schedule.

These problems necessitate a more robust and distributed slot allocation mechanism, which can accommodate these uncertainties and provide reliable performance in practice.

5.3 Dynamic Feedback System

Our proposed method allows tags within the network to autonomously discover a slot offset that minimizes collisions, even in the presence of synchronization issues and varying charging times. At the core of this approach is a dynamic feedback system in which each tag adjusts its slot offset based on ACK or NACK signals received from the reader. This trial-and-error process ensures that each tag can independently and iteratively find a conflict-free slot.

Tag State Machine. Each tag operates according to a state machine (see Fig. 7), with two states: MIGRATE and SETTLE.

- **MIGRATE:** Tags begin in this state, randomly selecting an initial slot offset and transmitting in their assigned slot. If a tag receives a NACK after transmission, it indicates a potential collision, prompting the tag to randomly select a new slot offset and remain in MIGRATE. Upon receiving an ACK, the tag transits to the SETTLE state.
- **SETTLE:** In this state, the tag has found a seemingly collision-free slot offset. It continues to transmit until it receives NACKs.

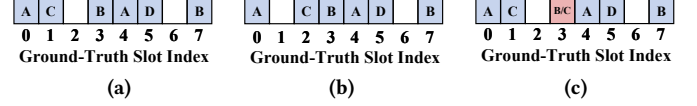


Figure 8: Impact of Beacon Loss on Tag Transmission Schedule. (a) Original transmission Pattern. (b) Unintended shift to unoccupied slots. (c) Collisions due to overlapping.

A single NACK does not immediately trigger a state change but increments a failure counter. If the count of consecutive failures reaches a threshold N ($=3$), the tag returns to MIGRATE to seek a new slot offset. Receiving an ACK resets the failure counter to 0.

Note that, in our design, the reader broadcasts ACK and NACK signals to all tags indiscriminately, without specifying individual tag IDs. Given this, it is crucial to ensure that only relevant tags—those that have just transmitted—respond to these signals. Therefore, tags are programmed to *respond to ACK/NACK only if they transmitted at the last slot*, and to disregard all other ACK/NACK signals.

Reader Feedback Mechanism. If the reader successfully receives an UL packet during a slot, it sets the ACK flag to true in its next DL beacon to signal the corresponding tag. However, due to the *capture effect*, the reader might still successfully decode one packet even if multiple tags are transmitting concurrently. To mitigate this, we detect clusters of backscattered signals in the IQ domain [23, 28, 29, 42]. If more than two clusters are identified, we infer that a collision has occurred. In this case, we set the ACK flag to false, even if the reader successfully decodes a UL packet.

Crucially, our state machine design allows flexibility to adapt to the network dynamics. In the subsequent sections, we detail how our approach inherently addresses issues, including **beacon loss** and **late-arriving tags**.

5.4 Beacon Loss Handling

Analysis. One advantage of our distributed slot allocation method is its tolerance to beacon loss. A missed beacon can lead to an outdated or incorrect slot count on a tag. For a centralized method introduced in Sec. 5.2, this desynchronization could disrupt the slot allocation strategy. However, in our distributed setup, each tag operates based on its local slot index. Even if a tag misses one or more packets and thereby receives incorrect slot index information, it does not rely on this absolute slot index to determine its actions. Instead, it depends on the feedback received post-transmission to determine the validity of its current slot offset. If receiving a NACK that indicates a collision possibly due to its desynchronized index, the tag can simply migrate to a new slot by changing its slot offset, thus self-correcting over time.

Missing a beacon packet by a tag effectively results in a shift of its transmission slot to the next slot: Let s denote the ground-truth index of the global slot, representing the correct slot index maintained by the reader. Initially, assume that the slot index s_i maintained by tag t_i is perfectly synchronized with the global index, i.e., $s_i = s$. However, upon missing a beacon, the local slot index s_i no longer increments, resulting in $s_i = s - 1$ for the subsequent slot. This desynchronization effectively shifts the transmission of tag t_i

by one slot. Mathematically, this shift can be expressed as follows:

$$\begin{aligned} s_i \bmod p_i = a_i &\Rightarrow (s - 1) \bmod p_i = a_i \\ &\Rightarrow s \bmod p_i = (a_i + 1) \bmod p_i. \end{aligned} \quad (3)$$

The original slot offset translates to $(a_i + 1) \bmod p_i$, indicating that the effective slot offset has been shifted by one. This unexpected shift can lead to two possible outcomes:

- **Unintended shift to unoccupied slots:** If the shift caused by the missed beacon moves the tag's transmission to unoccupied slots, the tag unexpectedly finds itself in a conflict-free schedule. In this case, it is safe for the tag to continue transmissions without requiring adjustments.
- **Collisions due to overlapping transmissions:** Conversely, the erroneous shift may position the tag into a slot already occupied by another tag, leading to collisions. These collisions typically result in *consecutive* NACKs from the reader until one tag transmits to MIGRATE.

Example. Fig. 8 provides an example. Suppose a network where slots 2 and 6 are unoccupied, while the remaining slots are occupied by tags A, B, C, and D. If tag C, originally scheduled to transmit in slot 1, misses a beacon and inadvertently shifts to slot 2, it moves to a free slot, thus avoiding potential collisions. However, if tag C misses another beacon and shifts from slot 2 to slot 3, it collides with tag B in slot 3.

Refinement. To mitigate the negative impacts of missed beacons, particularly consecutive collisions, we refine the protocol to include a simple detection mechanism for missed beacons. Each tag is programmed to expect periodic beacons from the reader. If a tag does not detect an expected beacon when its timer expires, it automatically re-enters MIGRATE, rather than waiting for multiple NACKs.

Consider the scenario depicted in Fig. 8(b). In the refined protocol, tag C re-enters MIGRATE before any potential collision with tag B occurs. Should a collision happen and tag C receives a NACK from the reader, it will immediately search for and adopt a new slot offset. Meanwhile, tag B remains in its original offset 3. This effectively minimizes disruptions by confining adjustments to the errant tag alone, enhancing the overall stability and efficiency of the network.

This approach minimizes disruptions by confining adjustments to the errant tag alone, thereby enhancing the overall stability and efficiency of the network.

5.5 Late-Arriving Tags Accommodation

Analysis. Our method inherently accommodates late-arriving tags, which might be activated after experiencing varied charging delays. As these tags will inherently start with an asynchronous slot index relative to already active tags, our distributed approach proves advantageous. Similar to the beacon loss scenario described in Sec. 5.4, the autonomous nature of our method allows these late-arriving tags to independently discover and adapt to the current transmission scheme by initially entering MIGRATE. Through a process of trial-and-error, and guided by feedback from the reader, these tags can dynamically integrate into the network, eliminating the need for additional synchronization.

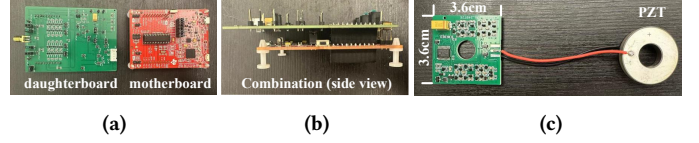


Figure 9: Tag Hardware Prototypes. (a) and (b) Modular prototype used for evaluation and power measurement. (c) Compact integrated tag with screw-mounting support for deployment.

Refinement. To help the new tags integrate into the network more smoothly, we propose a refinement to our design. Assuming that the period p_i of each tag is known to the reader, the reader can predict whether a current slot is occupied by exploiting the periodic transmission patterns of tags. We introduce a 1-bit **EMPTY** flag within the beacon broadcast by the reader. The **EMPTY** flag is defined as follows:

$$\text{EMPTY} = \prod_{i \in T_a} \mathbb{1}(\text{no packet received in slot } s - p_i), \quad (4)$$

where $\mathbb{1}()$ is a Boolean function, and T_a is the set of tags that appear. This flag indicates whether the current slot is empty. Newly arriving tags are allowed to transmit only if the **EMPTY** flag is true, ensuring they access only unused slots. This minimizes collisions and enables smooth integration into the current transmission pattern. Note that only newly arriving tags respond to the **EMPTY** flag.

5.6 Future Collision Avoidance

Early-arriving tags select transmission slots through a competitive process. Late-arriving tags, however, do not participate in this initial contention and may inadvertently disrupt the network. For instance, they might select seemingly free slots that are already occupied by earlier tags with longer transmission periods, leading to future collisions.

Example. Consider a scenario involving three tags: Tag A and B arrive early and both have a period of 4. They select offsets 2 and 3, respectively. Tag C arrives late with a transmission period of 2. As tag C begins to transmit in every alternate slot, it will always collide with either A or B. In this case, tag C faces future collisions.

Refinement. When a new tag is detected by the reader, the reader analyzes the periods of each tag and the current slot occupancy (All tags periods are known to the reader). If the reader finds that the existing allocation leaves no viable options for the new tag, the reader replies NACK to this new tag, even if its packet is received (e.g., Tag C). Meanwhile, the reader prioritizes selecting less crowded slots and issues successive NACKs to the tags occupying those slots (e.g., Tag A). This forces the partially settled tags to migrate. As a result, Tags A and C compete again for available slots. Since Tag C typically has a shorter period, it is more likely to settle first, while Tag A will eventually select a free slot.

5.7 Convergence Proof

We prove that our distributed slot allocation protocol converges to a collision-free state by modeling the system as an absorbing Markov chain. In this model, each state captures the slot offsets and transmission states (MIGRATE or SETTLE) of all tags.

We briefly sketch the key idea here for clarity. We model the slot allocation process as an absorbing Markov chain, where each state

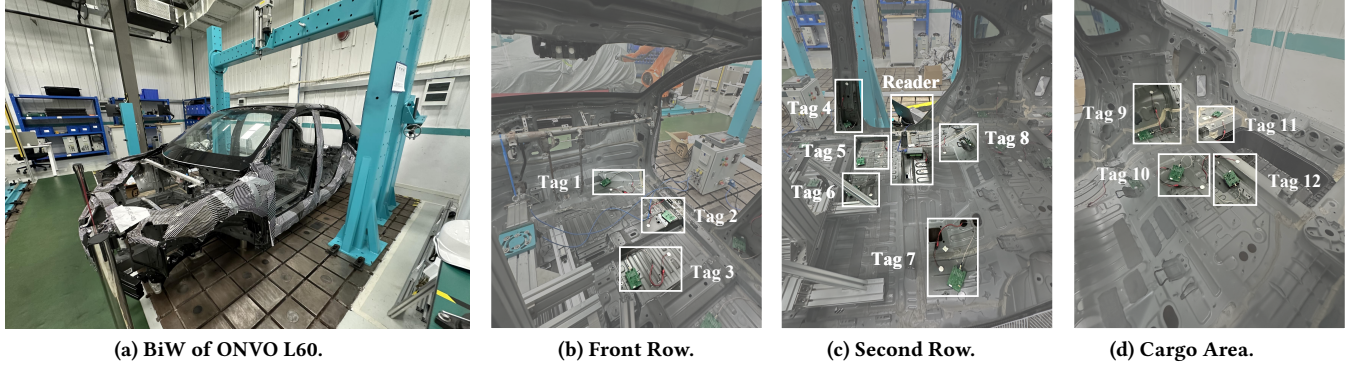


Figure 10: Deployment of ARACHNET. (a) Full view of the vehicle's BiW structure. (b) Tags 1–3 are attached near the front row. (c) Tags 4–8 and the reader are placed in the second row. (d) Tags 9–12 are deployed in the cargo area. All PZTs are directly mounted on metal surfaces using epoxy, forming a 12-tag network with one central reader.

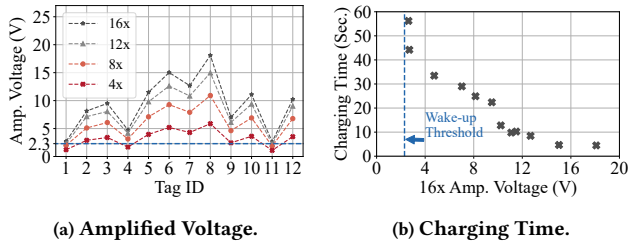


Figure 11: Amplified Voltage and Charging Time.

encodes the current configuration of tag offsets and their respective state machines. The absorbing states correspond to configurations in which all tags are in the SETTLE state and no collisions occur. Due to the randomized re-selection of slot offsets in the MIGRATE state, the system has a non-zero probability of transitioning from any transient state to an absorbing state. Since the overall state space is finite, the chain is guaranteed to reach an absorbing (i.e., collision-free) state with probability 1.

The detailed proof, including the formal construction of the Markov chain, state-transition definitions, and the complete convergence analysis, is provided in Appendix C.

6 EVALUATION

This study does not involve human participants, animal subjects, or sensitive data. Therefore, no ethical approval is required.

6.1 Implementation

We implement both the backscatter tag and reader. The tag can operate in a battery-free manner, while the reader is engineered for real-time performance, capable of transceiving the DL and UL packets in real time.

Backscatter Tag. The main designs of the tags and their specific components are detailed in Sec. 3. As illustrated in Fig. 9(a) and (b), the hardware prototype for the tag adopts a modular approach, dividing the system into two parts: a daughterboard and a motherboard. This modular setup facilitates easier debugging and precise measurement of power consumption. The daughterboard is a custom-designed circuit board developed specifically for this project, featuring a two-layer PCB manufactured by JLCPCB. The

motherboard is a MSP-EXP430G2ET Development Kit, containing an ultra-low-power MCU MSP430G2553. The daughterboard interfaces with the motherboard via two 10-pin connectors, physically mounted atop it. The daughterboard serves several functions: supplying 1.95 V–2.3 V power to the motherboard, converting the DL carrier signal to TTL voltage, and receiving UL bits to control the PZT impedance.

To achieve a more compact tag design, we also integrate the motherboard and the daughterboard, shown in Fig. 9(c). The bill of materials (BOM) cost for this compact tag is \$6.25.

Backscatter Reader. The reader employs two PZTs: TX PZT for transmitting and RX PZT for receiving. The reader uses a USB-connected data acquisition device, ART USB3136A, paired with a laptop, HP Zbook. For the transmission link, the laptop configures the data acquisition device to transmit a PWM signal at 90 kHz (the resonant frequency of the system), amplified by an external 18 W amplifier [15] to drive the TX PZT. The output peak voltage of the amplifier is 36 V, i.e., 72 V peak-to-peak voltage. The RX PZT is connected to the data acquisition device's analog input, sampling signals at 500 kHz. The samples are streamed to the laptop via USB.

The reader's software, developed in C++, handles DL transmission, UL reception, and network protocols in real-time. For DL transmission, the software can dynamically pause and resume DL transmissions to modulate PIE symbols through USB commands. For UL reception, the software includes processing blocks for down conversion, frequency offset calibration, Schmitt triggering, filtering, decimation, and packet decoding. Each two adjacent blocks share a buffer with a back-pressure mechanism to manage data flow. The packet passing CRC will be processed by the upper layers.

Deployment. Our experiments are conducted in the body stiffness and strength lab of NIO. Fig. 10(a) shows the BiW of an SUV (ONVO L60) used in our experiment. The vehicle measures about 4.8 m in length, and 1.9 m in width, slightly larger than the Tesla Model Y. As illustrated in Fig. 10(b)–(d), 12 tags are deployed within the BiW across three areas: the front row (Tags 1–3), the second row (Tags 4–8), and the cargo area (Tags 9–12). The PZT is attached to the BiW by using epoxy resin adhesive (i.e., AB glue). The reader is centrally placed in the second row, above the battery pack. In real-world deployments, sealing, insulation, and adhesive reinforcement would

Table 2: Tag Power Consumption in Different Modes.

Mode	Current (μA)		Voltage (V)	Power (μW)
	MCU	Total		
RX	6.4	12.4	2.0	24.8
TX	4.7	25.5	2.0	51.0
IDLE	0.6	3.8	2.0	7.6

be added to improve long-term robustness against heat, humidity, and material aging.

6.2 Energy Efficiency

Here, we validate that the tags can not only be activated, but also can maintain a long-term operation.

Amplified Voltage. As discussed in Sec. 3.2, a multi-stage voltage multiplier is employed to amplify the voltage from the PZT transducers. Fig. 11(a) shows the amplified voltages for tags at stage numbers 2, 4, 6, and 8 (i.e., amplified ratio 4x, 8x, 12x, and 16x). While the amplified voltage for each tag increases with the stage numbers, the rise is not proportional to the stage number since voltage drops across diodes.

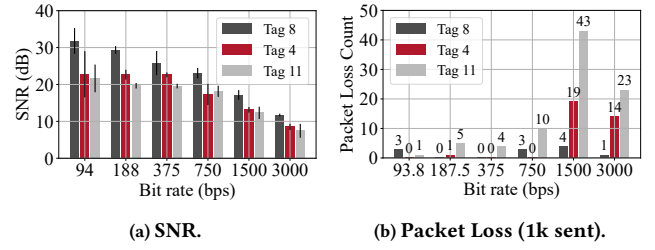
It is instructive to examine how the BiW structure affects the harvested voltage by looking at Tags 4 and 11. Tag 4, located at a turning face of the BiW structure, achieves 4.74V at 16x amplification, while Tag 11, located even further in the cargo area, reaches 2.70V. These lower voltages result from signal attenuation. For Tag 4, this is due to the geometric transition at the perpendicular junction, while for Tag 11, it's due to the long propagation distance through multiple structural elements. Nevertheless, at a stage number of 8, the amplified voltage for all 12 tags exceeds the activation threshold of 2.3 V, indicating that all tags can be activated. Thus, we set the stage number to 8 by default.

Charging Time and Net Charging Power. Next, we examine the time required for the reader to charge the tag to activation mode. To ensure realism, this experiment involves both the low-voltage cutoff circuit and the DL demodulation circuit. Consequently, the charging time is defined as the duration needed for the tag's super capacitor to charge from 0 V to the cutoff circuit's high voltage output of 2.3V. Since charging time is directly related to voltage, we plot it as a function of the amplified voltage (when the stage number is 8) in Fig. 11(b). As observed, the charging time ranges from 4.5 s to 56.2 s,⁴ corresponding to net charging powers of 587.8 μW and 47.1 μW (the supercapacitor is 1 mF), respectively, while accounting for power leakage from the low-voltage cutoff circuit and the DL demodulation circuit.

Power Consumption. In our protocol, the tags operate in three modes: receiving beacons (RX), backscattering packets (TX), and staying deep sleep (IDLE), whose power consumptions are detailed in Table 2. We also independently measured the current drawn by the MCU.

Notably, the RX and IDLE modes exhibit low power consumptions of 24.8 μW and 7.6 μW , respectively. The TX mode, however,

⁴These represent the time required to charge from scratch. Thanks to the low-voltage cutoff design, the tags typically resume charging from LTH rather than 0 V, enabling re-activation within 10 s.

**Figure 12: Key Performance of Uplink.**

consumes 51.0 μW , primarily due to the frequent toggling of the MOSFET by the MCU, which draws notable power through the MCU pin (i.e., MOSFET gate). It is important to highlight that the low power consumption of the MCU is a direct benefit of our software architecture, where all CPU behaviors are driven by interrupts, allowing the CPU to enter deep sleep mode (i.e., MSP430 LPM3) when not actively processing tasks. This design keeps the overall power usage within a manageable range, even in the more demanding TX mode, compared to the minimal charging power of 47.1 μW , and thus supports continuous operation of the tag in a duty-cycled mode.

6.3 Communication Efficiency

Next, we study the key performance of UL and DL.

Uplink. Fig. 12(a) illustrates the signal-to-noise ratio (SNR) for the three tags (Tags 8, 4, and 11 in Fig. 10) as a function of bit rate. The SNR is calculated by dividing the backscattering frequency power by the surrounding frequency power via Power Spectral Density (PSD). The nominal bit rates are 93.75 bps, 187.5 bps, 375 bps, 750 bps, 1,500 bps, and 3,000 bps, achieved by configuring the MCU clock division factors to 128, 64, 32, 16, 8, and 4, respectively, at a clock frequency of 12 kHz. The results indicate a decrease in SNR as the bit rate increases. This trend is attributed to the wider bandwidth over which the power is spread at higher bit rates. Among the tags, Tag 8 consistently shows the highest SNR across all bit rates, benefiting from its proximity to the reader, with an SNR exceeding 11.7 dB at 3,000 bps. Although Tag 4 is positioned at a structural face different from the reader, it maintains reasonable SNR levels. Similarly, Tag 11 is located in the distant cargo area, it still achieves an acceptable SNR of about 18.1 dB when the bit rate is no more than 750 bps.

Fig. 12(b) further presents the packet loss counts for the same tags across different bit rates. For each bit rate setting, we trigger each tag to send 1,000 packets to the reader. The packet loss count is calculated by subtracting the number of packets received from 1,000. The results indicate a slight increase in packet loss at higher bit rates. However, the packet error ratio remains below 0.5% for all settings.

In our design, the raw bit rate for UL is set to 375 bps by default, which provides a promising reliability.

Downlink. Similar to our uplink tests, the reader is configured to send 1,000 packets to tags, and we calculate the packet loss count for the three tags. As depicted in Fig. 13(a), the packet loss increases with the bit rate.

Notably, at bit rates of 1,000 and 2,000 bps, the packet loss increases significantly. This surge in packet loss is not attributed to

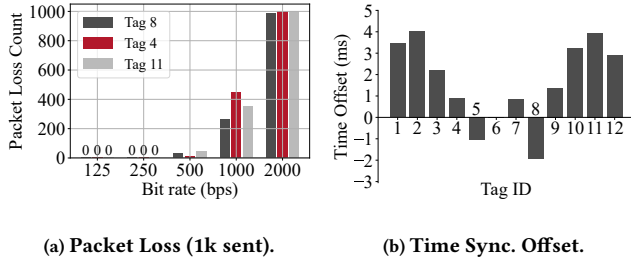


Figure 13: Key Performance of Downlink.

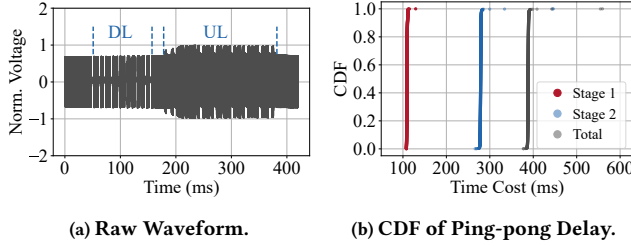


Figure 14: Ping-Pong Test.

signal quality issues, but rather to limitations of the hardware of both tag and reader. On one hand, the MCU operates at a clock frequency of only 12 kHz, and because it is powered by a varying voltage from the supercapacitor rather than a stable one from an LDO regulator, the timer lacks precision. This imprecision is critical given that the downlink modulation scheme uses PIE, which requires accurate timer-based pulse interval calculations at the high bit rate. On the other hand, the reader modulates PIE using software, which introduces about 0.1-0.3 ms time offset to each PIE symbol.

The DL beacon plays a crucial role in our distributed protocol, serving to synchronize tags. Consequently, the raw DL bit rate is set by default to 250 bps. Although this setting may reduce throughput, it enhances reliability.

Additionally, we measure the synchronization offset among the tags as shown in Fig. 13(b). In our distributed protocol, the reader broadcasts a beacon marking the end of the last slot and the beginning of the current one. We compare the time each tag receives this beacon relative to a reference tag, Tag 6. The results show that all tags can be well synchronized, with time offsets less than 5.0 ms. This level of synchronization is crucial for the operation of our protocol, ensuring minimal slot misalignment among the tags.

■ **Discussion: Is throughput acceptable for BiW monitoring?** The throughput supports second- to minute-level monitoring, sufficient for various applications. Minute-level intervals can capture gradual structural degradation, while second-level monitoring effectively detects early signs of thermal runaway or fire development of battery, which typically evolve over periods ranging from 30 seconds to hours, except in catastrophic events like severe crashes [40]. To further improve throughput, future extensions may explore higher-order modulation schemes [34], FDMA-based techniques [27], or spatial multiplexing via multiple readers distributed across the BiW.

Table 3: Tag Transmission Pattern.

TX Period	c1	c2	c3	c4	c5	c6	c7	c8	c9
4 slots	0	0	1	0	1	0	1	1	2
8 slots	0	0	2	6	3	1	1	1	0
16 slots	0	12	2	0	4	10	4	6	4
32 slots	12	0	7	6	4	0	4	0	0
Tag#	12	12	12	12	12	11*	10 [†]	8 [‡]	6 [§]
Slot Util.	0.38	0.75	0.84	0.94	1.00	0.75	0.75	0.75	0.75

*Excludes Tag 7; [†]Excludes Tags 4 and 7; [‡]Excludes Tags 1, 4, 7, and 9;

[§]Excludes Tags 1, 3, 4, 7, 9, and 11;

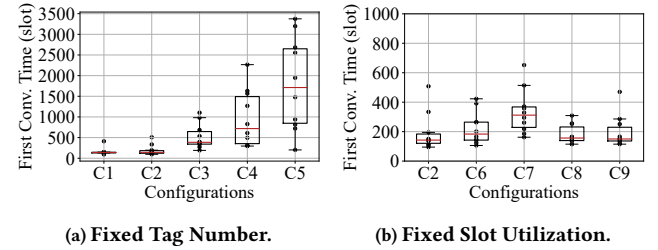


Figure 15: First Convergence Time.

6.4 Network Efficiency

Ping-Pong Latency. The ping-pong operation is the cornerstone of network. We measure the latency from when the reader transmits a DL packet to the reception of the decoded UL packet. Fig. 14(a) displays the raw waveform captured by the reader RX PZT, which clearly illustrates the tag behaviours: After receiving a DL beacon, the tag politely waits for 20 ms, and then replies a UL packet.

Fig. 14(b) shows the cumulative distribution function (CDF) of the delays, divided into two stages: Stage 1: Time for DL packet transmission. Stage 2: Time from DL transmission end to UL packet decoding, with 99% of Stage 2 delays under 281.9 ms. Considering the tag's wait and the UL duration, our reader software adds only about 58.9 ms of delay. This demonstrates the system's real-time capability, as the software delay is less than 30% of the UL packet duration. By default, the slot duration is empirically set to 1 s.

First Convergence Time. Next, we evaluate how quickly our distributed slot allocation protocol enables tags deployed across the BiW to settle into non-overlapping slots. Table 3 presents 9 tag transmission patterns used in this experiment. These patterns use four different transmission periods: 4, 8, 16, and 32. These patterns fall into two categories: (1) Fixed Number of Tags. This category includes patterns c1, c2, c3, c4, and c5, where the number of tags remains constant at 12, but the slot utilization varies. (2) Fixed Slot Utilization. This category includes patterns c2, c6, c7, c8, and c9, where the slot utilization U is fixed at 0.75, but with varying periods.

Fig. 15(a) and (b) depict the first convergence times for these patterns. The first convergence time is defined as the number of slots required for the reader to detect 32 consecutive non-collision slots following the transmission of a RESET packet. Fig. 15(a) illustrates that the first convergence time increases with slot utilization. For instance, the median convergence time increases significantly from

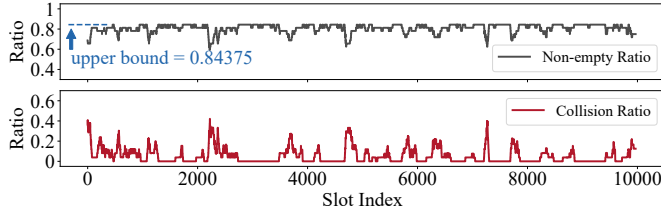


Figure 16: Non-empty Ratio and Collision Ratio.

139 to 1,712 as the slot utilization rises from 0.38 (c1) to 1.0 (c5). Fig. 15(b) shows less variance in convergence times for c2, suggesting that slot utilization is the predominant factor.

Long-Running Slot Statistics. Now, we explore the long-term performance of our distributed slot allocation protocol by analyzing two key metrics: the non-empty ratio and the collision ratio. The non-empty ratio is defined as the proportion of slots that contain at least one tag transmission (which may include collisions) within a recent window of slots, relative to the window size. The collision ratio is defined as the proportion of slots where more than one tag transmits a packet, also relative to the window size. Here, the window size is 32 slots, and the transmission pattern is c3 in Table 3.

As depicted in Fig. 16, both the non-empty ratio and the collision ratio exhibit notable fluctuations. The primary cause of these fluctuations is DL packet loss, which leads to slot asynchronization of tags. This disruption can trigger a new round of distributed slot allocation, resulting in disturbances across some tags. Meanwhile, minor fluctuations in the non-empty ratio are occasionally noted, hovering around the theoretical upper bound of 0.84375 (see c3 in Table 3). These variations are typically caused by failures in UL packet decoding, affecting only the non-empty ratio without further repercussions.

Nevertheless, our protocol allows tags to quickly adapt to disruptions and restore settlement. The average non-empty ratio is 81.2% and the average collision ratio is only 0.056, which demonstrates robustness in maintaining efficient slot management over long running. To provide conceptual context, an ALOHA baseline is presented in Appendix B.

6.5 Case Study: Metal Strain Measurement

Our tag integrates a strain sensor module, whose main design is similar to [25], but adapts the supply voltage to 1.8 V. To avoid causing damage to the existing BiW, we attach three strain gauges (Tags A, B, and C) to a metal surface instead, as shown in Fig. 17(a). The resistance of these gauges changes when the metal bending. Using a full Wheatstone bridge, we measure the differential voltage changes, which are digitized by the MCU's ADC and included in the UL packet. Recognizing that the ADC and the pre-amplifier are power-hungry (around 1 mW in our case), the tag samples at most one sample per slot for energy saving. We displace one end of the metal from -10 cm to 10 cm. Fig. 17(b) shows the voltage results from the three tags, demonstrating a clear correlation between voltage and displacement. This confirms the system's potential for structural health monitoring.

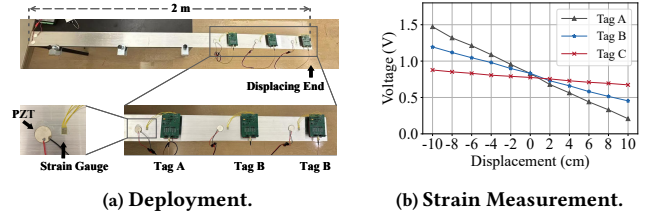


Figure 17: Case Study: Monitoring Metal Displacement.

7 RELATED WORK

Structural Health Monitoring. Structural health monitoring (SHM) technology, enabling continuous assessment of structural integrity through integrated sensor networks, has gained attention in aviation and maritime industries [6, 18, 30, 32]. For instance, products like the Fiber Bragg Grating sensor measure acceleration, allowing real-time monitoring of ship conditions [38]. In aviation, studies employ distributed fiber optic sensors based on optical time-domain reflectometry to detect strain and temperature changes in aircraft structures [22]. Traditional SHM technologies require sensors to gather structural state information. While the cost of deploying these sensors, specialized cables, and connectors is acceptable in the high-cost manufacturing contexts of aircraft and ships, it becomes significant for automobiles with lower manufacturing costs. Thus, traditional SHM methods are often deemed impractical for automobiles.

Backscatter Communication. Compared to traditional SHM technologies, backscatter communication doesn't require power sources, relying instead on a wireless, passive system to obtain structural state information, making it more practical for automotive SHM. Most past research on backscatter communication focus on RF backscatter in the air [35, 39, 44]. However, RF signal paths can be severely impeded in media like concrete and underwater environments. Consequently, studies explore alternative methods for these environments [2, 8, 19, 27]: In concrete, research demonstrates the use of embedded self-sensing nodes for SHM based on backscatter communication [19]. In underwater environments, acoustic communication is well studied [1, 14, 27], and recently an analytical model is proposed to analyze underwater backscatter channels comprehensively [2].

8 CONCLUSION

We presented ARACHNET, an acoustic backscatter network for monitoring the vehicle BiW. It utilizes battery-free tags that harvest energy from BiW vibrations and communicate efficiently via backscatter techniques. Evaluation results show the system's low power consumption and effective network performance, making ARACHNET a practical solution for continuous vehicular structural health monitoring.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and our shepherd for their valuable feedback in shaping the final version of this paper. We also thank Tao Xi (NIO) and Ruiguang Zhong (NIO) for their insightful advice. This work is partially supported by the National Science Fund of China under grant No. 62425207, and by the China Postdoctoral Science Foundation under grant No. GZB20240356.

REFERENCES

- [1] Sayed Saad Afzal, Waleed Akbar, Osmy Rodriguez, Mario Doumet, Unsoo Ha, Reza Ghaffarivardavagh, and Fadel Adib. 2022. Battery-free wireless imaging of underwater environments. *Nature communications* 13, 1 (2022), 5546.
- [2] Waleed Akbar, Ahmed Allam, and Fadel Adib. 2023. The underwater backscatter channel: Theory, link budget, and experimental validation. In *Proceedings of the ACM MobiCom*.
- [3] Ahmed Allam. 2021. Acoustic Power Transfer Leveraging Piezoelectricity and Metamaterials. (2021).
- [4] Autoevolution. 2023. Tesla Explains How the Single-Piece Megacasting for the Model Y Protects You in a Crash. <https://www.autoevolution.com/news/tesla-explains-how-the-single-piece-megacasting-in-the-model-y-protects-you-in-a-crash-212772.html>. (2023). Accessed: 2024-12-01.
- [5] Fabrice Auzanneau. 2013. Wire troubleshooting and diagnosis: Review and perspectives. *Progress In Electromagnetics Research B* 49 (2013), 253–279.
- [6] Carlos G Berrocal, Ignasi Fernandez, and Rasmus Rempling. 2021. Crack monitoring in reinforced concrete beams by distributed optical fiber sensors. *Structure and Infrastructure Engineering* 17, 1 (2021), 124–139.
- [7] Michael Buettner, Richa Prasad, Alanson Sample, Daniel Yeager, Ben Greenstein, Joshua R Smith, and David Wetherall. 2008. RFID sensor networks with the Intel WISP. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*. 393–394.
- [8] Jean-François Chaix, Vincent Garnier, and Gilles Corneloup. 2003. Concrete damage evolution analysis by backscattered ultrasonic waves. *Ndt & E International* 36, 7 (2003), 461–469.
- [9] Satyan Chandra, In Jae Chung, Adnan Esmail, Matthew Blum, and Rishabh Bhandari. 2022. Wiring system architecture. (2022). US Patent 11,260,809.
- [10] K Diamanti and C Soutis. 2010. Structural health monitoring techniques for aircraft composite structures. *Progress in Aerospace Sciences* 46, 8 (2010), 342–352.
- [11] Digikey. 2024. CDBU0130L-HF. <https://www.digikey.com/en/products/detail/omchip-technology/CDBU0130L-HF/3308503>. (2024). Accessed: 2024-12-01.
- [12] Digikey. 2024. T491 Industrial Grade MnO2. <https://www.digikey.com/en/products/detail/kemet/T491X108K006AT/1681738>. (2024). Accessed: 2024-12-01.
- [13] Ducker. 2022. 2022 Mega-Casting Trends for Automotive Manufacturers. <https://www.duckercarlisle.com/wp-content/uploads/2022/11/Mega-Casting-Whitepaper-May-2022-1.pdf>. (2022). Accessed: 2024-12-01.
- [14] Aline Eid, Jack Rademacher, Waleed Akbar, Purui Wang, Ahmed Allam, and Fadel Adib. 2023. Enabling long-range underwater backscatter via van Atta acoustic networks. In *Proceedings of the ACM SIGCOMM*.
- [15] ElecDemo. 2022. High voltage adjustable pulse drive module Piezoelectric ceramic PZT drive controller. <https://www.aliexpress.us/item/3256804731056919.html>. (2022). Accessed: 2024-12-01.
- [16] electrek. 2019. Tesla reveals revolutionary new wiring architecture to help robots build upcoming cars like Model Y. <https://electrek.co/2019/07/22/tesla-revolutionary-wiring-architecture-robots-model-y/>. (2019). Accessed: 2024-12-01.
- [17] Behrouz A Forouzan. 2007. *Data communications and networking*. Huga Media.
- [18] Iker García, Joseba Zubia, Gaizka Durana, Gotzon Aldabaldetrek, María Asunción Illarramendi, and Joel Villatoro. 2015. Optical fiber sensors for aircraft structural health monitoring. *Sensors* 15, 7 (2015), 15494–15519.
- [19] Zheng Gong, Lubing Han, Zhenlin An, Lei Yang, Siqi Ding, and Yu Xiang. 2022. Empowering smart buildings with self-sensing concrete for structural health monitoring. In *Proceedings of the ACM SIGCOMM*. 560–575.
- [20] MJ Griffin. 1978. The evaluation of vehicle vibration and seats. *Applied ergonomics* 9, 1 (1978), 15–21.
- [21] Michael J Griffin. 2007. Discomfort from feeling vehicle vibration. *Vehicle System Dynamics* 45, 7-8 (2007), 679–698.
- [22] Honglei Guo, Gaozhi Xiao, Nezih Mrad, and Jianping Yao. 2011. Fiber optic sensors for structural health monitoring of air platforms. *Sensors* 11, 4 (2011), 3687–3705.
- [23] Pan Hu, Pengyu Zhang, and Deepak Ganesan. 2015. Laissez-faire: Fully asymmetric backscatter communication. In *Proceedings of the ACM SIGCOMM*.
- [24] IEC. 2024. Vehicles, boats and internal combustion engines - Radio disturbance characteristics - Limits and methods of measurement for the protection of on-board receivers. <https://webstore.iec.ch/en/publication/64645/>. (2024). Accessed: 2024-12-01.
- [25] Texas Instruments. 2022. Single-Supply Strain Gauge Bridge Amplifier Circuit. <https://www.ti.com/lit/an/sboa247a/sboa247a.pdf>. (2022). Accessed: 2024-12-01.
- [26] Texas Instruments. 2024. MSP430G2553. <https://www.ti.com/product/MSP430G2553>. (2024). Accessed: 2024-12-01.
- [27] Junsu Jang and Fadel Adib. 2019. Underwater backscatter networking. In *Proceedings of the ACM SIGCOMM*. 187–199.
- [28] Meng Jin, Yuan He, Xin Meng, Dingyi Fang, and Xiaojiang Chen. 2018. Parallel backscatter in the wild: When burstiness and randomness play with you. In *Proceedings of the ACM MobiCom*.
- [29] Meng Jin, Yuan He, Xin Meng, Yilun Zheng, Dingyi Fang, and Xiaojiang Chen. 2017. Flitracer: Practical parallel decoding for backscatter communication. In *Proceedings of the ACM MobiCom*.
- [30] Rakesh Katam, Venkata Dilip Kumar Pasupuleti, and Prafulla Kalapatapu. 2023. A review on structural health monitoring: past to present. *Innovative Infrastructure Solutions* 8, 9 (2023), 248.
- [31] Adnan Kefal, Olgun Hizir, and Erkan Oterkus. 2015. A smart system to determine sensor locations for structural health monitoring of ship structures. In *Proceedings of the 9th international workshop on ship and marine hydrodynamics, Glasgow, UK*. 26–28.
- [32] Christoph Kralovec and Martin Schagerl. 2020. Review of structural health monitoring methods regarding a multi-sensor approach for damage assessment of metal and composite structures. *Sensors* 20, 3 (2020), 826.
- [33] Vincent Liu, Aaron Parks, Vamsi Talla, Shyamnath Gollakota, David Wetherall, and Joshua R Smith. 2013. Ambient backscatter: Wireless communication out of thin air. In *Proceedings of the ACM SIGCOMM*.
- [34] Peter Oppermann and Christian Renner. 2022. Higher-order modulation for acoustic backscatter communication in metals. In *Proceedings of the ACM SIGCOMM*.
- [35] Jiajue Ou, Mo Li, and Yuanqing Zheng. 2015. Come and be served: Parallel decoding for COTS RFID tags. In *Proceedings of the ACM MobiCom*.
- [36] Al Petrick et al. 2005. *The-IEEE 802.11 Handbook*. IEEE New York.
- [37] Charles H Sherman and John L Butler. 2007. *Transducers and arrays for underwater sound*. Vol. 4. Springer.
- [38] Smartec. 2024. MUST FBG Mono-Axial Accelerometer. <https://smartec.ch/en/product/must-fbg-mono-axial-accelerator-2/>. (2024). Accessed: 2024-12-01.
- [39] Joshua R Smith. 2013. *Wirelessly powered sensor networks and computational RFID*. Springer Science & Business Media.
- [40] Peiyi Sun, Roeland Bisschop, Huichang Niu, and Xinyan Huang. 2020. A review of battery fires in electric vehicles. *Fire technology* 56, 4 (2020), 1361–1410.
- [41] Teslarati. 2020. Tesla Model Y wiring efficiencies questioned in latest Munro teardown. <https://www.teslarati.com/tesla-model-y-shorter-wiring-efficiency-teardown-review/>. (2020). Accessed: 2024-12-01.
- [42] Jue Wang, Haitham Hassanieh, Dina Katabi, and Piotr Indyk. 2012. Efficient and reliable low-power backscatter networks. In *Proceedings of the ACM SIGCOMM*.
- [43] Austin Weber. 2024. Tesla's New Cybertruck Features Modular Wiring. <https://www.assemblymag.com/articles/98504-teslas-new-cybertruck-feature-s-modular-wiring>. (2024). Accessed: 2025-07-20.
- [44] Lei Yang, Yekui Chen, Xiang-Yang Li, Chaowei Xiao, Mo Li, and Yunhao Liu. 2014. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. In *Proceedings of the ACM MobiCom*.

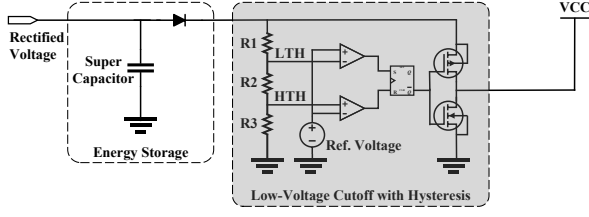


Figure 18: Low-voltage Cutoff Design with Hysteresis.

APPENDIX

Appendices are supporting material that has not been peer-reviewed.

A LOW-VOLTAGE CUTOFF CIRCUIT

Fig. 18 illustrates the low-voltage cutoff circuit design that manages power delivery between the supercapacitor and MCU. The circuit employs a comparator to monitor the supercapacitor's voltage, implementing an adaptive switching mechanism. When the voltage exceeds the high threshold (HTH), power transfers to the MCU; when it falls below the low threshold (LTH), the circuit disconnects to prevent excessive discharge.

The threshold voltages are determined by resistors R_1 , R_2 , and R_3 according to:

$$\begin{cases} V_{HTH} = V_{REF} \cdot \frac{R_1 + R_2 + R_3}{R_3} \\ V_{LTH} = V_{REF} \cdot \frac{R_1 + R_2 + R_3}{R_2 + R_3} \end{cases}, \quad (5)$$

where $V_{REF} = 1.24V$. The resistor values are set to standard values of $R_1 = 680 \text{ k}\Omega$, $R_2 = 180 \text{ k}\Omega$, and $R_3 = 1 \text{ M}\Omega$. This configuration establishes threshold voltages of $V_{HTH} = 2.3 \text{ V}$ and $V_{LTH} = 1.95 \text{ V}$ while maintaining circuit leakage below $1 \mu A$.

B ALOHA SIMULATION BASELINE

To conceptually evaluate the limitations of contention-based medium access, we simulate a pure ALOHA protocol under the same hardware and deployment constraints as our system. Each of the 12 battery-free tags transmits immediately after harvesting enough energy to reach the high threshold voltage (HTH) of 2.3 V. Due to the low-voltage cutoff mechanism, recharging resumes from 1.95 V and requires only 15.2% of the full charging duration. We adopt measured charging times ranging from 4.5 s to 56.2 s (see Sec. 6.2) and introduce 2% Gaussian noise to model real-world variability. During each 200 ms packet transmission, charging is paused. Over a 10,000-second simulation, we record all transmission events and check for overlaps to compute collision statistics on a per-tag basis.

As shown in Fig. 19, ALOHA results in imbalanced channel access and frequent collisions, with only 34.0% of the total UL transmissions being collision-free. Per-tag success rates range from 28.4% to 37.3%, with outcomes slightly correlated with charging time. Fast-charging tags (e.g., Tag 8, 4.5 s) transmit over 11,000 times but still collide in more than 60% of attempts. Meanwhile, slow-charging tags (e.g., Tag 11, 56.2 s) transmit less frequently but experience collision rates exceeding 70%. These highlight not only ALOHA's inefficiency, but also its inability to provide fair channel access across asymmetrically powered tags, reinforcing the need for coordinated protocols like our distributed slot scheduling.

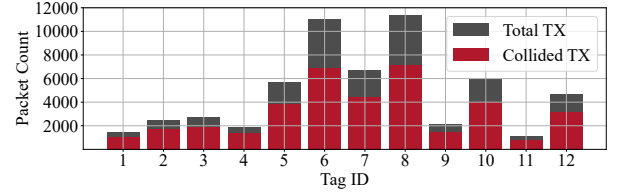


Figure 19: Per-Tag Transmission and Collision Statistics in ALOHA Simulation

C CONVERGENCE PROOF FOR DISTRIBUTED SLOT ALLOCATION

We now provide a proof of the convergence of the distributed slot allocation protocol.

C.1 Markov Chain Modeling

We define the state of tag i as

$$x_i = (z_i, a_i, c_i),$$

where $z_i \in \{\text{MIGRATE} \mid \text{SETTLE}\}$ denotes the primary state of tag i , a_i is the slot offset, and c_i is the number of successive NACKs counted. The state of the network consists of all states of tags

$$X = \{x_0, x_1, x_2, \dots, x_{T-1}\}$$

where T is the number of tags.

The system evolves through state transitions that occur at each slot boundary. For any tag i who transmit in this slot, its state transition is defined as follows. When tag i in the MIGRATE state:

- Upon receiving ACK:

$$(\text{MIGRATE}, a_i, c_i) \rightarrow (\text{SETTLE}, a_i, 0)$$

- Upon receiving NACK:

$$(\text{MIGRATE}, a_i, c_i) \rightarrow (\text{MIGRATE}, a'_i, c_i),$$

where a'_i is randomly selected.

- Upon beacon loss (detected by local timer):

$$(\text{MIGRATE}, a_i, c_i) \rightarrow (\text{MIGRATE}, a'_i, c_i),$$

When tag i in the SETTLE state:

- Upon receiving ACK:

$$(\text{SETTLE}, a_i, c_i) \rightarrow (\text{SETTLE}, a_i, 0)$$

- Upon receiving NACK:

$$(\text{SETTLE}, a_i, c_i) \rightarrow \begin{cases} (\text{SETTLE}, a_i, c_i + 1) & \text{if } c_i + 1 < N \\ (\text{MIGRATE}, a'_i, 0) & \text{else} \end{cases}$$

- Upon beacon loss:

$$(\text{SETTLE}, a_i, c_i) \rightarrow (\text{MIGRATE}, a'_i, 0)$$

For any tag that does not transmit in this slot: If it detects a beacon loss by its timer, it will transit to MIGRATE state. Otherwise, it will maintain its current state.

It is straightforward to verify that the system satisfies the Markov property, that the next state X' depends only on the current state X , independent of the system's history.

Next, we aim to prove that the network will converge to a collision-free state. To do so, we show the network is an absorbing Markov chain by proving that:

- **Existence of an Absorbing State:** There exists a state, referred to as the absorbing state, where all tags operate collision-free, and no further transitions occur.
- **Reachability of the Absorbing State:** Starting from any initial state, the system will eventually reach this absorbing state with a non-zero probability.

C.2 Existing of Absorbing State

DEFINITION 1. *The reachable state space X_r is defined as the set of states that can be reached from any initial state through valid protocol transitions.*

LEMMA 1. *For any state in X_r , if all tags are in the SETTLE state, then the network is collision-free.*

PROOF. We show that Lemma 1 holds in all scenarios—normal operation, beacon loss, and late-arriving tags.

Normal Operation: We first demonstrate that the property holds under normal operation, where no desynchronization or new arrivals occur.

Assume, for contradiction, that two tags t_i and t_j are in the SETTLE state and collide in a slot s^* . During their transition to the SETTLE state, both tags must have transmitted in slot s^* . A collision would have been detected by the reader, resulting in a NACK for both tags. This contradicts the requirement for receiving an ACK to transition to the SETTLE state. Therefore, in normal operation, no two tags in the SETTLE state can collide.

Beacon Loss: Next, we address additional complexities introduced by potential beacon loss, showing that desynchronized tags cannot remain in the SETTLE state and hence cannot contribute to collisions.

A tag in the SETTLE state that misses a beacon cannot correctly increment its local slot index, potentially leading to misaligned transmissions and collisions. However, our design includes a timer mechanism to detect beacon loss (see Sec. 5.4). When a tag in the SETTLE state does not receive a beacon within the expected time window, the timer triggers the tag to automatically transition back to the MIGRATE state. This ensures that tags experiencing beacon loss cannot remain in the SETTLE state. While collisions due to beacon loss may occur temporarily, our design prevents tags in the SETTLE state from contributing to these collisions. Consequently, beacon loss does not introduce a counterexample to the lemma.

Late-Arriving Tags: Here, we analyze the situation with late-arriving tags and demonstrate how the protocol ensures that they cannot transition to the SETTLE state without fully resolving potential conflicts.

Seemingly, a late-arriving tag might not fully compete with existing tags and transitions directly to the SETTLE state. If, before its arrival, all other tags are already in the SETTLE state, this case could potentially violate the lemma: Now all tags are in SETTLE, but the late-arriving tag may later collide with tags that have longer periods. Fortunately, our protocol prevents this scenario (see Sec. 5.6). The reader monitors slot occupancy and predicts future collisions based on the transmission periods of all tags. If a late-arriving tag's

pattern are likely to cause future collisions, the reader withholds the ACK, preventing the tag from transitioning to the SETTLE state. Therefore, our design ensures that tags in the SETTLE state do not contribute to these collisions. \square

DEFINITION 2. *The absorbing states set X^* is defined as*

$$X^* = \{X \in X_r | \forall i, z_i = \text{SETTLE}\}$$

LEMMA 2. *Any state in X^* is absorbing.*

PROOF. When the network transits to a state in X^* , no further transitions occur:⁵ For transmitting tags: they receive ACKs (no collisions by Lemma 1), maintaining their SETTLE state. For non-transmitting tags, they maintain current state. Therefore, no tag transitions out of the current state, making the states in X^* absorbing. \square

C.3 Reachability of the Absorbing State

Let us denote the number of tags in the MIGRATE tags as M .

LEMMA 3. *From any initial state with $M \geq 0$ tags in the MIGRATE state, the network will eventually reach the absorbing state with probability 1.*

PROOF. Here, we use mathematical induction on the number of tags in the MIGRATE state, M , to prove this lemma.

Base Case: If $M = 0$ tags are in MIGRATE, all tags are already in SETTLE, and thus the absorbing state is reached. By Lemma 1, this state is collision-free.

Inductive Hypothesis: Assume that for any system state where $M = k$ tags are in the MIGRATE state, the system will eventually reach the absorbing state with probability 1.

Inductive Step: We now consider a state with $M = k + 1$ tags in the MIGRATE state. We need to show that there exist a strictly positive probability that the system can transition to a state with $M = k$ tags in MIGRATE. Once this reduction occurs, the inductive hypothesis guarantees that the system will eventually reach the absorbing state.

From the slot utilization condition (see Eq. 1), we know that the slot utilization does not exceed 1. By the collision avoidance mechanism (Sec. 5.6), for each tag i in the MIGRATE state, there exists at least one free slot offset a^* in its transmission period.

Without loss of generality, consider a specific tag i . The probability that tag i selects the free offset a^* is

$$P(\text{tag } i \text{ selects offset } a^*) = \frac{1}{p_i} > 0. \quad (6)$$

This follows the uniform randomness of slot selection over p_i possible slot offsets. For any other tag j (where $j \neq i$), the probability that tag j does not select specific slot is:

⁵Formally, the network should be modeled as a quasi-absorbing Markov chain. However, to simplify the proof, we assume that the beacon loss is sufficiently low ($\ll \frac{1}{\text{maximum tag period}}$). This assumption is reasonable because the beacon loss rate in our system is less than 0.1% (see Sec. 6.3). Under this assumption, we treat the Markov chain as absorbing, rather than quasi-absorbing, and focus solely on the system's convergence to the state where all tags are in the SETTLE state.

Table 4: Qualitative Comparison of Monitoring Solutions for Vehicle BiW

Aspect	Wired Sensors	RF-based Sensors	ARACHNET
Power Source	Wired power	Battery-powered	Battery-free
Integration Complexity	High (<i>new wires needed</i>)	Medium (<i>needs RF-transparent spots</i>)	Low (<i>attached directly to BiW</i>)
Deployment Cost	High (<i>wires, connectors, labor</i>)	Medium	Medium
Maintainability	Good	Poor (<i>due to battery</i>)	Good
Compatibility with BiW	Limited	Limited (<i>metal blocks RF</i>)	Good (<i>BiW as comm. medium</i>)
Data Throughput	High	Medium	Low

$$P(\text{tag } j \text{ avoids offset } a^*) = \begin{cases} 1 - \frac{1}{p_j} & \text{if } p_j > a_i \\ 1 & \text{if } p_j \leq a_i \end{cases} \quad (7)$$

Therefore, the probability that tag j avoids a^* is lower-bounded by

$$P(\text{tag } j \text{ avoids offset } a^*) \geq 1 - \frac{1}{p_j}. \quad (8)$$

Since the slot selection of the k other tags are independent, the probability that all k other tags avoid a^* is

$$P(\text{all other } k \text{ tag avoid offset } a^*) \geq \prod_{j \neq i} (1 - \frac{1}{p_j}) \quad (9)$$

The combined probability that tag i select free offset a^* and all other k tags avoid slot a^* is

$$P(\text{tag } i \text{ succeeds at } a^*) \geq \frac{1}{p_i} \prod_{j \neq i} (1 - \frac{1}{p_j}) \quad (10)$$

Since p_i and p_j is finite, we have $\frac{1}{p_i}$ and $1 - \frac{1}{p_j}$ for all tags i and j . The product of strictly positive terms is also strictly positive:

$$P(\text{tag } i \text{ succeeds at } a^*) > 0 \quad (11)$$

Thus, there is a strictly positive probability that tag i successfully selects a free slot and transitions to SETTLE. While the number of tags in MIGRATE may not decrease in every step, the probability that it remains at $k + 1$ indefinitely tends to zero. Therefore, the number of tags in the MIGRATE state will eventually decrease from $k + 1$ to k , and the system converges in expectation. \square

By mathematical induction, we have proven that for any initial state with $M \geq 0$ tags in the MIGRATE state, the system will

eventually reach an absorbing state with probability 1. Once in one absorbing state, the system remains collision-free.

C.4 Absorbing Markov Chain

THEOREM 4. *The network forms an absorbing Markov chain.*

PROOF. A Markov chain is absorbing if it has at least one absorbing state and there exists a positive probability to reach an absorbing state from any initial state. By Lemma 1, we prove that states in X^* where all tags are in SETTLE are collision-free, and Lemma 2 shows these states are absorbing. By Lemma 3, we prove that from any initial state with $M \geq 0$ tags in MIGRATE, the system will eventually reach X^* with probability 1. Once in X^* , by Lemma 1, the network remains collision-free. Therefore, the network forms an absorbing Markov chain that converges to a collision-free state. \square

D QUALITATIVE COMPARISON WITH CONVENTIONAL SOLUTIONS

As shown in Table 4, wired sensors offer reliable power and high data throughput, but suffer from high integration complexity and deployment cost, particularly when retrofitting the BiW, where new wiring is required. RF-based sensors reduce wiring needs but introduce challenges such as battery maintenance, limited compatibility with metallic BiW structures, and signal degradation due to RF-blocking materials. In contrast, ARACHNET provides a battery-free, low-cost solution that leverages the BiW itself as a communication medium. With low integration overhead, strong compatibility with the vehicle's metal body, and long-term maintainability, ARACHNET is especially well-suited for scalable, dense structural health monitoring in automotive environments where traditional approaches fall short.