HASH: A Novel WSN Protocol

Ahmed Fayez Elsousy
Electronics and Communications Engineering Department
The American University in Cairo
Cairo, Egypt
a.elsousy@aucegypt.edu

Hammad Omar
Electronics and Communications Engineering Department
The American University in Cairo
Cairo, Egypt
hammadomar@aucegypt.edu

Hassan Mohamed Hassan
Electronics and Communications Engineering Department
The American University in Cairo
Cairo, Egypt
hassanmohamed@aucegypt.edu

Sara Abdelhalim
Electronics and Communications Engineering Department
The American University in Cairo
Cairo, Egypt
saraabdelhalim@aucegypt.edu

***Abstract*—Wireless Sensor Networks are application-specific networks. This paper presents the design of a novel (WSN)-centric protocol called HASH. HASH, while being a complete MAC protocol, is an extensible protocol. The proposed protocol gives control to the network designer to add new features to it to tailor it for their use case. HASH accommodates a dynamic network where sensors can join and leave the network dynamically, trying to avoid collisions as much as possible. Overall, the proposed protocol provides an effective solution for WSN applications offering scalability and reliability.**

***Index Terms*—Wireless Sensor Networks, MAC Protocol**

I. Introduction

The rise of Internet of Things (IoT) technologies has transformed how we engage with and oversee our surroundings. Within the realm of smart buildings, IoT-driven solutions are pivotal in improving energy efficiency, occupant comfort, and overall building functionality. Wireless Sensor Networks (WSNs) emerge as a fundamental technology in this sphere, presenting a cost-efficient and scalable method for monitoring various environmental factors like temperature, humidity, and occupancy. This paper introduces a comprehensive WSN-based IoT application designed specifically for temperature monitoring within smart buildings. Our proposal addresses the escalating need for real-time environmental surveillance and management, offering building administrators actionable insights to fine-tune energy consumption and guarantee occupant well-being. The deployment of wireless sensor nodes throughout the building facilitates continual and widespread monitoring of temperature levels across diverse zones. A centralized server, or sink node, gathers data from these dispersed sensors, streamlining remote monitoring and analysis processes. Leveraging the capabilities of WSNs, our application streamlines data collection, aggregation, and transmission, ensuring prompt responses to temperature variations and compliance with predetermined thresholds. Furthermore, our system boasts high adaptability and scalability, capable of dynamically accommodating new sensor nodes. This adaptability guarantees the scalability and robustness of the monitoring infrastructure, facilitating seamless integration with evolving building layouts and needs.

II. System Description

The system comprises two essential node types: a sink node and a sensor (source) node. A sink node acts as a server or a central receiver, collecting data from sensors in a contention-based manner. Sensors contend for medium access. This choice was made so that the network could be as dynamic as possible. The system incorporates a push model, where the sensor nodes initiate the data transmission to the sink. The system was designed with temperature sensing in mind. So, the sink node allows for proper decision making after taking data (the temperature) from sensors. However, the system can work with any data type and can execute any decision making logic the designer may wish to use.

III. MAC Protocol

The proposed protocol is a contention-based protocol. It aims to minimize collisions and improve efficiency in data transmission while maintaining network flexibility by employing an approach that involves the use of Ready-To-Send (RTS) and Clear-to-Send (CTS) flow control packets. Refer to Figures 1 and 2 for the block diagrams of the system. The procedure goes as follows:

- For a node to join the network, it sends out a join request (Hi) packet to a sink node. When the sink node listens to it, it sends back a request acceptance (Hi back) packet.
- Before transmitting data, a sensor node sends out an RTS packet, informing the sink node that it wants to transmit data. The RTS packet contains information about the sender's identity and the destination, which is the sink in this case.
- Meanwhile, the sink is listening for packets. Once the sink receives a packet it identifies it. If it finds that the sink is the destination and the packet is an RTS packet, it sends out a CTS packet pointed at the node that sent the RTS packet.

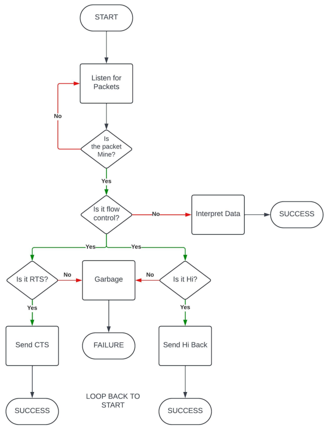

Fig. 1. Sink Node MAC logic

- The sensor will be waiting for CTS from the sink node. When it receives a packet, it examines it. If it is a CTS packet pointed at this sensor node, it proceeds to send the data it has.
- After sending the CTS packet, the sink will wait to receive the data packet pointed to it and will run any decision-making logic the designer assigned to the sink.
- Periodically, the sink node checks for dead nodes by listening for alive packets coming from joined nodes.

Using this protocol algorithm, collisions are reduced and data is delivered as correctly as possible. In addition, nodes can turn off their antenna whenever there is no data to send. This is a design choice for the network designer. More data safety features like parity checks, checksums, etc. can be implemented to further improve error immunity [1].

IV. Packet Structure

The structure used is the same structure used in the standard Internet Protocol (IP). As shown in Figure 3 is divided into a header part that contains information like fragmentation, checksum, source and destination addresses, etc. Then the payload part holds the data that is sent and/or received between nodes. However, we decided to implement our own packet composition inside the IP packet structure to further hone the proposed protocol for WSN applications, scalability and efficiency, as shown in Table I with an example of a packet. Table II shows how each node type is identified by its flag.

Dest Flag	**Dest ID**	**Src Flag**	**Src ID**	**Flow Control**	**Msg**
$0F$	01	03	02	1	23

TABLE I
HASH Packet composition

A flag in the packet identifies the node type of that created the packet and the node that is the destination of the packet. The ID of a node is the number or address of a node. The flow control flag identifies the packet as a flow control packet or not. Then the message comes afterwards, which can be data

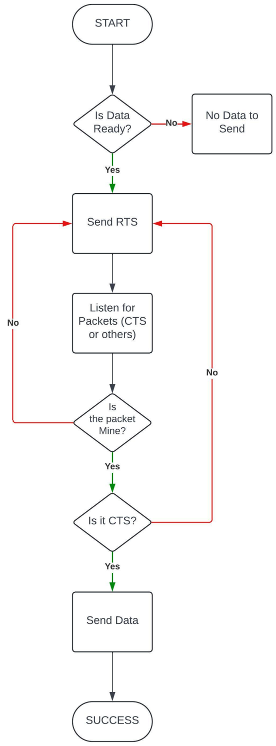

Fig. 2. Sensor Node MAC Logic

Node	**Flag (in hexadecimal)**
Sink	$0F$
Sensor	03

TABLE II
Node flags

or flow control messages. Each of the aforementioned parts of the packet is of 1 byte size except the message which can hold more than 1 byte of data.

V. Statistics

Inter-packet spacing and Packet Delivery Ratio (PDR) are closely related and define the performance and reliability of the network. As the IPS decreases, indicating a shorter time between packets, there is a shorter time to execute the protocol logic. This can result in packet loss. Our methodology were to make 2 runs of sending 10 data packets with a network of 3 sensor nodes and sink node using the proposed protocol and checked how many dara packets reached the sink. The experiment results are in Figure 4. The decision to make 2 runs of packets was to minimize the collision with already existing Wi-Fi packets in the medium as we are using the Wi-Fi physical medium, which is the ISM 2.4 GHz radio band. As the IPS increased, the PDR increased as well, which is expected. However, the latency of the system was an issue, especially with the longer time space between packets.

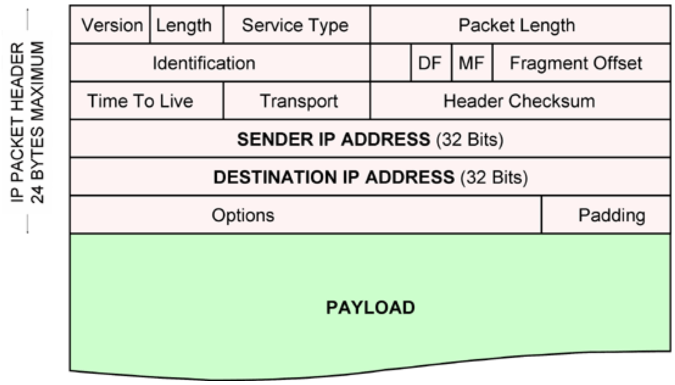

Fig. 3. Standard IP Packet Structure

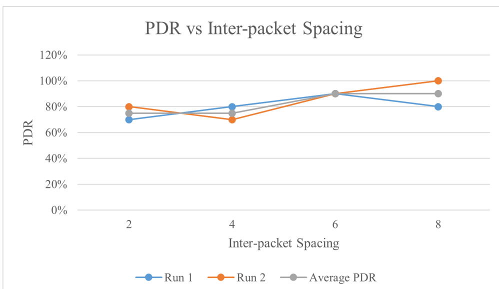

Fig. 4. Packet Delivery Ratio Inter-packet spacing

VI. Conclusion

HASH, the proposed MAC protocol, is a contention-based collision-avoidant protocol made with Wireless Sensor Networks first in mind. It is modular, extensible, dynamic, and adaptive. This enables the network designer to tailor it to their own design requirements.

VII. Future Work

The proposed protocol is flexible and allows for adding logic layers on top of its base easily. Hence, it is not hard to implement a routing layer on top of it by using hierarchical routing with a sink backbone. Another useful addition would be to add logic to choose what channel the network will be operating on and change it dynamically to always be operating on the best channel possible.

Acknowledgment

This project would not have been possible without the team who worked on it. The team worked together and helped in brainstorming and solving the challenges that were a hurdle.

References

[1] W. Dargie and C. Poellabauer, *Fundamentals of wireless sensor networks*. John Wiley & Sons, 11 2010.