

Exploring Generalizable Wireless Embedded System Protocols for Low-Cost Field Robots

Arvind Kaushik*, Mukul Yadav†, Vikhram Sundararaghavan‡, and Neehal Sharrma†

*Department of Robotics Systems Engineering, Arizona State University, Tempe, Arizona, USA

†Open Horizon Robotics, Pune, Maharashtra, India

‡Department of Mechanical Engineering, SRM, Chennai, Tamil Nadu, India

Abstract—Low-cost field mobile robots need to rely on expensive external networking hardware components for short and long distance communication. To that end, natively supported protocols in low-cost MCUs may be a suitable alternative to drive down the technical complexity and costs for these systems. However, despite the variety of options available, most protocols are bound to the specific hardware platform, further restricting modularity. We explore various natively-supported as well as generic embedded communication protocols and evaluate their generalizability to different low-cost MCU platforms to highlight their potential, demonstrated using the NodeMCU.

Index Terms—Embedded Communication, Microcontrollers, ESPNow, MQTT, LoRaWAN, MicroROS, Low-Cost

I. INTRODUCTION

Robotics is a very huge field with a lot of practical implementations and a huge market, especially in applications that require multiple robots working in tandem with one another. As may be trivially known, multi-robot systems rely on robust communication protocols between agents, and thus, the embedded systems aboard each robot – usually microcontrollers (MCUs) and single-board computers (SBCs) – play a very important role in this [citation needed]. However, although a variety of low-cost commercial off-the-shelf (COTS) embedded systems exist in the market, since each manufacturer follows different board architectures, not every device will have the same native networking capabilities. This leads engineers to use dedicated external modules attached to the main board to implement cross-board communication, which may prove to be expensive. This may constrain robots to using only one type of device across all robots to leverage their native networking features, which can also be suboptimal for the application. This work explores the following questions:

- Is it possible to use only natively supported networking protocols to achieve short and long-distance communication between robots?
- To what extent can we deploy such protocols, and which devices can we deploy this across?

This material is based on original unfunded work. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors. All correspondence should be addressed to Arvind Kaushik arvindemail@asu.edu

II. RELATED WORK

Low-cost embedded devices have been a popular choice for multi-device interconnectivity, especially in the context of IoT [1]. When deploying multiple devices, reducing the cost and resources utilized per device is beneficial – a similar consideration made when deploying multi-robot systems. Thus, some research has been conducted into evaluating how such low-cost systems can be used in mobile robots while meeting stringent Size, Weight, Power and Cost (SWaP-C) constraints as well as the types of communication methods possible in such systems [2]. Given the hypothesis we wish to explore, some prior works using natively-supported and generic protocols exist, but we could not find any that extensively explore and contrast these protocols in a single study, especially for mobile robot applications. Thus, in this work, we seek to fill this informational gap.

III. METHODOLOGY

For this work, we decided to explore the following:

- **ESPNow**: ESP32 protocol for short-range Bluetooth communication, native to the NodeMCU board family.
- **MicroROS**: Minimal build of ROS2 deployable onto MCUs and SBCs.
- **MQTT**: Generic networking broker protocol deployable on almost all network-capable devices.
- **LoRaWAN**: Popular long-range protocol often used in IoT applications.

All the protocols listed above have been widely used in low-cost robotics applications and are deployable on native hardware to some extent. To implement these protocols, we used a low-cost DIY quadruped and humanoid walker robot, also shown in Fig. 2. Each robot is outfitted with an ESP32 NodeMCU board and other onboard electronics. The total cost for both robots is under \$50.

IV. EXPERIMENTS AND ANALYSIS

For each of the protocols we wished to explore in this work, we used the two robots in Fig. 2 to set up a direct communication line between them. Each experiment implemented ESPNow, MicroROS, MQTT, and LoRaWAN in order, each protocol seemingly showing functional improvements over the previous one towards generalizability. Table I collates salient

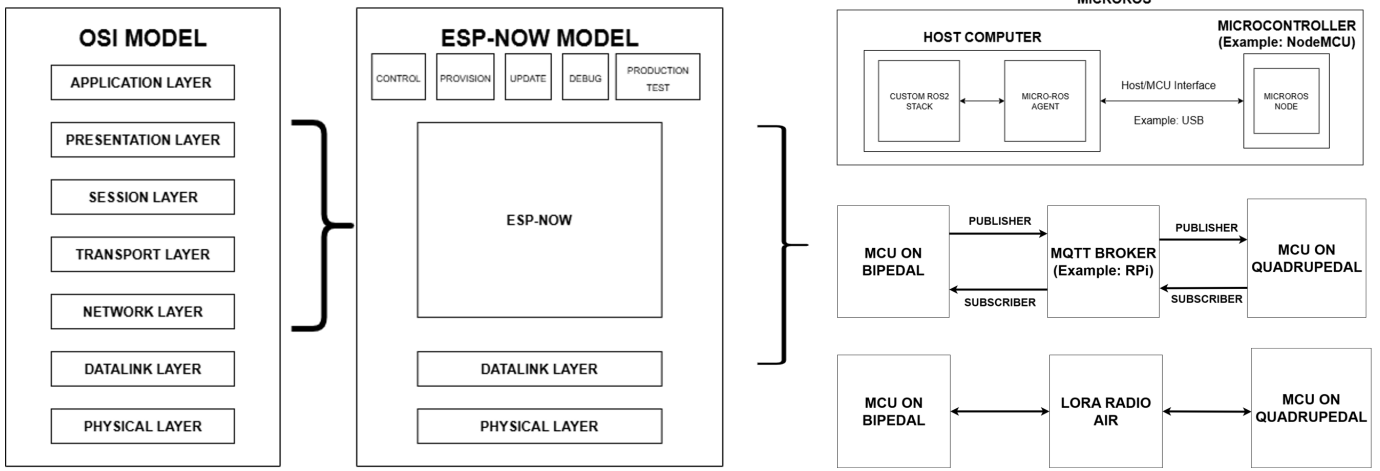


Fig. 1. Architectural diagrams comparing the OSI model with other protocols.

Protocol	Advantages	Disadvantages
ESPNow	Native out-of-the-box (OOTB) support	Unreliable for long-range communication; Only compatible with other NodeMCUs; No datalogging available
MicroROS	Compatible with any MCU/SBC Compatible with existing ROS2 tools and libraries	Complex to implement Limited bandwidth makes it unreliable for long-distance communication Requires a MicroROS agent to run simultaneously alongside normal code
LoRaWAN	Very long communication possible ESP32s with integrated LoRa modules available	Signal weakness and distortion if antenna missing
MQTT	Central mediator MCU for easy triaging No tight coupling between each subscriber	Require separate hardware for central broker

TABLE I
COMPARISON TABLE OF THE VARIOUS PROTOCOLS EXPLORED IN THIS WORK.

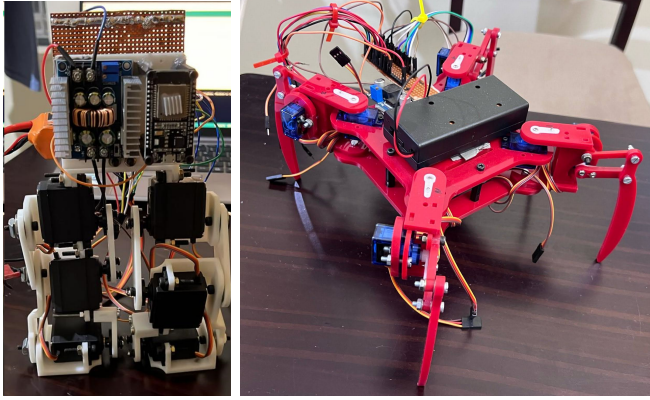


Fig. 2. The low-cost humanoid walker and quadruped robots used to deploy the algorithms explored in this work.

findings from these experiments. From the table, we can see that ESPNow is highly board-dependent, which negates it as a good protocol for generalized communication. MicroROS and MQTT appear to be generalizable to all boards, albeit with some caveats – the always-on service for MicroROS and the separate central broker for MQTT – but LoRaWAN seems to be the most generalizable to all use cases with reliable integrability into a real-time operating system (RTOS) that most MCUs nowadays rely on for multimodal IoT applications.

V. CONCLUSION AND FUTURE WORK

By observing Table I and Fig. 2, one can easily see the strengths of each protocol depending on the situation. For example, ESPNow is best for ultra-low-cost applications requiring quick, OOTB support without having to spend effort setting up any communication channels. Similarly, MicroROS allows MCUs to leverage the cross-platform compatibility of ROS but deployed onto a small platform that enables access to powerful ROS2 libraries for higher-order processing.

We also stumbled across works discussing possible LLM integration for programming mobile robots [3]. Preliminary exploration with integrating ChatGPT into the current communication setup allowed for adaptive commands based on data passed into the context window; further research is necessary to fully flesh out the full capabilities of LLM-based communication protocols.

REFERENCES

- [1] V. A. S. Herrera, H. P. de Araújo, C. G. Penteado, M. Gazziro, and J. P. Carmo, “Low-cost embedded system applications for smart cities,” *Big Data and Cognitive Computing*, vol. 9, no. 2, 2025. [Online]. Available: <https://www.mdpi.com/2504-2289/9/2/19>
- [2] A. L. Rodrigues Gonçalves, D. Alves de Lima, and A. Maia, “Visible light communication applied to low cost embedded systems,” *IEEE Latin America Transactions*, vol. 22, no. 1, pp. 15–21, 2024.
- [3] Z. Hu, F. Lucchetti, C. Schlesinger, Y. Saxena, A. Freeman, S. Modak, A. Guha, and J. Biswas, “Deploying and evaluating llms to program service mobile robots,” *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2853–2860, 2024.