ISSN 2523-4455. MicrosystElectronAcoust, 2026, vol. 31, no. 1 355559.1

UDC 681.518:615.477:004.7

DOI: 10.20535/2523-4455.mea.355559

WiFi-Based Control System for Active Lower Limb Orthosis: A Distributed Architecture Approach

A. V. Kotsiubailo, PhD student
O. M. Lysenko, Dr.Sc.(Eng.), Prof., 0000-0003-1051-1149
Department of Design of Electronic Computing Equipment http://keoa.kpi.ua
Faculty of Electronics
National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, 00syn5v21
Kyiv, Ukraine

***Abstract*—This research presents the design, multi-threaded software implementation, and empirical evaluation of an active, electromyography (EMG)-driven upper-limb orthosis utilizing a distributed master-slave embedded control architecture. Engineered specifically to address the clinical challenges of severe upper-limb paresis—such as those resulting from brachial plexus trauma—the system bypasses traditional localized muscle control by acquiring voluntary surface EMG signals from the sternocleidomastoid (SCM) neck muscle, utilizing it as a stable proxy for motion intent. To ensure hard real-time deterministic execution and high-fidelity signal processing, the system is physically and logically decoupled into three operational layers. A dedicated analog front-end (AD8232) and an STM32F407 microcontroller handle localized biosignal conditioning and digitization. A Jetson Nano single-board computer operates as the supervisory node, executing a strictly governed finite state machine (FSM) to perform digital filtering, feature extraction, and motion classification. Finally, an ESP32 microcontroller paired with a BTS7960 high-current H-bridge manages real-time brushed DC motor actuation. Communication across these distinct hardware layers is achieved wirelessly via a TCP/IP protocol over Wi-Fi, eliminating cumbersome physical tethers and establishing a foundation for advanced Internet of Things (IoT) integration. Software execution relies on a multi-threaded, mutex-protected environment capable of achieving an average sample-to-result latency of 6.02 milliseconds. Real-time joint angle feedback is continuously provided by an AS5048A absolute magnetic encoder via an SPI interface, ensuring kinematic safety and rigid adherence to predefined anatomical limits. Experimental results validate the system's capacity for safe, highly responsive operation, demonstrating that complex classification algorithms and wireless distributed topologies can successfully meet the rigorous latency requirements of wearable rehabilitation robotics, though future iterations may necessitate the integration of brushless actuators to mitigate static friction and low-speed operational limitations.**

Keywords — upper limb orthosis; electromyography signal; neck muscle activity.

I. INTRODUCTION

Upper-limb motor impairments caused by neurological disorders or trauma often result in reduced functional independence and limited ability to perform activities of daily living. Wearable robotic orthoses and exoskeletons have emerged as a promising solution to support rehabilitation and functional assistance by providing controlled mechanical support to impaired joints [1][2].

A critical challenge in the design of such systems is the development of intuitive and reliable human–machine interfaces. Conventional control strategies based on residual limb motion, joint torque sensing, or manual input devices may be unsuitable for users with severe motor deficits. As a result, electromyography (EMG)-based control has gained significant attention as a means of directly capturing user intent through neuromuscular activity [3].

Most EMG-controlled upper-limb orthoses rely on signals acquired from the muscles directly involved in the assisted motion [4]. However, in patients with partial or complete paresis, these muscles may exhibit weak, noisy, or inconsistent activation patterns. To address this limitation, alternative muscle groups that remain voluntarily controllable can be exploited [5]. In particular, neck muscles such as the sternocleidomastoid (SCM) have been shown to provide stable and repeatable EMG signals even in cases of severe upper-limb impairment [6], [7].

This work presents the design of an EMG-controlled elbow orthosis that utilizes SCM muscle activity as the primary control input. The system integrates a dedicated on-board EMG acquisition module, consisting of an AD8232 analog front-end [8] and an STM32 microcontroller, enabling local biosignal preprocessing and feature extraction directly on the wearable device. This approach

DOI: 10.20535/2523-4455.mea.355559

improves signal robustness, reduces communication bandwidth, and decouples biosignal acquisition from high-level processing tasks.

The proposed architecture employs a distributed control framework, in which EMG acquisition and preprocessing, motion intent recognition, and real-time actuation are implemented as independent software layers. High-level decision-making is separated from safety-critical motor control, ensuring reliable operation and fault tolerance in wearable conditions.

The objective of this study is to demonstrate a modular and scalable EMG-based control system for elbow assistance that supports natural, hands-free interaction and can be adapted to users with limited residual limb function. By combining neck-muscle EMG sensing with embedded signal processing and distributed control [9], [10], the proposed solution aims to advance the practicality and usability of assistive upper-limb orthoses, developed in our previous work [7].

II. SYSTEM ARCHITECTURE

The developed elbow orthosis is implemented as a distributed embedded system, in which high-level supervisory control is separated from low-level actuation and sensing tasks. This architectural approach improves system reliability, simplifies software development, and enables future scalability. The overall system consists of a EMG acquisition unit, supervisory embedded computer, a microcontroller-based actuator unit, sensing elements, and a wireless communication interface, as illustrated conceptually in Fig. 1.

The system architecture follows a master–slave model. EMG acquisition unit collects data and transfers data to a single-board computer (Jetson Nano) which operates as the supervisory controller, responsible for user interaction, system state management, and network communication. An ESP32 microcontroller serves as the actuator-level controller, executing real-time motor control commands and managing hardware-level safety states.

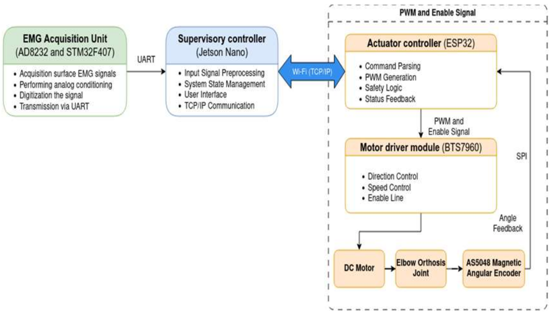

Fig. 1 Block diagram of the distributed embedded control architecture of the elbow orthosis

Communication between the supervisory and actuator layers is realized via a Wi-Fi connection using a TCP/IP protocol [8], [11]. This design choice eliminates the need for physical communication cables between control units, which is beneficial for wearable and modular rehabilitation devices.

The core architectural components include:

- **EMG Acquisition Unit** (AD8232 and STM32F407)
- **Supervisory controller** (Jetson Nano),
- **Actuator controller** (ESP32),
- **Motor driver module** (BTS7960),
- **Angular position sensor** (AS5048A),
- **DC motor and mechanical elbow joint interface**.

A. *EMG Acquisition Unit*

Surface electromyographic signals are acquired using an AD8232-based analog front-end, optimized for low-amplitude biopotential measurement. The conditioned EMG signal is digitized and preprocessed by an STM32F407 microcontroller.

The EMG acquisition unit operates independently from the actuator control loop and transmits processed EMG features to the supervisory unit via UART protocol. This separation reduces noise coupling from power electronics and improves signal stability during motor operation.

B. *Supervisory Control Layer*

The supervisory layer is implemented on a Jetson Nano embedded computer running a Linux-based operating system. This layer hosts a Python-based control server responsible for:

- establishing and maintaining network connections with actuator nodes,
- transmitting control commands and configuration parameters,
- receiving status feedback from the orthosis,
- maintaining global system state variables (e.g., enabled/disabled state, motor activity).

The supervisory software is designed around a socket-based TCP server, enabling reliable bidirectional communication. Each connected actuator node is handled in a separate execution thread, allowing future extension to multi-node or multi-joint orthotic systems.

A key architectural feature is the explicit separation between system enable/disable control and motor activation. Upon startup, the orthosis is forced into a disabled state, preventing unintended motion. Only after an explicit enable command from the supervisory layer can motor-related commands be executed. This state-based logic enhances operational safety and reduces the risk of accidental actuation.

C. Actuator Control Layer

The actuator control layer is implemented on an ESP32 microcontroller, selected for its integrated Wi-Fi capabilities, sufficient computational resources, and suitability for real-time motor control tasks [12]. The ESP32 firmware is responsible for:

- interpreting commands received from the supervisory controller,
- controlling motor direction and speed via PWM signals,
- enabling or disabling power delivery to the motor driver,
- generating status messages reflecting the current system state.

The ESP32 operates as a TCP client, initiating and maintaining a persistent connection to the supervisory server [11]. Command messages are parsed as text-based instructions, simplifying debugging and system integration. This lightweight protocol allows rapid modification and extension of command sets without requiring changes to the underlying communication stack.

D. Motor Drive and Power Interface

Motor actuation is achieved using a BTS7960 high-current H-bridge motor driver [13]. This driver is capable of supplying sufficient current for assistive elbow motion while providing electrical isolation between control and power stages. The ESP32 generates pulse-width modulation (PWM) signals on two independent channels to control motor direction and speed.

An additional enable line is used to physically activate or deactivate the motor driver. This hardware-level enable mechanism operates independently of software motor commands, providing an extra safety layer. When the orthosis is disabled, the enable line is forced low, ensuring that no motor motion can occur even in the presence of erroneous control signals.

E. Sensing and Feedback Integration

Angular position feedback is provided by a magnetic absolute encoder (AS5048A), mounted coaxially with the elbow joint rotation axis [14]. The encoder supplies high-resolution angular measurements, enabling precise monitoring of joint position and motion range.

The sensor data are used to enforce predefined angular limits and to support closed-loop motion control strategies. In the current implementation, the sensing subsystem is integrated to provide real-time joint position awareness and system monitoring, forming the basis for future adaptive control extensions.

III. SOFTWARE IMPLEMENTATION

The orthosis software is implemented as a multi-threaded, event-driven system (Fig.2) running on the Jetson Nano, designed to integrate EMG acquisition, signal processing, classification, motor actuation, and joint angle feedback in a deterministic and safe manner. The software is organized around a central finite state machine (FSM) [15], which governs state transitions and coordinates the execution of all threads to ensure that actuation only occurs in response to validated EMG input and within safe joint limits.

EMG signals are continuously streamed from the STM32F407 acquisition module over Wi-Fi. The Acquisition Thread on the Jetson Nano continuously reads these samples and stores them in a circular buffer which size is 500 samples. To prevent data races, access to the buffer is protected by a mutex [16], and when a full window of EMG samples is collected, a window_ready_flag is set to trigger signal processing.

The Processing Thread is triggered by the window-ready event. It applies digital filtering, rectification, and normalization to the EMG window and extracts relevant features, which are stored in a feature buffer protected by a mutex. Upon completing signal conditioning, a processing_done_flag is set, signaling the Classification Thread. This thread evaluates the feature buffer and classifies intended motion such as elbow flexion or extension. Validated motor commands are enqueued into a thread-safe command queue, also protected by a mutex.

The Actuation Thread executes on the ESP32 and performs two critical functions: it continuously monitors the command queue to execute motor commands via the BTS7960 H-bridge, and it performs real-time acquisition of joint angle measurements from the AS5048A magnetic rotary encoder. The encoder is read over the SPI interface, and angular positions are stored in a mutex-protected angle buffer. Each angle sample is timestamped and immediately used for local safety checks and feedback to the FSM on the Jetson Nano. Safety routines monitor joint limits, preventing over-extension, and when a new angle sample is available, an angle_ready_flag is set to synchronize the Jetson Nano motion control [17].

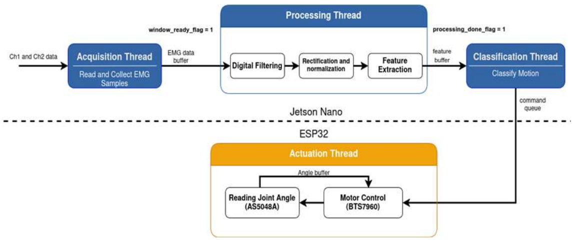

Fig. 2 Multi-threaded EMG-controlled orthosis software architecture

DOI: 10.20535/2523-4455.mea.355559

DOI: 10.20535/2523-4455.mea.355559

Synchronization between threads is achieved through atomic flags, mutex-protected buffers, and thread-safe queues, while timeouts prevent deadlocks and discard stale commands. All threads operate under the governance of the FSM, which defines states including Idle, Signal Conditioning, Onset/Offset Detection, Classification, Actuation, and Return-to-Idle. State transitions are triggered by EMG acquisition events, feature processing completion, activation detection, classification results, joint angle feedback, and safety timeouts. This architecture ensures low-latency, deterministic execution, isolates acquisition, processing, classification, and actuation into concurrent threads, and maintains safe, responsive operation of the orthosis.

IV. RESULTS

The implemented orthosis system was evaluated to measure its real-time performance and the latency of EMG-driven motor actuation. All tests were conducted under standard operational conditions, with EMG signals acquired at 1 kHz [18], motor commands executed via the BTS7960 H-bridge, and joint angles monitored using the AS5048A encoder.

Real-time performance metrics were collected for each software stage, including windowing, filtering, onset/offset detection, classification, and total processing. The windowing stage exhibited an average execution time of 1.20 ms, with a maximum of 16.94 ms and a standard deviation of 1.48 ms. Signal filtering was performed efficiently, with an average of 0.36 ms, a maximum of 0.67 ms, and minimal variability (Std: 0.03 ms). The onset-offset detection module required on average 0.32 ms (Max: 0.71 ms, Std: 0.04 ms), reflecting low-latency detection of muscle activation. Classification, which is the most computationally intensive stage, averaged 46.14 ms, reaching a maximum of 48.86 ms, with a standard deviation of 1.06 ms. Across all stages, the total processing time per EMG window averaged 1.69 ms, with a maximum of 53.22 ms and Std of 8.03 ms.

The overall sample-to-result latency, representing the time from EMG acquisition to validated motor command execution on the actuator, was measured at an average of 6.02 ms, with a maximum of 58.64 ms and a standard deviation of 8.52 ms. These results demonstrate that the multi-threaded architecture, synchronization via mutexes and flags, and FSM-based control allow the system to operate with deterministic timing and minimal delays, ensuring that motor actuation closely follows muscle activation while respecting safety limits.

The integration of joint angle feedback from the AS5048A encoder further confirms that the orthosis responds accurately to EMG inputs, maintaining actuation within predefined angular limits. These results highlight the effectiveness of the real-time software implementation and validate the multi-threaded, FSM-driven design in providing safe and responsive orthotic assistance.

V. DISCUSSION AND LIMITATIONS

While the presented orthosis demonstrates effective EMG-driven control and real-time responsiveness, several limitations and practical nuances were identified during system development and testing. The current implementation relies on brushed DC motors [18], which introduce audible noise during operation. To mitigate the impact on EMG acquisition and user comfort, the system is organized as two logically independent subsystems, separating signal processing and actuation. Although this architecture ensures safety and deterministic operation, it adds complexity in synchronizing state transitions and feedback, particularly when both EMG-driven and motor state events occur concurrently.

Another limitation is related to the mechanical characteristics of the actuator and linkage system. At low motor speeds, the orthosis exhibits limited responsiveness and may fail to initiate movement reliably. This is partly due to the static friction and inertia of the mechanism, which reduce actuation precision and limit the ability to perform fine-grained motions at minimal EMG activations. Consequently, the system shows reduced adaptability in scenarios requiring slow or subtle joint movements.

Additionally, the current feedback loop, while effective for safety and FSM-driven control, depends on the accurate and continuous sampling of the AS5048A encoder. Any intermittent communication delays or packet loss can slightly affect the real-time coordination between EMG detection, classification, and motor actuation. Despite these limitations, the system provides a robust platform for upper-limb assistance, demonstrating predictable performance for voluntary movements above the low-speed threshold and ensuring joint protection through integrated angle monitoring. Future work may include the integration of brushless motors, improved low-speed control, and more tightly coupled hardware-software synchronization to enhance smoothness and responsiveness.

CONCLUSION

This work presents a multi-threaded, EMG-driven upper-limb orthosis integrating real-time signal acquisition, processing, classification, and motor actuation with joint angle feedback. The system demonstrates deterministic performance, with low-latency responses between EMG detection and motor execution, enabled by a finite state machine (FSM) coordinating acquisition, processing, and actuation threads. The integration of the AS5048A encoder ensures safe and accurate joint movement, while thread-safe buffers and mutex-based synchronization maintain data integrity across concurrent operations.

DOI: 10.20535/2523-4455.mea.355559

Experimental evaluation confirms that the orthosis reliably interprets voluntary muscle activations and executes corresponding movements within safe angular limits, with an average sample-to-result latency of approximately 6 ms. Despite limitations related to brushed motor noise and reduced performance at low speeds, the proposed architecture provides a robust platform for EMG-controlled upper-limb assistance, offering potential for rehabilitation and daily support applications. Future improvements may include the adoption of brushless actuators, enhanced low-speed control, and more tightly integrated hardware-software synchronization, further improving responsiveness, comfort, and usability.

REFERENCES

[1] R. A. R. C. Gopura, D. S. V. Bandara, K. Kiguchi, and G. K. I. Mann, "Developments in hardware systems of active upper-limb exoskeleton robots: A review," Rob. Auton. Syst., vol. 75, pp. 203–220, Jan. 2016, doi: 10.1016/j.robot.2015.10.001.

[2] A. M. Dollar and H. Herr, "Lower Extremity Exoskeletons and Active Orthoses: Challenges and State-of-the-Art," IEEE Trans. Robot., vol. 24, no. 1, pp. 144–158, Feb. 2008, doi: 10.1109/TRO.2008.915453.

[3] M. Asghari Oskoei and H. Hu, "Myoelectric control systems—A survey," Biomed. Signal Process. Control, vol. 2, no. 4, pp. 275–294, Oct. 2007, doi: 10.1016/j.bspc.2007.07.009.

[4] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," IEEE Trans. Biomed. Eng., vol. 50, no. 7, pp. 848–854, Jul. 2003, doi: 10.1109/TBME.2003.813539.

[5] M. Barsotti et al., "A full upper limb robotic exoskeleton for reaching and grasping rehabilitation triggered by MI-BCI," in 2015 IEEE International Conference on Rehabilitation Robotics (ICORR), IEEE, Aug. 2015, pp. 49–54. doi: 10.1109/ICORR.2015.7281174.

[6] J. R. Cram, G. S. Kasman, and J. Holtz, Introduction to surface electromyography. Aspen Publishers, 1998.

[7] A. V. Kotsiubailo et al., "Designing Minimalistic Powered Arm Orthosis for Brachial Plexus Injuries," Visnyk NTUU KPI Seriia - Radiotekhnika Radioaparatobuduvannia, no. 96, pp. 50–61, Jun. 2024, doi: 10.20535/RADAP.2024.96.50-61.

[8] "AD8232 Single-Lead Heart Rate Monitor Front End (Datasheet)," 2012. Accessed: Mar. 25, 2026. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ad8232.pdf

[9] A. Phinyomark, P. Phukpattaranont, and C. Limsakul, "Feature reduction and selection for EMG signal classification," Expert Syst. Appl., vol. 39, no. 8, pp. 7420–7431, Jun. 2012, doi: 10.1016/j.eswa.2012.01.102.

[10] A. Wang, J. Dong, R. Teng, P. Liu, X. Yue, and X. Zhang, "A Hierarchical Distributed Control System Design for Lower Limb Rehabilitation Robot," Technologies, vol. 13, no. 10, p. 462, Oct. 2025, doi: 10.3390/technologies13100462.

[11] H. Alemdar and C. Ersoy, "Wireless sensor networks for healthcare: A survey," Comput. Networks, vol. 54, no. 15, pp. 2688–2710, Oct. 2010, doi: 10.1016/j.comnet.2010.05.003.

[12] "ESP32 Technical Reference Manual (Datasheet)," 2021. Accessed: Mar. 25, 2026. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

[13] "BTS7960 High Current Half Bridge (Datasheet)," 2011. Accessed: Mar. 25, 2026. [Online]. Available: https://www.infineon.com/dgdl/BTS7960_Datasheet.pdf?folderId=db3a304412b407950112b408e8c9

[14] "AS5048A/AS5048B 14-bit On-Axis Magnetic Rotary Position Sensor," 2013. Accessed: Mar. 25, 2026. [Online]. Available: https://ams-osram.com/products/sensor-solutions/position-sensors/ams-as5048a-high-resolution-position-sensor

[15] N. Vitiello et al., "NEUROExos: A Powered Elbow Exoskeleton for Physical Rehabilitation," IEEE Trans. Robot., vol. 29, no. 1, pp. 220–235, Feb. 2013, doi: 10.1109/TRO.2012.2211492.

[16] A. Williams, C++ concurrency in action : practical multithreading. Manning, 2012.

[17] "Jetson Nano Developer Kit User Guide," 2019. Accessed: Mar. 25, 2026. [Online]. Available: https://developer.nvidia.com/embedded/jetson-nano-developer-kit

[18] K. W. Horch and D. Kipke, Neuroprosthetics : theory and practice. World Scientific Publishing Co. Pte. Ltd., 2017.

Надійшла до редакції 25 березня 2026 року
Прийнята до друку 15 квітня 2026 року
Опублікована 01 травня 2026 року.

DOI: 10.20535/2523-4455.mea.355559

УДК 681.518:615.477:004.7

Розподілена архітектура системи Wi-Fi-керування активним ортезом верхніх кінцівок

А. В. Коцюбайло, аспірант

О. М. Лисенко, д-р техн. наук, проф., 0000-0003-1051-1149
Кафедра конструювання електронно-обчислювальної апаратури http://keoa.kpi.ua
Факультет електроніки
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського" 00syn5v21
Київ, Україна

Анотація— Це дослідження представляє розробку, багатопотокову програмну реалізацію та емпіричну оцінку активного ортеза верхніх кінцівок з електроміографічним (ЕМГ) керуванням, що використовує розподілену вбудовану архітектуру керування типу «ведучий-ведений» (master-slave). Система спроектована спеціально для вирішення клінічних проблем, пов'язаних із тяжкими парезами верхніх кінцівок — наприклад, внаслідок травм плечового сплетення. Вона обходить традиційні методи локального керування м'язами, зчитуючи сигнали довільної поверхневої ЕМГ з груднинно-ключично-соскоподібного (ГКС) м’яза шиї, використовуючи його як стабільний проксі-сигнал інтенції руху. Для забезпечення детермінованого виконання у жорсткому реальному часі та високої точності обробки сигналів, система фізично та логічно розділена на три робочі рівні. Спеціалізований аналоговий інтерфейс (AD8232) та мікроконтролер STM32F407 відповідають за локальну підготовку та оцифрування біосигналів. Одноплатний комп'ютер Jetson Nano функціонує як керуючий вузол, виконуючи суворо регламентований скінченний автомат (FSM) для цифрової фільтрації, виділення ознак та класифікації рухів. Нарешті, мікроконтролер ESP32 у поєднанні з високострумовим H-мостом BTS7960 керує роботою щіткового двигуна постійного струму в реальному часі. Зв'язок між цими різними апаратними рівнями здійснюється бездротовим способом за протоколом TCP/IP через мережу Wi-Fi, що усуває громіздкі фізичні з'єднання та створює основу для інтеграції в розширену екосистему Інтернету речей (IoT). Програмне забезпечення базується на багатопотоковому середовищі із захистом м'ютексами, що дозволяє досягти середньої затримки «від вибірки до результату» (sample-to-result) на рівні 6,02 мілісекунди. Зворотний зв'язок щодо кута суглоба в реальному часі безперервно забезпечується абсолютним магнітним енкодером AS5048A через інтерфейс SPI, що гарантує кінематичну безпеку та суворе дотримання визначених анатомічних меж. Експериментальні результати підтверджують здатність системи до безпечної та високочутливої роботи, демонструючи, що складні алгоритми класифікації та бездротові розподілені топології можуть успішно відповідати суворим вимогам щодо затримки в носимій реабілітаційній робототехніці. Водночас майбутні ітерації можуть потребувати інтеграції безщіткових приводів для мінімізації статичного тертя та обмежень роботи на низьких швидкостях.

Ключові слова — ортез верхніх кінцівок, сигнал електроміографії, активність м'язів шиї.