

# Control Remoto de Vehículo a Escala Usando ESP32 y Bluetooth con Mando PS5

## *Remote Control of a Scale Vehicle Using ESP32 and Bluetooth with PS5 Controller*

- **Juan José González Hernández**, es alumno de la Universidad Politécnica de Guanajuato (México) (19030830o@upgto.edu.mx), (<https://orcid.org/0009-0002-9842-9771>). Estudiante.
- **Julio Cesar Peña-Aguirre**, es profesor en la Universidad Politécnica de Guanajuato (México) (jcpena@upgto.edu.mx), (<https://orcid.org/0000-0002-6211-7760>). Doctor.
- **Yosafat Jetsemani Samano-Flores**, es profesor en la Universidad Politécnica de Guanajuato (México) (patricia.torres@uaslp.mx), (<https://orcid.org/0000-0001-6122-476X>). Máster.
- **Tomás Serrano-Ramírez**, es profesor en la Universidad Politécnica de Guanajuato (México) (juan.dd@matehuala.tecnm.mx), (<https://orcid.org/0000-0001-6118-3830>), Máster.

**Resumen:** El presente estudio analiza el desarrollo de un sistema de control remoto para un vehículo a escala, utilizando un microcontrolador ESP32 y conectividad Bluetooth mediante un mando de PS5. Este proyecto se justifica por la necesidad de explorar soluciones de bajo costo y simplicidad para el control de vehículos pequeños, aprovechando tecnologías accesibles y ampliamente documentadas como Arduino. El objetivo es implementar un sistema eficiente y autónomo que permita el control inalámbrico de las funciones básicas del vehículo, incluyendo motores y luces indicadoras. La investigación se clasifica como experimental, ya que se diseñaron y evaluaron prototipos en condiciones controladas. La metodología incluyó la integración de hardware (motores, servomotores, LEDs, y ESP32) y software programado en Arduino para la gestión de comandos Bluetooth. Los resultados mostraron que el sistema es efectivo dentro de un rango de 10 metros, con un tiempo de respuesta mínimo y un consumo de energía adecuado, permitiendo una autonomía de 30 minutos. Sin embargo, se identificaron limitaciones en cuanto a la necesidad de mantener una línea de visión despejada para la conexión Bluetooth y la posibilidad de optimizar el código para mejorar la eficiencia energética. Las conclusiones destacan la viabilidad del sistema para aplicaciones similares y proponen mejoras futuras, como la integración de sensores de proximidad y la expansión a conectividad Wi-Fi para ampliar su alcance y funcionalidad.

**Palabras clave:** Control inalámbrico, microcontrolador, Bluetooth, vehículo, Arduino.

**Cómo citar:** González-Hernández, J.J.; Peña-Aguirre, J.C.; Samano-Flores, Y.J.; y Serrano-Ramírez, T. (2024). Control Remoto de Vehículo a Escala Usando ESP32 y Bluetooth con Mando PS5. *Tecnología, Ciencia y Estudios Organizacionales*, 6(12), pp. 206-214. <https://doi.org/10.56913/teceo.6.12.206-214>

Recepción: 14-10-2024

Aprobación: 23-10-2024

**Abstract:** The present study analyzes the development of a remote-control system for a scale vehicle using an ESP32 microcontroller and Bluetooth connectivity with a PS5 controller. The project is motivated by the need to explore cost-effective and straightforward solutions for small vehicle control, utilizing accessible and well-documented technologies like Arduino. The main objective is to implement an efficient, low-cost, and autonomous system that enables wireless management of the vehicle's essential functions, including motor control and indicator lights. This experimental research involved designing and testing prototypes under controlled conditions to evaluate performance and identify limitations. The methodology combined hardware integration, including DC motors, a servomotor, LEDs, and the ESP32, with software development in Arduino to manage Bluetooth commands and execute control tasks. The results demonstrated that the system is effective within a 10-meter range, with a minimal response time that supports smooth operation. The system also achieved an energy consumption level that allows for a 30-minute operational period with a standard battery. However, some limitations were noted, such as the need for a clear line of sight for stable Bluetooth connectivity and the opportunity to optimize the code to enhance energy efficiency. Future improvements could include integrating additional modules, such as proximity sensors, and expanding the system's capabilities to enable remote control over the Internet via Wi-Fi, thus extending its range and application potential. These findings confirm the system's viability for similar applications and provide a foundation for further enhancements.

**Keywords:** Wireless control, microcontroller, Bluetooth, vehicle, Arduino.

### Introducción

En la actualidad, el control inalámbrico ha revolucionado la manera en que interactuamos con dispositivos y vehículos, permitiendo una mayor versatilidad al eliminar las limitaciones de los controles físicos convencionales. Esta tecnología ha facilitado el desarrollo de sistemas de control remoto que son no solo más eficientes, sino también más accesibles y funcionales. El uso de microcontroladores, como el ESP32, es el eje central en varios estudios que destacan la importancia de estos en la automatización, robótica y aplicaciones militares. El artículo de (Marapalli et al., 2021) detalla el diseño de un vehículo autónomo militar controlado por gestos para patrullaje de fronteras, integrando tecnologías como IoT, GPS y cámaras para detectar obstáculos. De manera similar, (Hamici, 2023) propone un robot autónomo basado en un modelo matemático de movimiento, también apoyado en el ESP32 y con la capacidad de planificación de trayectorias. (Gonzales, s/f; Yadav Kanneboina et al., 2024) abordan el desarrollo de sistemas autónomos y de automatización de dispositivos, subrayando la eficiencia energética, la seguridad y el uso de comunicación GSM y GPS.

Varios estudios se centran en la robótica y el control remoto de vehículos. El trabajo de (Yordanov & Deliyiski, 2023) diseñan un sistema de control remoto para una pequeña embarcación usando comunicación Bluetooth y ESP32, mientras que (Cornella i Barba & Sangenís Rafart, 2020) presentan un coche RC controlado vía Wi-Fi mediante una aplicación de C#. En el trabajo de (Cheng et al., 2022; Dwinanto et al., 2023) desarrollan vehículos de control remoto y robots inteligentes, respectivamente, utilizando microcontroladores para tareas de rescate y extinción de incendios, enfatizando el uso de cámaras y sensores para la captura de datos y la mejora de la movilidad.

En cuanto a la eficiencia energética, (Swathi et al., 2022) proponen un sistema de alumbrado público inteligente que emplea sensores de movimiento para optimizar el uso de energía, destacando la importancia de la automatización en la conservación de recursos.

El enfoque en la enseñanza también es abordado por (Abekiri et al., 2023), quienes desarrollan una plataforma de laboratorios remotos para prácticas educativas utilizando ESP32 y NOD-red, permitiendo a los estudiantes realizar experimentos de manera remota, fomentando el aprendizaje autónomo.

En este proyecto, se propone el uso de un microcontrolador ESP32 programado mediante la plataforma Arduino para controlar de forma remota un vehículo a escala utilizando un mando de PS5 (Play Station 5, por sus siglas en inglés), haciendo uso del protocolo de comunicación Bluetooth.

El ESP32 se seleccionó debido a sus características avanzadas, que incluyen una capacidad de procesamiento robusta y la integración tanto de conectividad Bluetooth como Wi-Fi en un solo chip. Esto lo convierte en una opción ideal para proyectos que requieren un control inalámbrico eficiente y de bajo costo. Por otro lado, la plataforma Arduino proporciona un entorno de programación intuitivo y ampliamente documentado, lo que facilita el desarrollo y la implementación del sistema. En la literatura hay antecedentes de trabajos entre la plataforma de Arduino IDE y la tarjeta ESP32, por ejemplo algunos utilizan el ESP32 para desarrollar sistemas de monitoreo y adquisición de datos en tiempo real. Por ejemplo, (Allafi & Iqbal, 2017) presentan un servidor web de bajo costo para monitorear sistemas fotovoltaicos en tiempo real utilizando el ESP32, que recolecta datos de sensores y los almacena en una tarjeta SD accesible remotamente. De manera similar, (Ikiss, 2020) muestra un sistema de adquisición de datos para medir variables eléctricas como corriente y voltaje, con dos placas ESP32 conectadas a circuitos de muestreo que transmiten datos a una computadora vía Wi-Fi. (Babiuch et al., 2019) también exploran el uso del ESP32 para procesamiento de datos, integrándolo con módulos IoT y pantallas OLED para medir y procesar datos en tiempo real.

El ESP32 también se ha utilizado ampliamente en proyectos de automatización del hogar y otras aplicaciones IoT. (Niranjana et al., 2022) desarrollan un sistema de automatización del hogar utilizando el ESP32 y el servidor Blynk para operar electrodomésticos de manera remota a través de internet. Por otro lado, (Hercog et al., 2023) destacan la importancia educativa del ESP32 en el desarrollo de habilidades IoT en instituciones académicas, presentando herramientas y proyectos educativos que introducen a los estudiantes en la implementación de dispositivos IoT con ESP32.

El trabajo reportado por (Rai & Rehman, 2019) proponen un sistema de vigilancia inteligente basado en el ESP32, utilizando su capacidad Wi-Fi para transmitir video en tiempo real y mostrarlo en un módulo TFT. Esta implementación demuestra la capacidad del ESP32 para desarrollar sistemas de seguridad de bajo costo que pueden ser útiles en entornos industriales y domésticos.

Por otro lado, el trabajo (Dokic et al., 2020) comparan la eficiencia de MicroPython y Arduino C en el ESP32 para dispositivos de red neuronal, destacando las aplicaciones de aprendizaje automático en microcontroladores.

El objetivo principal de este proyecto es adaptar un carrito a control remoto (RC) utilizando un ESP32 programado con Arduino para ser controlado a través de un mando de PS5 mediante conexión Bluetooth (rodneymbakiskan, 2022/2024). Este sistema permitirá el manejo remoto de las

funciones básicas del vehículo, incluyendo la operación de los motores y las luces indicadoras, proporcionando una experiencia de control precisa y versátil.

Este trabajo demuestra cómo se puede implementar un sistema de control inalámbrico económico y de fácil integración en proyectos de vehículos remotos utilizando un ESP32 y la plataforma Arduino (*Hands-on ESP32 with Arduino IDE*, s/f). La elección de estos componentes simplifica el desarrollo, al mismo tiempo que mantiene la flexibilidad y eficiencia del sistema. Asimismo, el proyecto busca servir como referencia para el diseño de aplicaciones similares, destacando las ventajas de utilizar dispositivos de bajo costo y ampliamente accesibles en el desarrollo de sistemas embebidos y de control remoto.

### Método

El desarrollo del sistema se divide en dos partes principales: el diseño del hardware y la implementación del software.

#### Hardware Utilizado

El material utilizado para el desarrollo del proyecto se enlista a continuación

- Carrito de juguete.
- ESP32.
- Motores y servomotor.
- Fuente de alimentación.
- LEDs indicadores.
- Mando de PS5.

El proyecto utiliza un vehículo de juguete como base, el cual ha sido modificado para integrar el microcontrolador ESP32, los motores CD (Corriente Directa) y el controlador de potencia. El vehículo actúa como la plataforma móvil del proyecto; por otro lado el microcontrolador ESP32, es el componente central del sistema, encargado de recibir comandos a través de Bluetooth desde un mando de PS5 y controlar tanto los motores como las luces indicadoras del vehículo; Se emplean motores CD para el movimiento lineal del carrito y un servomotor MG90S para el control de la dirección, proporcionando así la capacidad de maniobrar en diferentes direcciones. El sistema es alimentado por baterías que suministran energía tanto al ESP32 como a los motores y los LED's (Light Emissor Diode, por sus siglas en inglés), asegurando la operación autónoma del vehículo. Se incorporan luces para señalar las direcciones del movimiento y la reversa, mejorando la interacción y el control visual del vehículo.

#### Software

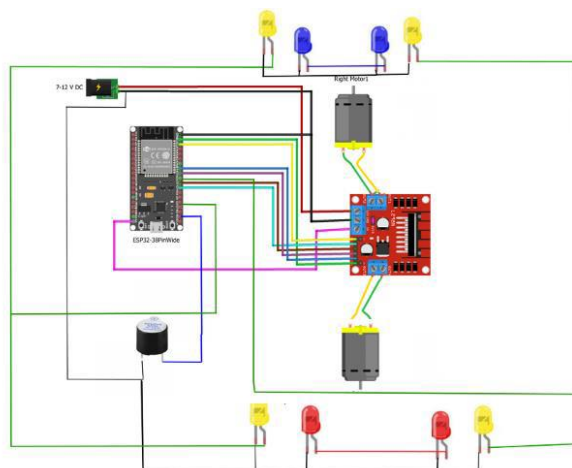
El software del sistema fue desarrollado utilizando el Arduino IDE, lo que permitió aprovechar una amplia variedad de bibliotecas ya existentes para la comunicación Bluetooth y el control de motores. Esto facilitó la integración y configuración de los distintos componentes, haciendo el desarrollo más eficiente y accesible.

#### Esquema de Conexiones

El ESP32 se conecta a los motores a través de un controlador de motor tipo puente H, lo que permite controlar la dirección y la velocidad de los motores CD, como se muestra en la Figura 1. Además, se utilizan pines GPIO para manejar el servomotor y los LEDs indicadores. La



**Figura 1.** Conexiones ESP32 – Motores CD, bocina, drivers, y LED's.



### Implementación del Protocolo Bluetooth

Para habilitar la comunicación Bluetooth en el ESP32, se utilizó la librería BluetoothSerial en Arduino. Esta librería facilita la recepción de comandos desde el mando PS5, los cuales son interpretados y ejecutados por el ESP32. El código desarrollado permite que el vehículo interprete acciones como avanzar, retroceder, girar a la izquierda o derecha, y activar las luces indicadoras.

### Desarrollo del Código en Arduino

El código se estructura en funciones específicas para cada acción, incluyendo el control de los motores y el servomotor, así como la activación de las luces de señalización.

Los comandos recibidos por Bluetooth son procesados y ejecutados de acuerdo con la lógica programada en el ESP32, asegurando una respuesta precisa del sistema.

### Procedimiento de la programación.

El funcionamiento del sistema sigue el siguiente procedimiento:

1. El ESP32 permanece en espera de comandos Bluetooth provenientes del mando PS5.
2. Cuando se recibe un comando, este se procesa para identificar si corresponde a un movimiento del vehículo o a la activación de las luces.
3. Dependiendo del comando, el ESP32 ejecuta la acción correspondiente, ya sea el control de los motores para el movimiento o la activación de los LEDs para las señales visuales.

### Resultados

Se llevaron a cabo diversas pruebas para evaluar el desempeño del carrito controlado por Bluetooth, enfocándose en la respuesta a los comandos de dirección y en el alcance máximo de la señal.

Los ensayos demostraron que el sistema responde dentro de un rango de 10 metros, con un retardo mínimo en la transmisión de los comandos desde el mando PS5 hasta el ESP32. Este tiempo de respuesta permite una operación fluida y precisa del vehículo. La Figura 2 muestra la conexión

física entre la ESP32 y los componentes que se mostraron en la Figuta 1, como lo es el puente H, motores CD, bocina y LED's.



**Figura 2.** *Conexiones del circuito en físico.*



En términos de consumo de energía, se observó que la configuración del sistema permitió una autonomía de aproximadamente 30 minutos utilizando una batería estándar. Este rendimiento se considera adecuado para pruebas iniciales, aunque se reconoce que hay espacio para optimizar el sistema en términos de eficiencia energética.

Sin embargo, también se identificaron ciertas limitaciones durante las pruebas. La conexión Bluetooth mostró dependencia de una línea de visión despejada para mantener la estabilidad de la comunicación, lo que implica que cualquier obstáculo significativo entre el mando y el vehículo podría afectar la conectividad. Además, se detectó la necesidad de optimizar el código para mejorar el manejo de la energía, con el fin de extender la autonomía del sistema en futuras iteraciones del proyecto.

### **Discusión**

Los resultados obtenidos demuestran que el sistema de control remoto desarrollado, basado en Bluetooth y utilizando el microcontrolador ESP32, es adecuado para aplicaciones en vehículos pequeños donde se prioriza la simplicidad y el bajo costo. La respuesta eficiente dentro de un rango de 10 metros y el retardo mínimo en la transmisión de comandos confirman que esta configuración es efectiva para lograr un control preciso y rápido, esencial en proyectos de este tipo.

Al comparar este enfoque con otros basados en tecnologías como Wi-Fi, se observa que el uso de Bluetooth presenta ventajas en términos de consumo energético, ya que es considerablemente más bajo, lo que permite una mayor autonomía con la misma capacidad de batería. No obstante, la principal desventaja de Bluetooth es su alcance limitado en comparación con Wi-Fi, lo cual restringe las aplicaciones a distancias cortas y a entornos donde la línea de visión entre el mando y el vehículo pueda mantenerse sin obstáculos significativos.

En cuanto a posibles mejoras, se sugiere una optimización del código para reducir aún más el consumo de energía y maximizar la autonomía del sistema. Adicionalmente, la inclusión de módulos adicionales, como sensores de proximidad, podría incrementar la funcionalidad del vehículo al permitirle reaccionar de manera automática ante obstáculos. Por último, para expandir las capacidades del sistema, se podría integrar conectividad Wi-Fi, permitiendo así el control del

vehículo a través de Internet, lo que ampliaría considerablemente su alcance y aplicabilidad en entornos más complejos.

### Referencias

- Abekiri, N., Rachdy, A., Ajaamoum, M., Nassiri, B., Elmahni, L., & Oubail, Y. (2023). Platform for hands-on remote labs based on the ESP32 and NOD-red. *Scientific African*, 19, e01502. <https://doi.org/10.1016/j.sciaf.2022.e01502>
- Allafi, I., & Iqbal, T. (2017). Design and implementation of a low cost web server using ESP32 for real-time photovoltaic system monitoring. *2017 IEEE Electrical Power and Energy Conference (EPEC)*, 1–5. <https://doi.org/10.1109/EPEC.2017.8286184>
- Babiuch, M., Foltýnek, P., & Smutný, P. (2019). Using the ESP32 Microcontroller for Data Processing. *2019 20th International Carpathian Control Conference (ICCC)*, 1–6. <https://doi.org/10.1109/CarpathianCC.2019.8765944>
- Cheng, J., Ye, M., & Zhang, Q. (2022). Design of intelligent remote control car based on bluetooth. *2022 18th International Conference on Computational Intelligence and Security (CIS)*, 205–209. <https://doi.org/10.1109/CIS58238.2022.00050>
- Cornella i Barba, G., & Sangenis Rafart, E. (2020). *WIFI remote controlled RC car commanded by a C# application & the ESP32 WIFI module* [Bachelor thesis, Universitat Politècnica de Catalunya]. <https://upcommons.upc.edu/handle/2117/341789>
- Dokic, K., Radisic, B., & Cobovic, M. (2020). MicroPython or Arduino C for ESP32—Efficiency for Neural Network Edge Devices. En C. Brito-Loeza, A. Espinosa-Romero, A. Martin-Gonzalez, & A. Safi (Eds.), *Intelligent Computing Systems* (pp. 33–43). Springer International Publishing. [https://doi.org/10.1007/978-3-030-43364-2\\_4](https://doi.org/10.1007/978-3-030-43364-2_4)
- Dwinanto, B., Lubis, A. A., Karjadi, M., Yulianto, B., & Harianto, B. (2023). Prototype Smart Car Mecanum Wheel Fire Extinguisher Based On Esp 32 Cam With Bluetooth Combination. *Jurnal Indonesia Sosial Teknologi*, 4(8), 1110–1122. <https://doi.org/10.59141/jist.v4i8.689>
- Gonzales, H. G. (s/f). *REVOLUTIONIZING APPLIANCE AUTOMATION: ESP32 AND BLYNK*. Recuperado el 14 de octubre de 2024, de [https://www.researchgate.net/profile/Helen-Grace-Gonzales/publication/372315159\\_REVOLUTIONIZING\\_APPLIANCE\\_AUTOMATION\\_ESP32\\_AND\\_BLYNK/links/64af8d6095bbbe0c6e2f83e6/REVOLUTIONIZING-APPLIANCE-AUTOMATION-ESP32-AND-BLYNK.pdf](https://www.researchgate.net/profile/Helen-Grace-Gonzales/publication/372315159_REVOLUTIONIZING_APPLIANCE_AUTOMATION_ESP32_AND_BLYNK/links/64af8d6095bbbe0c6e2f83e6/REVOLUTIONIZING-APPLIANCE-AUTOMATION-ESP32-AND-BLYNK.pdf)
- Hamici, N. (2023). *Design and Implementation of an Autonomous Mobile Robot Based on ESP-32* [Working Paper]. University of Guelma. <http://dspace.univ-guelma.dz/jspui/handle/123456789/14998>
- Hands-on ESP32 with Arduino IDE: Unleash the power of IoT with ESP32 and build exciting projects with this practical guide.* (s/f). Recuperado el 14 de octubre de 2024, de <https://ieeexplore.ieee.org/abstract/document/10460876>
- Hercog, D., Lerher, T., Truntič, M., & Težak, O. (2023). Design and Implementation of ESP32-Based IoT Devices. *Sensors*, 23(15), Article 15. <https://doi.org/10.3390/s23156739>

- Ikiss, J. (2020). *Sistema de adquisición de datos con ESP32* [Bachelor thesis, Universitat Politècnica de Catalunya]. <https://upcommons.upc.edu/handle/2117/344400>
- Marapalli, K., Bansode, A., Dundgekar, P., & Rathod, N. (2021). AIGER An Intelligent Vehicle for Military Purpose. *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 1, 1052–1057. <https://doi.org/10.1109/ICACCS51430.2021.9441998>
- Niranjana, R., S, A., M, V., & S, V. (2022). Effectual Home Automation using ESP32 NodeMCU. *2022 International Conference on Automation, Computing and Renewable Systems (ICACRS)*, 1–5. <https://doi.org/10.1109/ICACRS55517.2022.10028992>
- Rai, P., & Rehman, M. (2019). ESP32 Based Smart Surveillance System. *2019 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, 1–3. <https://doi.org/10.1109/ICOMET.2019.8673463>
- rodneybakiskan. (2024). *Rodneybakiskan/ps5-esp32* [C]. <https://github.com/rodneybakiskan/ps5-esp32> (Obra original publicada en 2022)
- Swathi, B., Reshma, K., Azeez, R. F., & Mohan, R. C. (2022). *Micro Python Powered Smart Street Light using IOT*. <https://www.academia.edu/download/88421373/CSEIT1228381.pdf>
- Yadav Kanneboina, R., Satish Kumar, G., Vidyadhar, R. P., Reddy, V. K., & Tammineni, S. (2024). Development of Autonomous Vehicle System using ESP32 Microcontroller. *2024 Asia Pacific Conference on Innovation in Technology (APCIT)*, 1–6. <https://doi.org/10.1109/APCIT62007.2024.10673667>
- Yordanov, Y., & Deliyski, R. (2023). DESIGN AND DEVELOPMENT OF A REMOTE CONTROL FOR A SMALL WATERCRAFT. *CHALLENGES IN HIGHER EDUCATION AND RESEARCH IN 21 ST CENTURY*, 69.