Enviado: 24/07/2024 *Recibido: 30/08/2024* *Aceptado: 02/11/2024*

UNIVERSIDAD DISTRITAL
FRANCISCO JOSE DE CALDAS

VISIÓN ELECTRÓNICA

Algo más que un estado sólido

https://doi.org/10.14483/issn.2248-4728

VISIÓN ELECTRÓNICA

Quadruped Robot Prototype for Agricultural Mobile Robotics

Prototipo de robot cuadrúpedo para robótica móvil agrícola

Gissel Daniela Cristiano Ayala[1], Samuel Alejandro Sabogal Mora[2], David Esteban Fuquen Garces[3], Emily Angelica Villanueva Serna[4], Henry Borrero Guerrero[5]

Abstract

Agricultural robots with wheels have contributed significantly to agriculture, generating interest in developing robots with improved maneuverability in crop fields where wheels do not work well [1]. This has led to the development of legged mobile robots as a viable alternative to wheeled robots [2,3,4], and for this reason, a study group at the Universidad Francisco José de Caldas Distrital in Colombia has been developing a small-scale quadruped robot in order to examine its maneuverability capabilities in difficult terrain. In this paper, the corresponding robot and its results are described. Using 3D printing, we developed a robot structure that could accommodate eight servomotors. Using the CircuitPython programming language, we programmed a Raspberry Pico-W microcontroller as the control unit [5]. Thanks to a graphical

---

[1] Electronic Engineering student, Universidad Distrital Francisco José deCaldas, Colombia. E-mail: gdcristianoa@udistrital.edu.co ORCID: https://orcid.org/0009-0000-5043-2897

[2] Electronic Engineering student, Universidad Distrital Francisco José deCaldas, Colombia. E-mail: sasabogalm@udistrital.edu.co ORCID: https://orcid.org/0009-0009-7706-5025

[3] Electronic Engineering student, Universidad Distrital Francisco José deCaldas, Colombia. E-mail: defuqueng@udistrital.edu.co ORCID: https://orcid.org/0009-0008-3184-9126

[4] Electronic Engineer, Universidad Distrital Francisco José de Caldas, Colombia. E-mail: eavillanuevas@udistrital.edu.co

[5] Electronic Engineer, Universidad de los Llanos, Colombia. PhD in Mechanical Engineering, University of São Paulo, Brazil. Postdoctoral Researcher, BITPOINTER SAS, Colombia, in agreement with the Ministry of Science and Technology and Innovation of Colombia. Email: hbguerrero@ieee.org. ORCID: https://orcid. org/0000-0003-4243-4205

user interface developed in HTML, a quadruped robot could be controlled remotely from a computer or even a mobile phone. As reported in the results, our quadruped robot approach has demonstrated promising results in its maneuverability on off-road soils, opening the possibility of developing autonomous quadruped robots for agricultural use.

Keywords: Agricultural Robot, Legged Locomotion, CircuitPython, Mobile Robotics, Quadruped Robot, Raspberry Pi Pico.

Resumen

Los robots agrícolas con ruedas han contribuido significativamente a la agricultura, lo que ha generado interés en desarrollar robots con mejor maniobrabilidad en campos de cultivo donde las ruedas no funcionan adecuadamente. Esto ha dado lugar al desarrollo de robots móviles con patas como una alternativa viable a los robots con ruedas. Por esta razón, un grupo de estudio en la Universidad Distrital Francisco José de Caldas, en Colombia, ha estado desarrollando un robot cuadrúpedo a pequeña escala con el fin de examinar sus capacidades de maniobra en terrenos difíciles. En este artículo se describe el robot correspondiente y sus resultados. Utilizando impresión 3D, se diseñó una estructura capaz de alojar ocho servomotores. Mediante el lenguaje de programación CircuitPython, se programó un microcontrolador Raspberry Pi Pico-W como unidad de control. Gracias a una interfaz gráfica desarrollada en HTML, el robot cuadrúpedo puede ser controlado de forma remota desde un computador o incluso desde un teléfono móvil. Según lo reportado en los resultados, nuestro enfoque con el robot cuadrúpedo ha demostrado resultados prometedores en su maniobrabilidad sobre suelos irregulares abriendo la posibilidad de desarrollar robots cuadrúpedos autónomos para uso agrícola.

Palabras clave: Locomoción con patas, CircuitPython, Robótica agrícola, Robótica móvil, Robot cuadrúpedo, Raspberry Pi Pico.

1. Introduction

In the course of conducting a research project focused on the development of autonomous navigation systems for agricultural mobile robotics [6], we observed that wheeled robots are unsuitable for certain environments, such as the Colombian Andes, where agriculture significantly contributes to economic development [7,8].

As indicated in [9,10], the deployment and operation of mobile robots in outdoor environments present significant challenges due to the unstructured nature of these settings, necessitating robust locomotion and substantial computational capabilities. According to [10], quadruped robots are generally favored for their mobility, ease of control, and stability [11]. The interest in wheeled agricultural robots has led to the development of robots with enhanced maneuverability in crop fields, where wheeled systems often underperform. A review of the literature on wheeled robots indicates that this type of robot is not extensively utilized in

Colombia, underscoring the importance of advancing this area of knowledge within the region. Consequently, a group of students from the Universidad Distrital Francisco José de Caldas in Colombia is engaged in the development of a small-scale quadruped robot to assess its maneuverability capabilities in simulated terrains.

The objectives outlined are as follows: General Objective: To create a prototype of a quadruped robot to assess its capabilities on various terrains and its potential use in agricultural support. Specific Objectives: To utilize the Raspberry Pi Pico microcontroller for motion control implementation. To assess the prototype's effectiveness on simulated terrains. To enable remote operation of the robot by connecting it to a computer or mobile device.

2. Theoretical Framework

Over time, the design of quadruped robots has evolved to become more efficient, compact, and capable. Early designs were often large and relied on hydraulic systems, which limited their practical applications due to size, weight, and power constraints [12, 13, 14].

Advances in electric actuators and battery technology have facilitated the development of lighter and more agile robots with extended operating times. Quadruped locomotion is commonly employed when the task assigned to the robot necessitates mobility [15]. When implementing legged locomotion in a robot, it is crucial to consider its position and speed, while also ensuring that the robot maintains balance and does not fall [16, 17], relying solely on joint movement through motors [18, 19].

CircuitPython is a programming language designed to simplify experimentation and learning to code on a low-cost microcontroller. It includes an organized set of libraries developed by Adafruit that complement CircuitPython. These libraries enable control of sensors, displays, motors, communications, and more [20].

The Raspberry Pi Pico W is a development board based on the RP2040 microcontroller, designed and built by the engineers at Raspberry Pi. It has been designed as a flexible, lowcost development platform with a 2.4 GHz wireless interface [21]. The SG90 servo is one of the most versatile and widely used servomotors in various robotics projects. It is small, yet it provides considerable torque of 1.8 kg/cm, making it suitable for a wide range of robots [22].

3D printing is a process of creating objects by depositing layers of material on top of each other. It is referred to as additive manufacturing (AM), as opposed to traditional subtractive methods such as CNC milling, particularly when used for industrial production. This technology has existed for about four decades, having been invented in the early 1980s. Although 3D printing initially began as a slow and expensive technique, significant technological advances have made modern AM technologies more affordable and faster than ever before [23]. Quadruped locomotion is the technique by which a four-legged robot is given movement, aiming to mimic the various forms of locomotion found in animals through nature [24]. The kinematic diagram of the leg makes it possible to identify the equations of the kinematic chain in both Cartesian and joint coordinates, considering only geometric aspects. Dynamic analysis, on the other hand, allows for the determination of forces, as well as the velocities and accelerations generated in the various joints due to these forces, whether caused by moments of inertia, centers of gravity, external forces, among others [25, 26, 27].

3. Results

a. Energy

Table 1: Energy consumption by type of movement.

Movement	Total Time (s)	Average current (mA)	Energy consumed (mAh)	Observations
Forward	1.33	520	0.19	Stable running, average consumption
Backward	1.34	580	0.22	Retraction with moderate load
Right	1.1	460	0.14	Moderate expenditure, 3 active legs
Left	1.06	470	0.14	Similar to right, slight variation
Dance	0.69	800	0.15	Fast movement, high consumption
Greet	16.5	650	2.92	Prolonged movement with an active but stable neck

Source: own.

b. Speed

Table 2: Locomotion speed by movement.

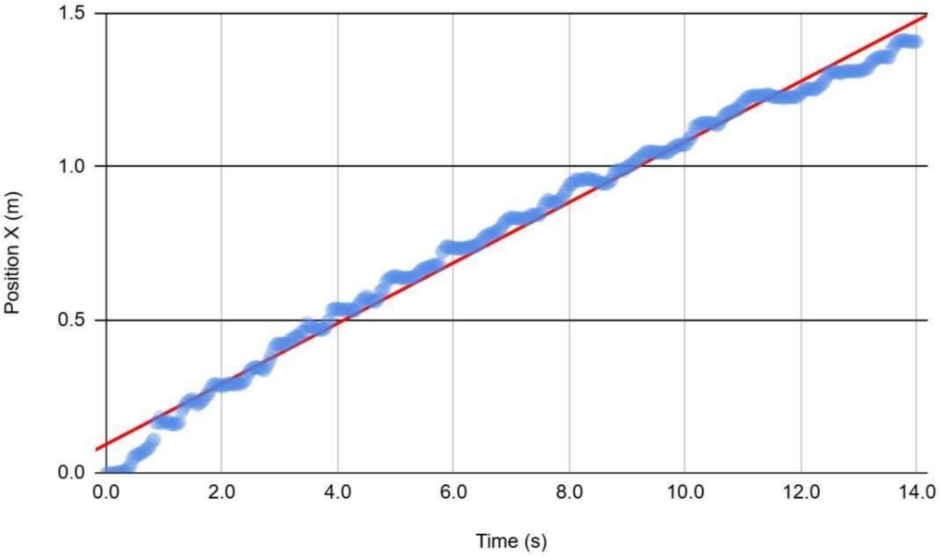
Movement	Time between steps (s)	Full cycle (s)	Frequency (Hz)	Distance (cm)	Total time (s)	Speed (cm/s)
Forward	0.3	1.33	0.75	100	14.12	7.082
Backward	0.3	1.34	0.75	100	12.21	8.19

Source: own.

c. Coordinates

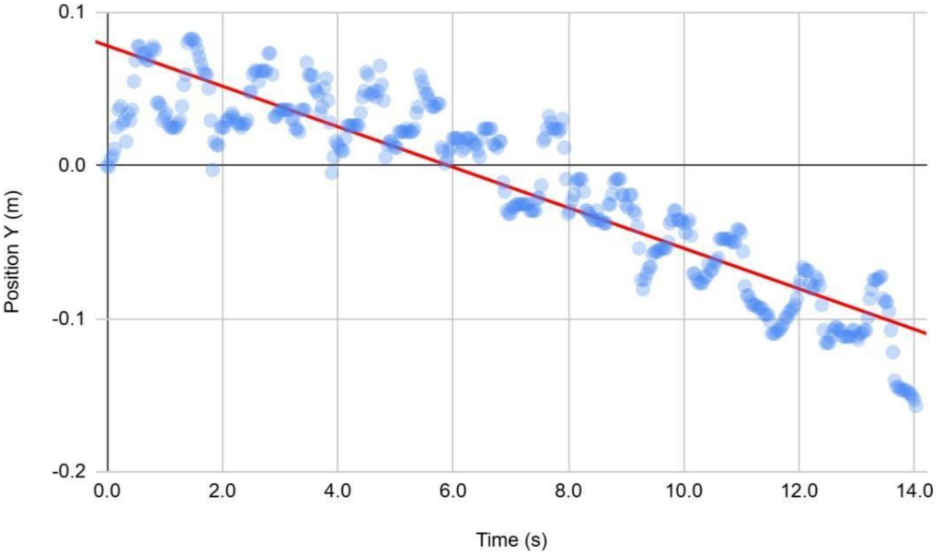
i. Forward

Figure 1: Time VS Position X “Forward”.



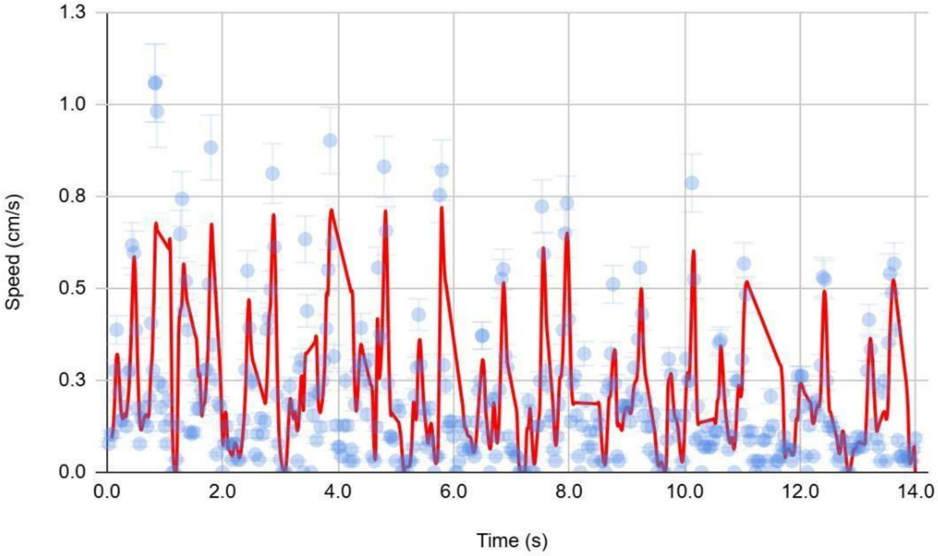
Source: own.

Figure 2: Time VS Position Y “Forward”.



Source: own.

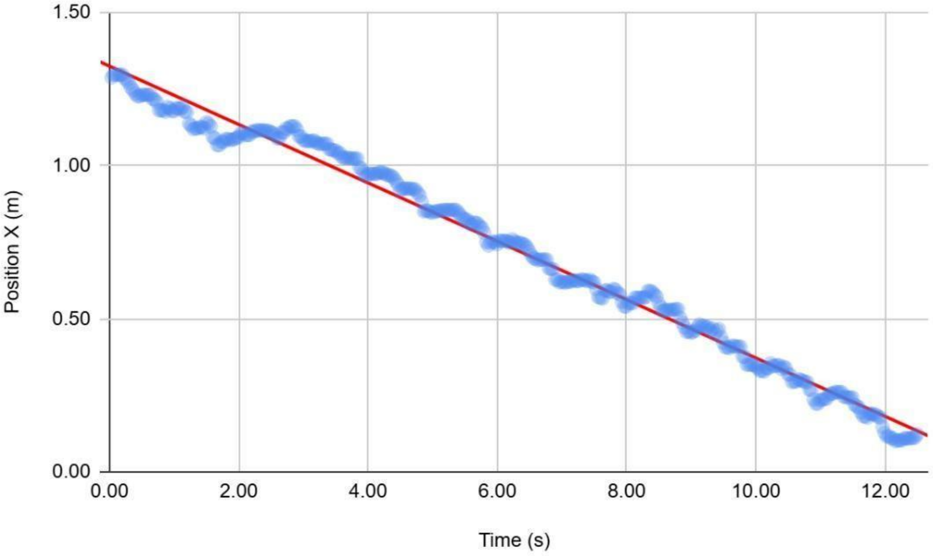
Figure 3: Time VS Speed “Forward”.



Source: own.

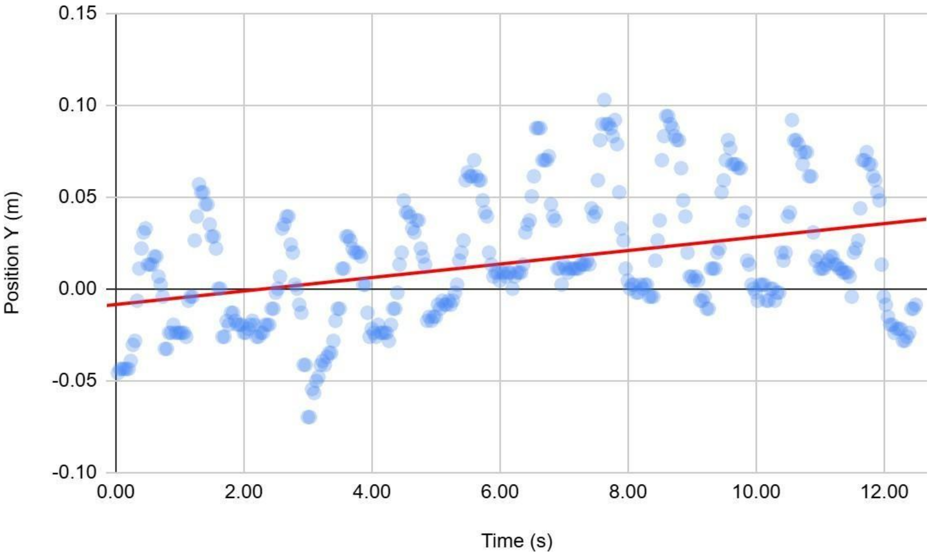
ii. Backward

Figure 4: Time VS Position X "Backward".



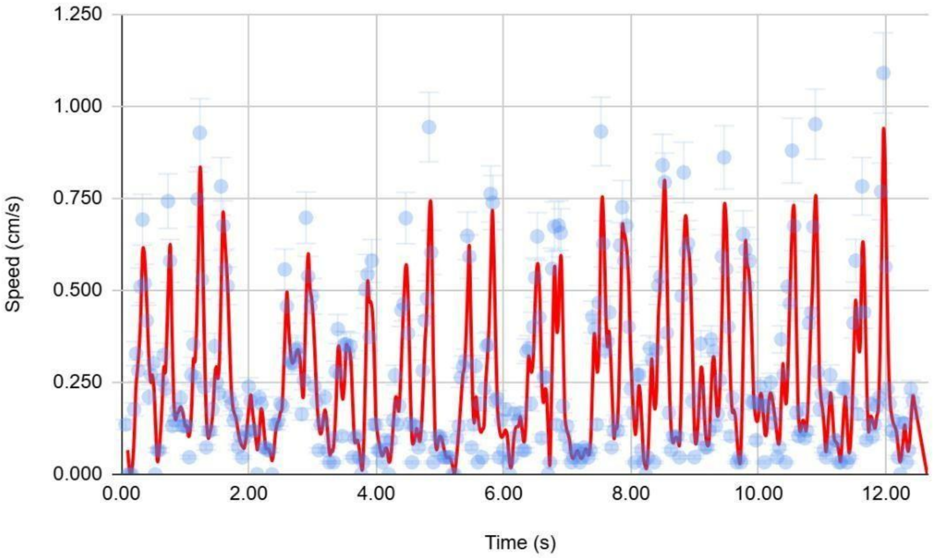
Source: own.

Figure 5: Time VS Position Y "Backward".



Source: own.

Figure c: Time VS Speed "Backward".



Source: own.

d. Gait Efficiency

Table 3: Gait Efficiency (walking pattern)

Movement	**Frequency (Hz)**	**Speed (cm/s)**	**Average current (mA)**	**Stability (Visual)**
Forward	0.75	75.2	520	Tilt to one side
Backward	0.75	74.6	580	Tilt to one side
Right	0.9	20(rotation)	460	Stable

Left	0.94	20.3(rotation)	470	Stable
Dance	1.5	0.0(fixed)	800	Stable
Greet	0.12	0.0(fixed)	650	Stable

Source: own.

4. Methodology

The development of our quadruped robot commenced with a prototyping phase, drawing inspiration from the design and fabrication notes documented on the blog Burari Web [28]. This blog details the construction of a four-legged walking robot utilizing cost-effective components and 3D printing techniques. The material served as a valuable visual reference for defining both the mechanical design and the functional structure of our model. Building upon this initial concept, we proposed the fabrication of robots through 3D printing, a process executed within our university laboratories. This approach allowed us to leverage available resources and adapt the design to meet our specific technical requirements.

To manage the system, we opted for a Raspberry Pi Pico W, initially programmed using MicroPython, due to its low power consumption, wireless connectivity, and ease of programming. For the locomotion system, SG90 micro servomotors were employed in each joint of the robot, selected for their availability, cost-effectiveness, and appropriate dimensions for small-scale projects.

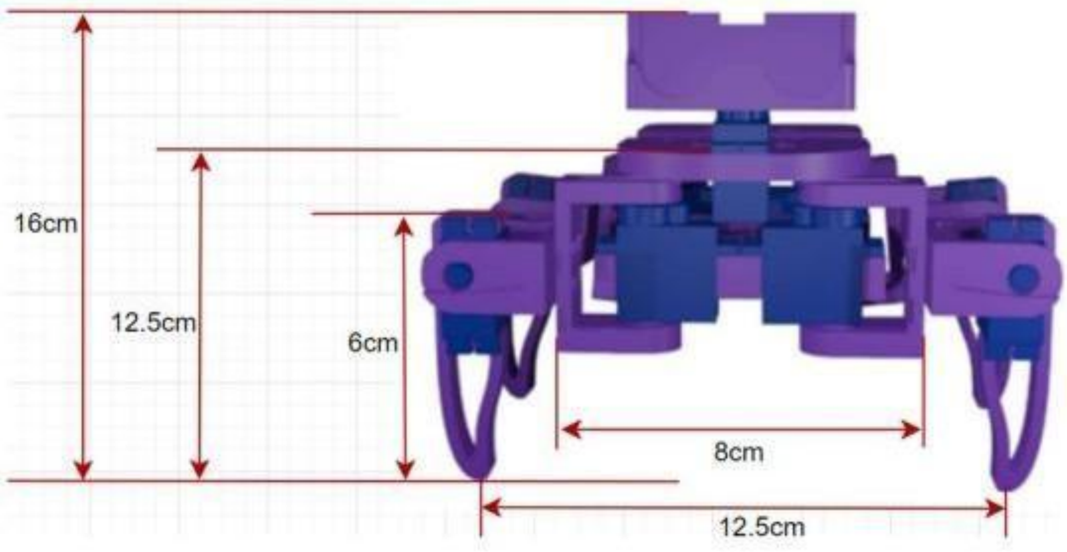
During the initial testing phases, we encountered several technical challenges. A primary issue was the frequent disconnection of the servo cables, attributed to abrupt movements and the mechanical stress induced by the robot's operation. Furthermore, we identified voltage fluctuations and inadequacies in the power supply, which adversely impacted the system's performance. To mitigate these issues, we replaced the cables with those of a heavier gauge, facilitating more stable current conduction and enhancing the overall system response. In light of these observations, we propose the integration of a PCA9685 servo controller, which enabled the concurrent management of multiple motors and markedly enhanced control precision. This modification also necessitated the transition of our programming environment from MicroPython to CircuitPython, facilitating the development of more modular, readable, and efficient code, thereby optimizing the robot's performance.

This project, currently under development, has evolved into a comprehensive educational endeavor, encompassing the phases of design, manufacturing, system programming, and optimization. Beyond its educational significance, the prototype serves as a versatile platform for future testing on challenging terrains, with potential applications in academic research, as well as in cost-effective robotic exploration scenarios.

5. Discussion

The prototype, designated as Moradito, exhibits a compact design, measuring approximately 16 cm in height and 14 cm in width, as illustrated in Figure 7. It incorporates eight SG90 servomotors distributed across its four limbs, thereby conferring two degrees of freedom to each leg.

Figure 7: Robot measurements.

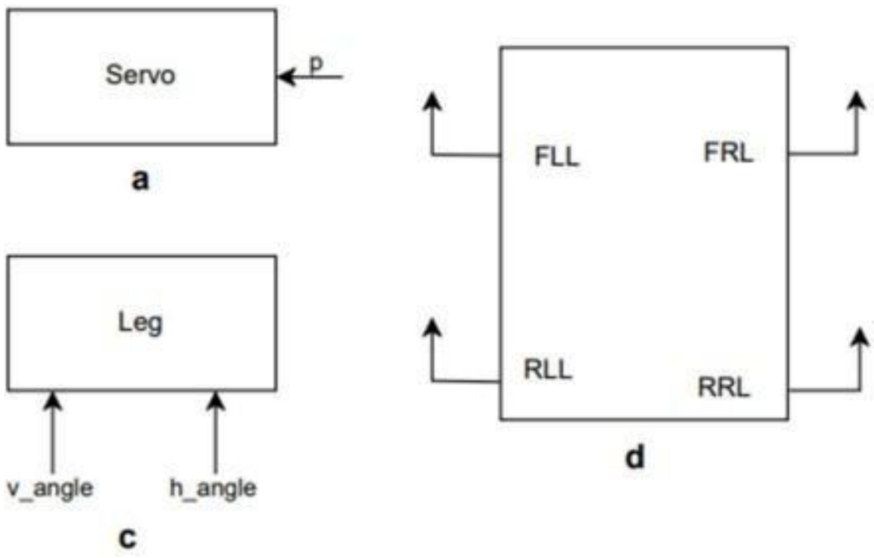


Source: own.

This configuration facilitates the execution of fundamental movements, including forward and backward motion, lateral turning, and the execution of preprogrammed sequences such as "Dance" or "Greet." During experimental testing, several technical achievements were identified: the robot demonstrated functionality under remote control via a web interface developed in HTML, exhibited stable gait patterns on flat surfaces, and achieved effective displacements at speeds of up to 8.19 cm/s. Furthermore, the object-oriented programmingbased [29] control system enabled a modular and scalable architecture.

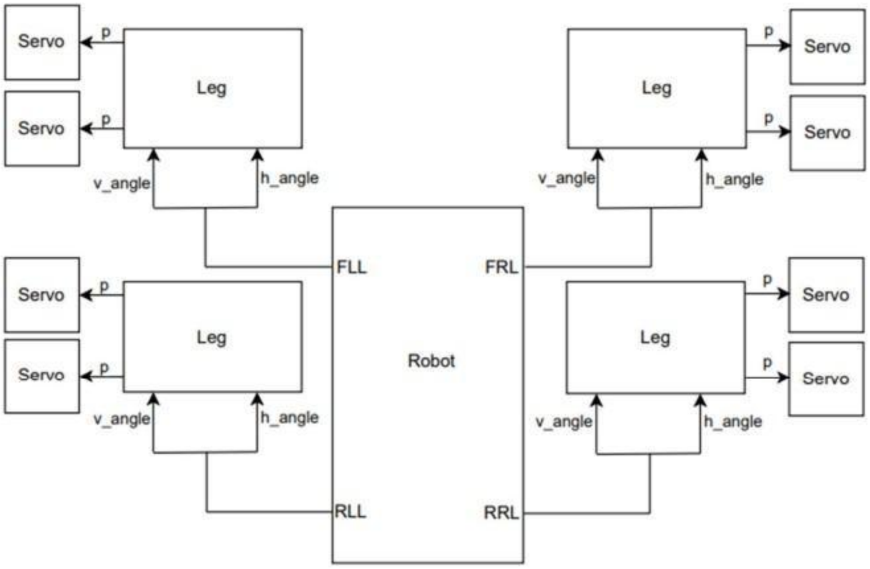
Nevertheless, several significant limitations were identified. A discernible tilt was observed during forward and backward movements, which compromises stability on more challenging terrain. Additionally, the SG90 motors demonstrated torque limitations, particularly during extended movements or when subjected to additional load, as exemplified by the "greet" function. The lack of proximity or navigation sensors further constrains the potential for autonomy.

Figure 8: Blocks representing the CircuitPython modules.



Source: own.

Figure S: Instantiated classes and modules.



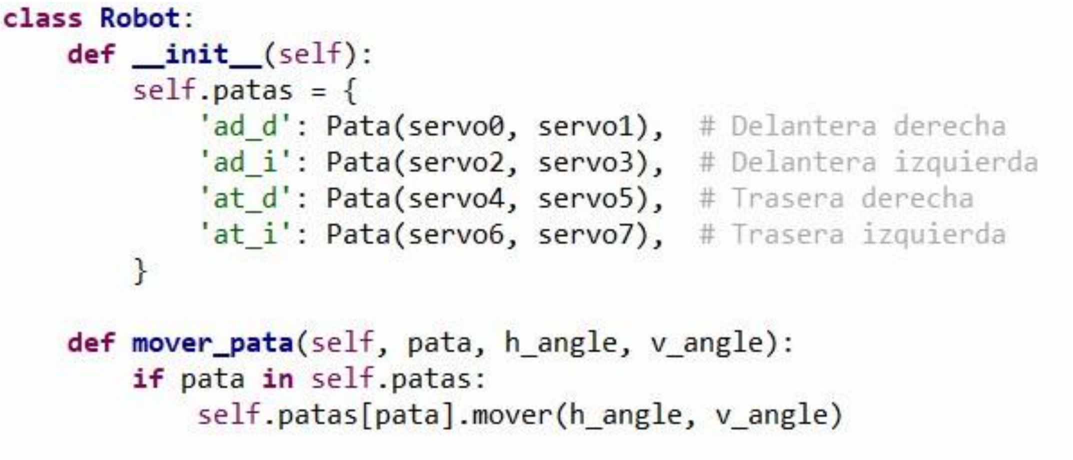
Source: own.

The findings indicate that this type of platform could be feasible for application in real-world agricultural settings, contingent upon the optimization of factors such as actuator torque, sensor integration (LIDAR), and enhancements in the gait algorithm.

Figure 8 presents a series of blocks, each representing a CircuitPython module. Specifically, the block labeled as "servo" (Figure 8a) denotes a fundamental module designed to control the position of a single servo. This module includes a class named "Servo," which serves as a basic class for positioning a servo at a specified angle. Consequently, this class incorporates a method, referred to in our code as "setServo(p)," where the parameter "p" specifies the angle at which the servo is to be positioned. According to the block diagram, the angle parameter is externally provided to its corresponding block, in this case, from another CircuitPython module. In reference to the blocks in Figure 8, Figure 7b depicts a block identified as "Leg," which contains a class named "Leg." This class executes two methods, termed "h_servo(h_angle)" and "v_servo(v_angle)," respectively. The mnemonics "h_servo" and "v_servo" denote the horizontal and vertical servos at each robot leg, while "h_angle" and "v_angle" represent the desired angles for positioning the servos.

Figures 10 and 11 depict the internal software architecture implemented in CircuitPython for the purpose of controlling the quadruped robot's movements. Figure 10 illustrates the implementation of the Robot class, which functions as a controller overseeing the movement of each of the robot's four legs. Each leg is designated by a unique identifier: 'ad_d' (Front Right Leg), 'ad_i' (Front Left Leg), 'at_d' (Rear Right Leg), and 'at_i' (Rear Left Leg). These identifiers correspond to instances of the Pata class (Spanish for “leg”), with each instance receiving two servos (horizontal and vertical) as parameters. The method mover_pata() accepts a leg identifier and target angles for both axes, subsequently delegating the command to the corresponding leg instance.

Figure 10: Instance class Robot.

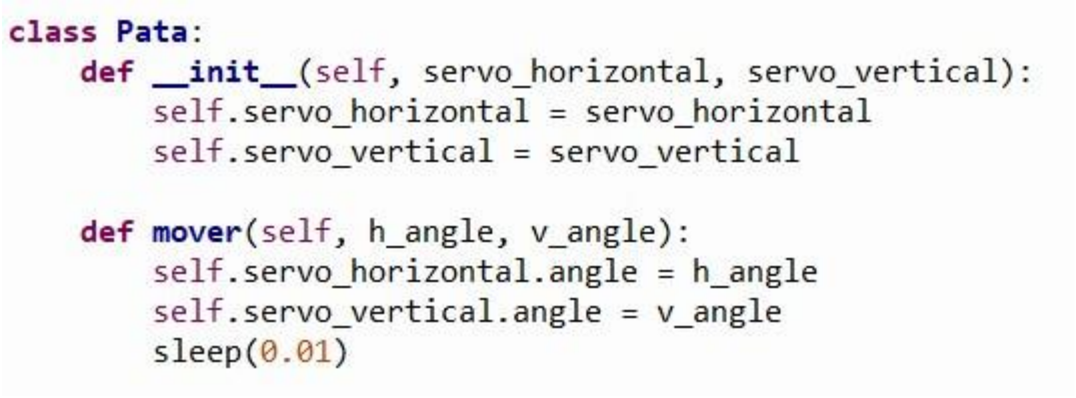


```python
class Robot:
    def __init__(self):
        self.patas = {
            'ad_d': Pata(servo0, servo1),  # Delantera derecha
            'ad_i': Pata(servo2, servo3),  # Delantera izquierda
            'at_d': Pata(servo4, servo5),  # Trasera derecha
            'at_i': Pata(servo6, servo7),  # Trasera izquierda
        }

    def mover_pata(self, pata, h_angle, v_angle):
        if pata in self.patas:
            self.patas[pata].mover(h_angle, v_angle)
```

Source: own.

Figure 11: Instance class Pata.



```python
class Pata:
    def __init__(self, servo_horizontal, servo_vertical):
        self.servo_horizontal = servo_horizontal
        self.servo_vertical = servo_vertical

    def mover(self, h_angle, v_angle):
        self.servo_horizontal.angle = h_angle
        self.servo_vertical.angle = v_angle
        sleep(0.01)
```

Source: own.

Figure 11 illustrates the implementation of the Pata class, which is designed to manage the control of the two servos associated with a single leg. The method mover() accepts horizontal (h_angle) and vertical (v_angle) angles as parameters, thereby directly updating the position of each servo. A brief delay (sleep(0.01)) is incorporated to ensure smooth and stable transitions between sequential commands.

The two classes constitute the fundamental logic underlying the coordinated control of the robot's legs, effectively abstracting the lower-level servo control mechanisms. By modularizing each leg as an independent entity, the Robot class is capable of managing intricate walking gaits and motion patterns in a scalable and systematically organized manner.

Figure 8 illustrates the vertical and horizontal servos. In Figure 8c, an illustration of the robot block is presented. This module includes a class named Robot, which executes methods that reutilize those from the Leg class to control the movements of the four prototype legs. Within this class, two methods are implemented to manage each leg of the robot. The legs in the Robot class are designated by the following mnemonics: FLL (Frontal Left Leg), RLL (Rear Left Leg), FRL (Frontal Right Leg), and RRL (Rear Right Leg). Correspondingly, the Leg class is instantiated four times, as depicted in Figure 9, along with the parameters h_angle and v_angle. According to Figure 9, the Servo class is instantiated eight times.

a. Quantitative analysis

From the collected data, it is observed that the "Greet" movement presents the highest energy consumption (2.92 mAh), which is consistent with its long duration (16.5 s) and the active use of the neck. In contrast, the "Dance" movement presents a high instantly consumption (800 mA) in a short interval (0.69 s), indicating a high instantaneous current requirement.

The velocity of the backward movement, measured at 8.19 cm/s, surpasses that of the forward movement, which is recorded at 7.08 cm/s. This discrepancy may be attributed to the support pattern and retraction force, as illustrated in Figures 1 through 6.

In conclusion, it is noteworthy that lateral movements (right and left) exhibit low velocity yet high visual stability, which may be advantageous for precision maneuvers. This observation is corroborated by the gait efficiency data (Table 3), indicating that these movements necessitate less current (460–470 mA) and demonstrate stable patterns.

6. Conclusions

The present study details the successful development of a quadruped mobile robot prototype, demonstrating its efficacy in traversing various simulated terrains. By employing a straightforward locomotion system utilizing servomotors programmed in CircuitPython, the robot achieved coordinated "trotting" movements across its legs, thereby maintaining both stability and continuous motion on uneven surfaces. This outcome substantiates the viability of such a system in environments where wheeled and differential drive systems encounter significant limitations.

One of the principal contributions of this project is its cost-effectiveness, achieved through the integration of inexpensive electronic components, 3D printing, and a modular, replicable control architecture. This renders it an appealing solution for the agricultural sector, particularly in rural environments where resources for implementing advanced automation are limited.

In contrast to high-budget solutions designed for industrial applications, this prototype seeks to fulfill practical requirements such as inspection, monitoring, and support for light logistics, without necessitating complex or expensive infrastructure.

The robot's design incorporated both mechanical modeling and the development of the locomotion system from the ground up to achieve straightforward, functional, and timebased servo activation sequences. While aspects such as autonomous navigation and energy autonomy have not yet been addressed, the system establishes a robust foundation for future advancements.

In this context, several opportunities for project expansion have been identified, including the integration of environmental perception sensors and the optimization of the power system through the use of solar panels or higher-capacity batteries. A recommended subsequent step involves validating the robot's behavior in real-world field scenarios—such as mud, slopes, vegetation, or unpredictable obstacles—as its performance under actual agricultural conditions has not yet been examined.

Acknowledgments

The authors sincerely thank the Colombian Ministry of Science, Technology, and Innovation (MINCIENCIAS) and the Francisco José de Caldas Fund for their support through the allocation of resources in Call 934 of 2023, contract 332-2023, project 99209. In compliance with the disclosure requirements of this contract, the authors refer to MINCIENCIAS and the Francisco José de Caldas Fund in this work. The authors also thank the LASER research group at the Francisco José de Caldas District University for their guidance and advice during the development of this research project. Special thanks go to BitPointer S.A.S., beneficiary of project 99209, in particular to its manager, Juan Fajardo Barrero, and its administrative director, Alexandra Reyes Agudelo, for their valuable contributions and commitment to this initiative.

References

[1] A. O. Hourani y M. Z. Iskandarani, “Design, Modelling, and Analysis of Legged Robot for Terrains Exploration”, Int. J. Adv. Sci., Eng. Inf. Technol., vol. 13, n. º 3, pp. 1127–1128, junio de 2023. https://doi.org/10.18517/ijaseit.13.3.19000

[2] S. Bazeille et al., “Quadruped robot trotting over irregular terrain assisted by stereo-vision”, Intell. Service Robot., vol. 7, n. º 2, pp. 67–69, marzo de 2014. https://doi.org/10.1007/s11370-014-0147-9

[3] A. O. Baturone, Robótica: Manipuladores y Robots Móviles. Marcombo, 2005.

[4] X. Gao et al., "Review of Wheeled Mobile Robots' Navigation Problems and Application Prospects in Agriculture," in IEEE Access, vol. 6, pp. 49248-49268, 2018, doi:10.1109/ACCESS.2018.2868848.

[5] Robotics at Home with Raspberry Pi Pico: Build autonomous robots with the versatile low-cost Raspberry Pi Pico controller and Python. Packt Publ., 2023.

[6] N. Pudchuen, C. Deelertpaiboon, W. Jitviriya and A. Phunopas, "VENRiR: Vision Enhance for Navigating 4-legged Robot in Rough Terrain," 2020 59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), Chiang Mai, Thailand, 2020, pp. 1410-1415, doi: 10.23919/SICE48898.2020.9240424.

[7] C. E. López Rodríguez and Y. L. Vargas Castiblanco, "Importancia de los procesos de automatización en el sector agrícola colombiano," ID EST – Revista Investigación, Desarrollo, Educación, Servicio y Trabajo, vol. 4, no. 2, p. 4, 2024. https://revista.fundes.edu.co/index.php/revista/article/view/264.

[8] Desarrollo de la agricultura colombiana. Fedesarrollo, 2014

[9] C. D. Bellicoso et al., "Advances in real-world applications for legged robots," Journal of Field Robotics, vol. 35, no. 8, pp. 1311–1326, Nov. 2018, doi: 10.1002/rob.21839.

[10] C. Quail, E. Emonot–de Carolis and F. Auat Cheein, "Legged Robots in the Agricultural Context: Analysing Their Traverse Capabilities and Performance," IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society, Singapore, Singapore, 2023, pp. 01-07, doi: 10.1109/IECON51785.2023.10312233.

[11] "Development of a quadruped robot platform for optimizing wheat and corn field for phenotyping". University of Minnesota driven to discover. https://conservancy.umn.edu/items/69f6a205-2789-4095-8c6d-df2e227bf462.com

[12] M. F. Silva y J. A. Tenreiro Machado, “A Historical Perspective of Legged Robots”, J. Vib. Control, vol. 13, n. º 9-10, pp. 1447–1455, septiembre de 2007. https://doi.org/10.1177/1077546307078276

[13] Q. Li, F. Cicirelli, A. Vinci, A. Guerrieri, W. Qi y G. Fortino, “Quadruped Robots: Bridging Mechanical Design, Control, and Applications”, Robotics, vol. 14, n. º 5, pp. 5–8, abril de 2025. https://doi.org/10.3390/robotics14050057

[14] M. Aguilera Hernández, M. Bautista y J. Iruegas, “Diseño y Control de Robots Móviles”, Asoc. Mex. Mecatronica AC., pp. 1–6, 2003.

[15] J. M. Robles Atuesta. “Diseño y prototipado del mecanismo de locomoción para un robot cuadrúpedo”. Universidad de los Andes Colombia. https://hdl.handle.net/1992/45035

[16] J. Fu y F. Gao, “Dynamic stability analyzes for a parallel–serial legged quadruped robot”, Int. J. Adv. Robotic Syst., vol. 19, n. º 5, pp. 3–5, septiembre de 2022. https://doi.org/10.1177/17298806221132081

[17] J. Li, J. Wang, S. X. Yang, K. Zhou y H. Tang, “Gait Planning and Stability Control of a Quadruped Robot”, Comput. Intell. Neurosci., vol. 2016, pp. 1–13, 2016. https://doi.org/10.1155/2016/9853070

[18] Q. Cong et al., “Stability Study and Simulation of Quadruped Robots with Variable Parameters”, Appl. Bionics Biomechanics, vol. 2022, pp. 1–9, enero de 2022. https://doi.org/10.1155/2022/9968042

[19] P.-B. Wieber, R. Tedrake y S. Kuindersma, “Modeling and Control of Legged Robots,” en Springer Handbook of Robotics, 2ª ed., B. Siciliano y O. Khatib, eds., Springer, 2016, pp. 1213–1223. https://doi.org/10.1007/978-3-319-32552-1_48.

[20] R. Zwetsloot. “Control servos with CircuitPython and Raspberry Pi”. Raspberry Pi Official Magazine. https://magazine.raspberrypi.com/articles/control-servoscircuitpython-raspberry-pi?.com

[21] C. Bell, Beginning MicroPython with the Raspberry Pi Pico. Berkeley, CA: Apress, 2022. https://doi.org/10.1007/978-1-4842-8135-2

[22] Automacion, “Servomotores: control, precisión y velocidad”, AADECA, vol. 4, p. 2, 2017.

[23] J. Kim, T. Kang, D. Song y S.-J. Yi, “Design and Control of a Open-Source, Low Cost, 3D Printed Dynamic Quadruped Robot”, Appl. Sci., vol. 11, n. º 9, p. 3762, abril de 2021. Accedido el 15 de julio de 2025. https://doi.org/10.3390/app11093762

[24] E. Garcia, P. González-de-Santos y J. Estremera, Quadrupedal Locomotion: An Introduction to the Control of Four-legged Robots. Springer, 2006.

[25] Z. Gacovski, Ed., Mobile Robots – Current Trends InTech, 2011. https://doi.org/10.5772/2305

[26] G. Raghavendra, B. B. V. L. Deepak y M. Gupta, Eds., Recent Advances in Mechanical Engineering, Volume 1. Singapore: Springer Nature Singap., 2024. https://doi.org/10.1007/978-981-97-0918-2

[27] Z. Y. Du y B. Liu, Advanced Mechanical Engineering. Trans Tech Publ. LTD, 2010. https://doi.org/10.4028/b-v26glh

[28] 東京バード. (2021, 12 de julio). *Arduino で動かす 4 足歩行ロボット製作ノート！Arduino 学習に便利なロボくんなので使って下さい！【STL データ公開】*. ぶらり@web 走り書き. https://burariweb.info/electronic-work/4legged-waking-robot-production-notes.html

[29] M. Stefik and D. G. Bobrow, “Object-Oriented Programming: Themes and Variations”, *AIMag*, vol. 6, no. 4, p. 40, Dec. 1985. https://doi.org/10.1609/aimag.v6i4.508