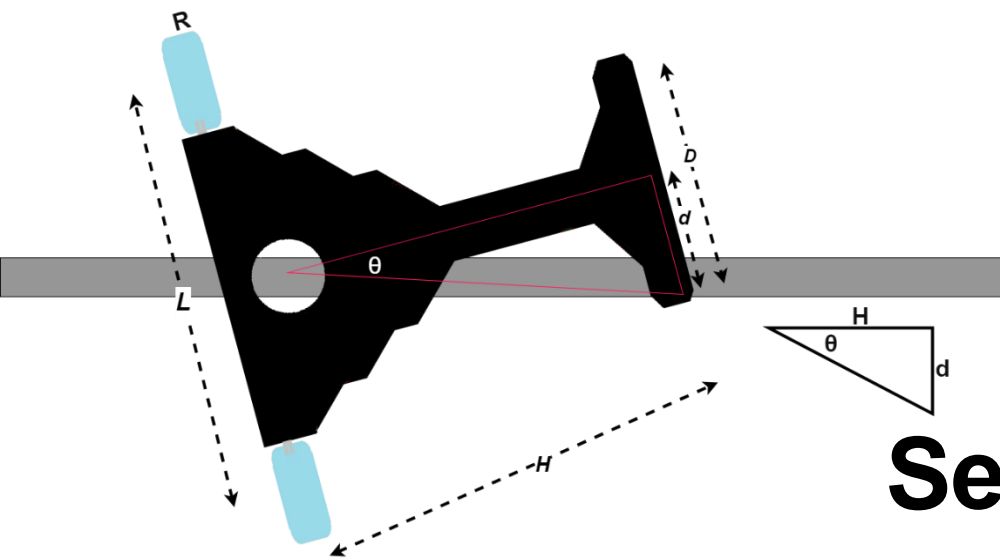


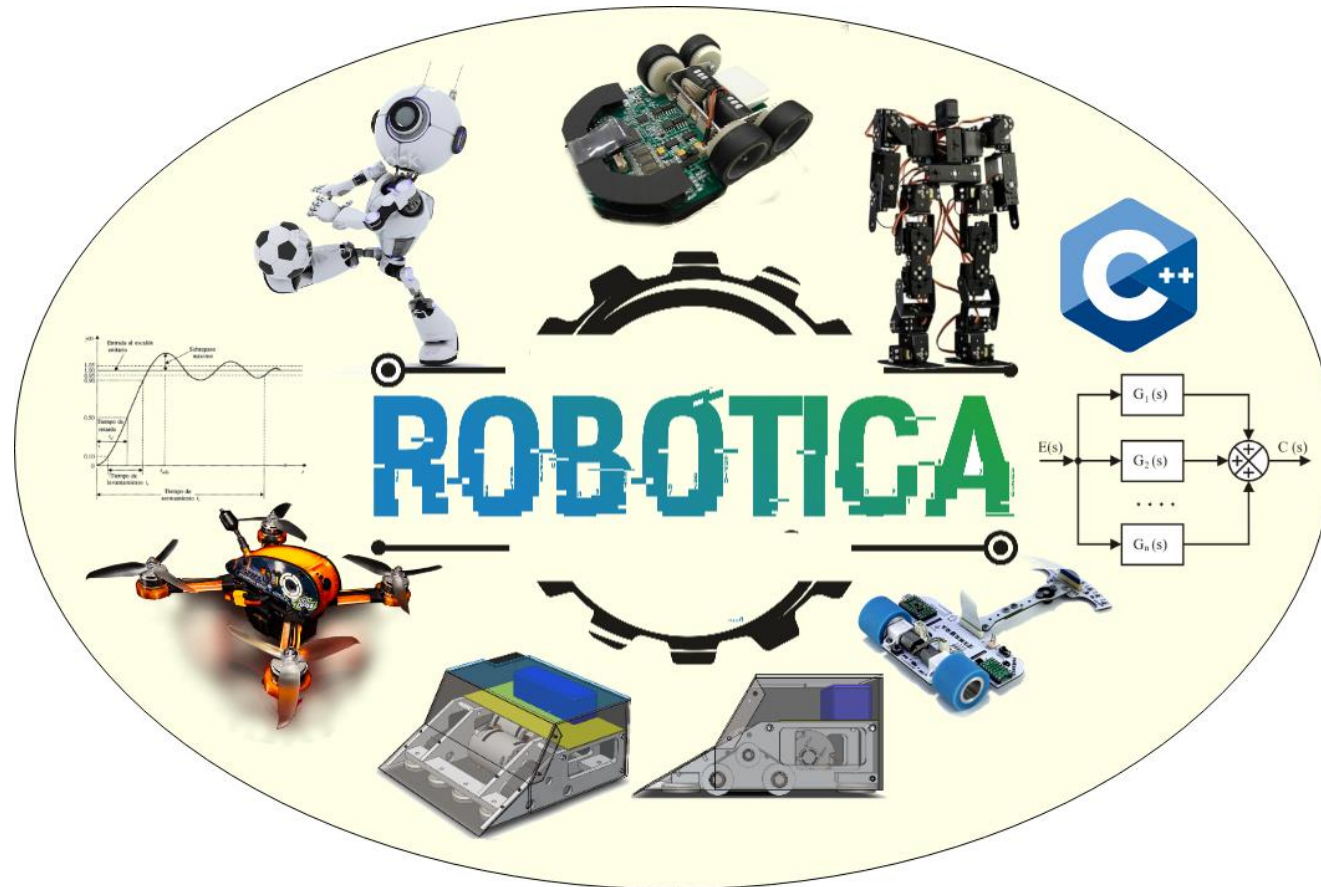
DISEÑO E IMPLEMENTACIÓN DE UN ROBOT SEGUIDOR DE LÍNEA DE COMPETENCIA PARA LA CATEGORÍA VELOCISTA

Autor: Yeison D TAPIERO
Director: Harold F MURCIA

Semillero de Investigación: SIRUI
Fecha: 15 de Julio de 2019



Introducción



La robótica móvil es una de las ramas de la ingeniería que mas avances ha presentado en las últimas décadas. Nuevas configuraciones mecánicas, estructuras de control y navegación, desarrollos en vehículos no tripulados y nuevas aplicaciones han cambiado la forma en que conocíamos los sistemas robóticos.

1

Planteamiento del Problema

2

Objetivos

3

Metodología

4

Desarrollo

5

Resultados

6

**Conclusiones y Trabajo
Futuro**

Planteamiento del Problema



Figura 1.1 Principales torneos naciones de robótica.

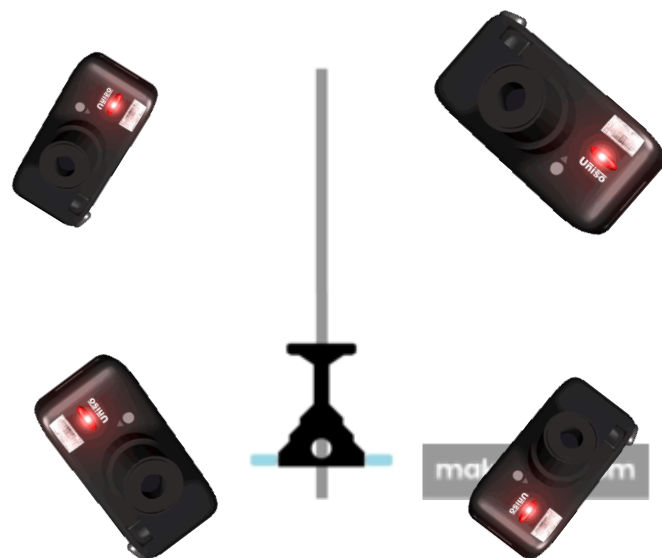
Fuente: Tomada de Internet.



Figura 1.2 Noticias destacadas en robotica. **Fuente:**

Tomada de Internet.

Planteamiento del Problema



Video 1 LFR en la industria. **Fuente:** Autor.

Planteamiento del Problema

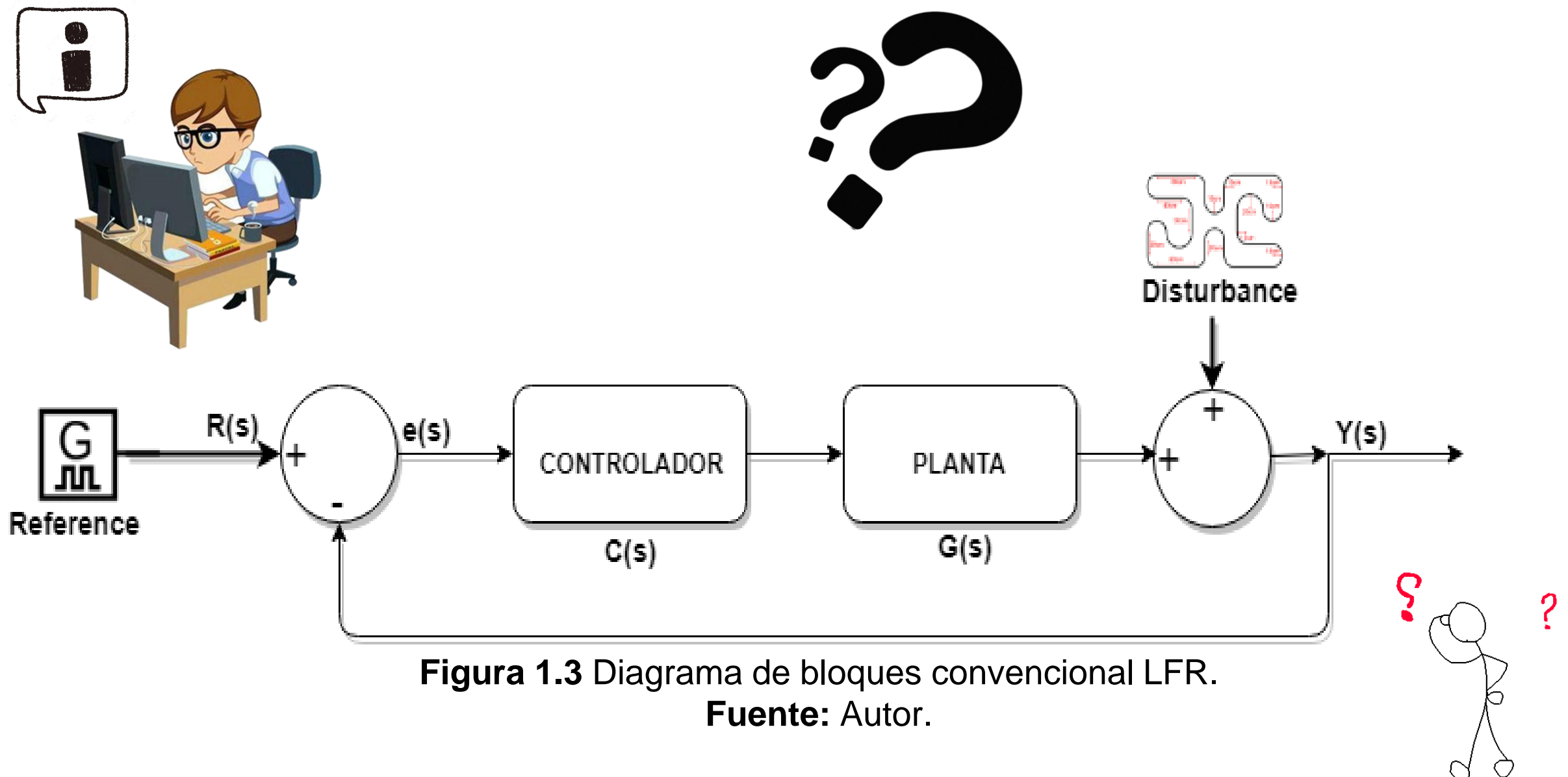


Figura 1.3 Diagrama de bloques convencional LFR.

Fuente: Autor.

¿Cuál es la mejor estrategia de control para un robot seguidor de línea velocista en topología diferencial?

Objetivos

Objetivo General

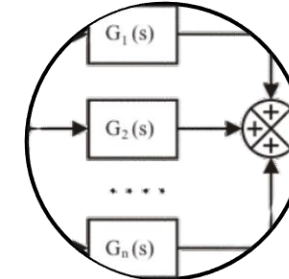
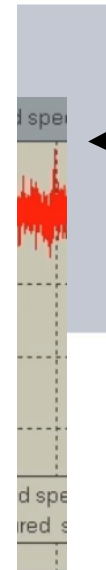
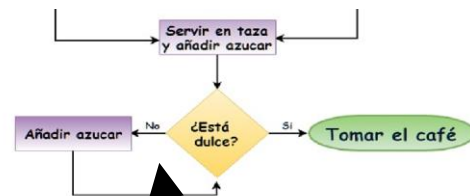
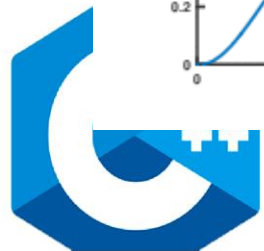
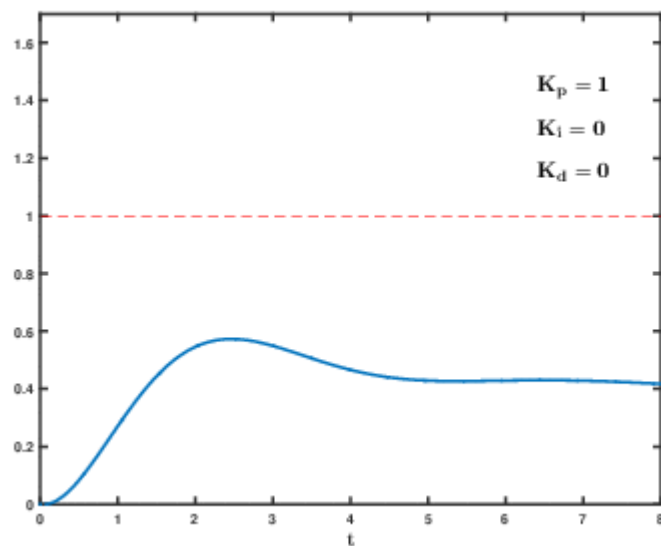
Esta propuesta de proyecto tiene como principal objetivo diseñar, construir y modelar un robot Seguidor de línea para la categoría velocista, con características técnicas de nivel avanzado, que cumplan con los estándares de las competencias internacionales.

Objetivos Específicos

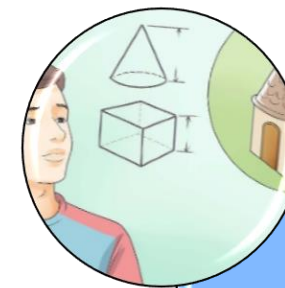
- Diseñar e implementar las capas mecánica y electrónica de un (1) robot seguidor de línea.
- Obtener un modelo matemático simple que describa la cinemática y dinámica de un robot seguidor de línea.
- Diseñar, implementar y comparar tres (3) estrategias de control para establecer el robot seguidor de línea en la trayectoria de referencia. .



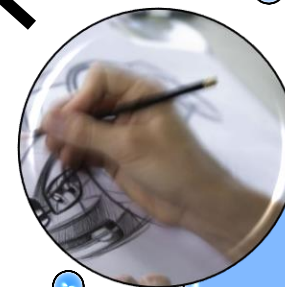
Metodología



Implementación
de estrategias de
Control



Identificación y
Validación del
Modelo



Implementación



Selección de
materiales.



Búsqueda de
Información

Desarrollo

- Forma, peso y tamaño máximo de 25x25 cm.
- Material del chasis.
- Distribución de peso.
- Rápida identificación del entorno.
- Tiempos cortos de procesamiento.
- Autonomía mayor a 3 minutos.

Criterios y Restricciones

- Microcontrolador maestro: Mbed NXP LPC1768 32-bit.
- Microcontrolador esclavo: Atmega 328p 8-bit.
- Arreglo de sensores QTR-8A.
- LSM6DS33 MiniIMU-9 v5
- Driver de motores DVR8833
- Motores pololu 10:1 12v 3000rpm con encoder magnético de 12p/r
- Bluetooth Mate Silver de Sparkfun
- Pololu 12V Step-Up
- Turbina EDF27 con motor 11000 kV y ESC de 6ª
- Batería LiPo 7.4v 500mh

Dispositivos Electrónicos

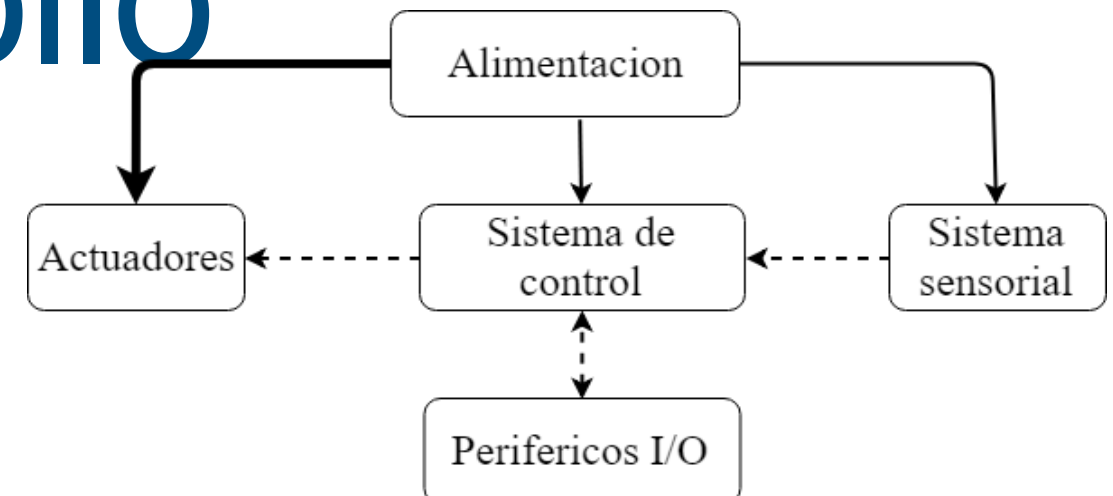


Figura 4.1 Diagrama general del sistema en hardware

Fuente: Autor.

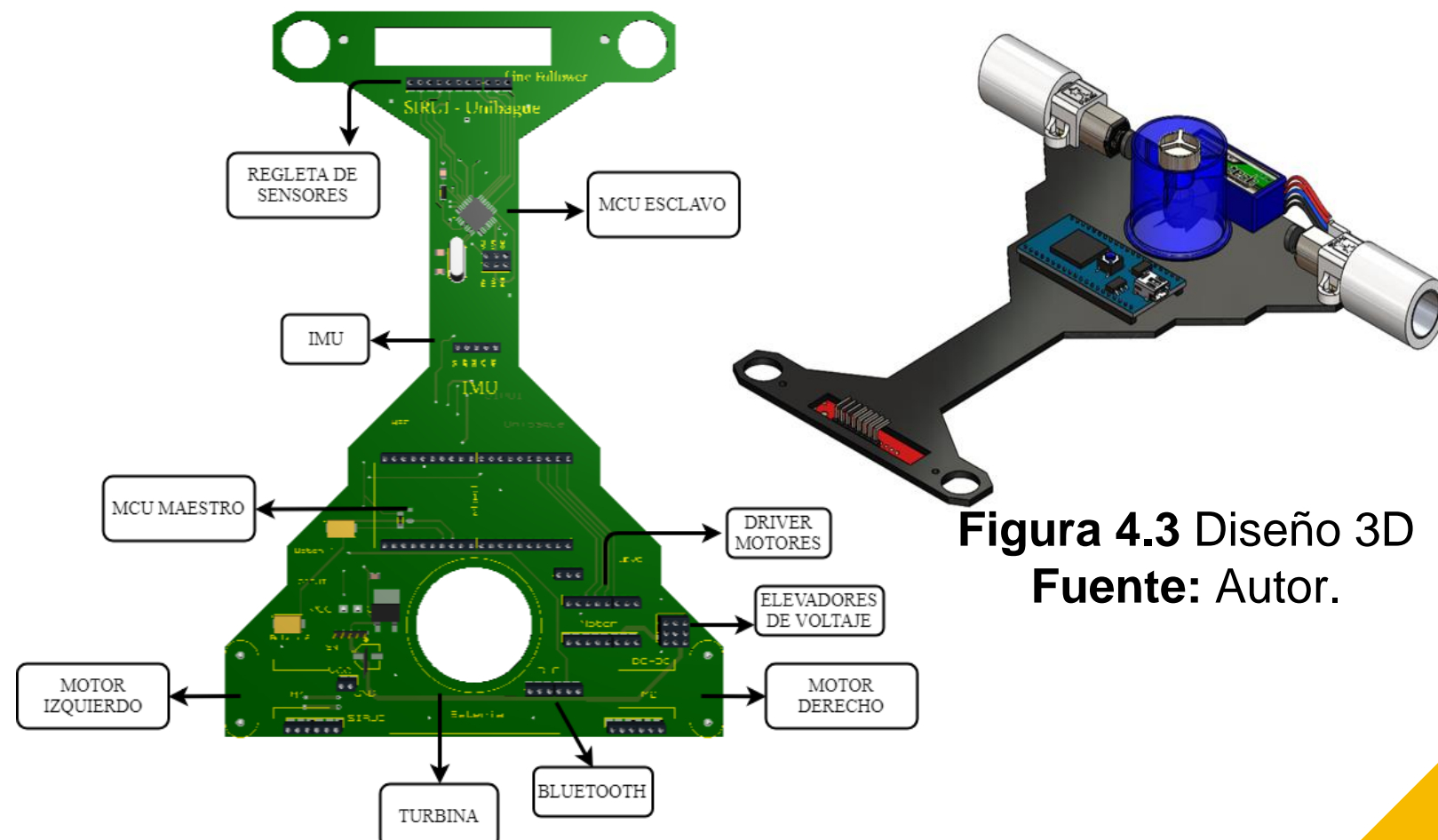


Figura 4.3 Diseño 3D

Fuente: Autor.

Figura 4.2 Simulación PCB prototipo Fuente: Autor.

PERCEPCIÓN

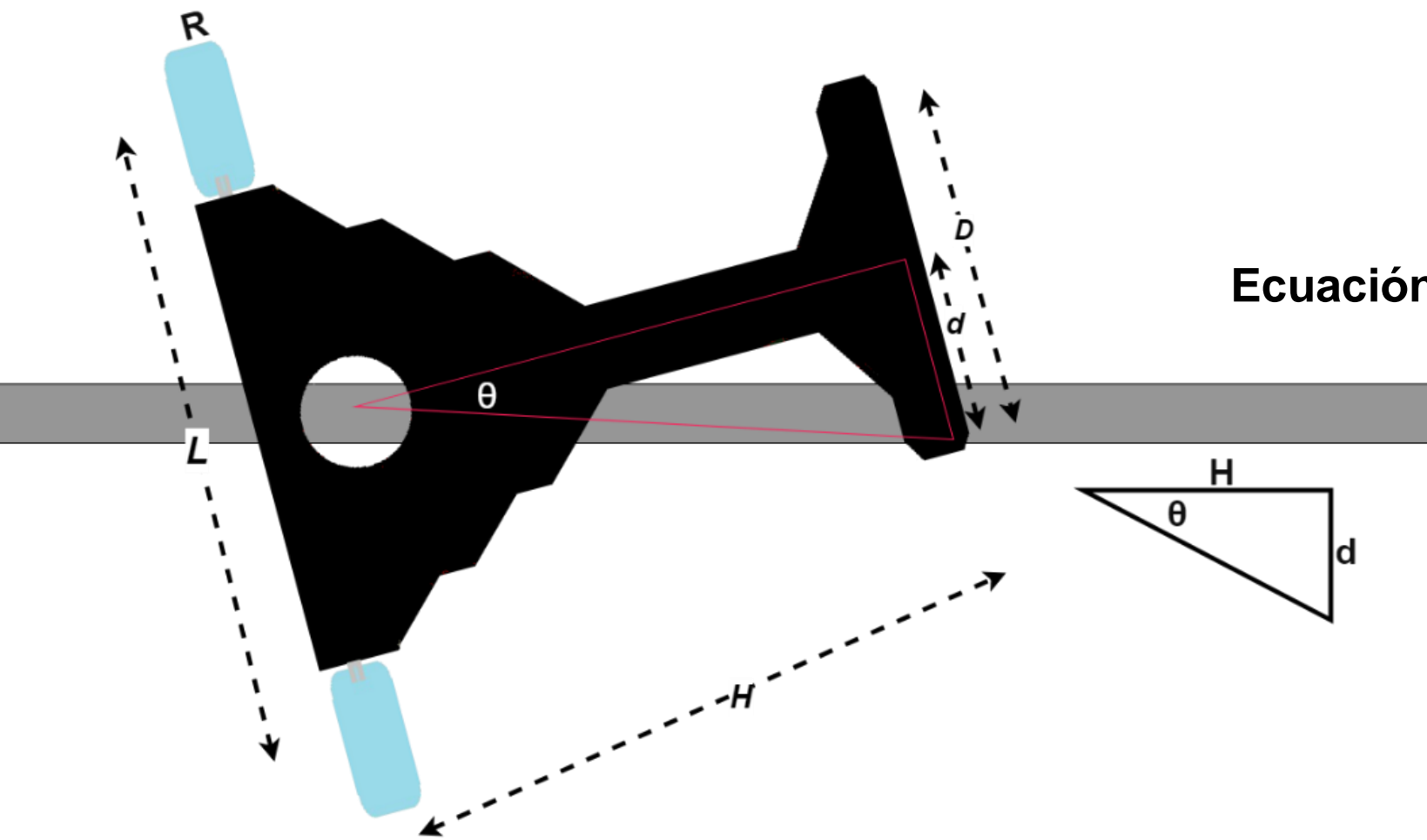


Figura 4.4 Análisis geométrico de un LFR
Fuente: Autor.

$$S = \frac{m - m_{\min}}{m_{\max} - m_{\min}} * 1000$$

Ecuación 1. Normalización de valores proveniente de un sensor IR.

$$D_{S[p]} = \frac{\sum_{i=1}^n S(p_i) p_i}{\sum_{i=1}^n S(p_i)}$$

Ecuación 2. Cálculo de posición con respecto a línea.

PERCEPCIÓN

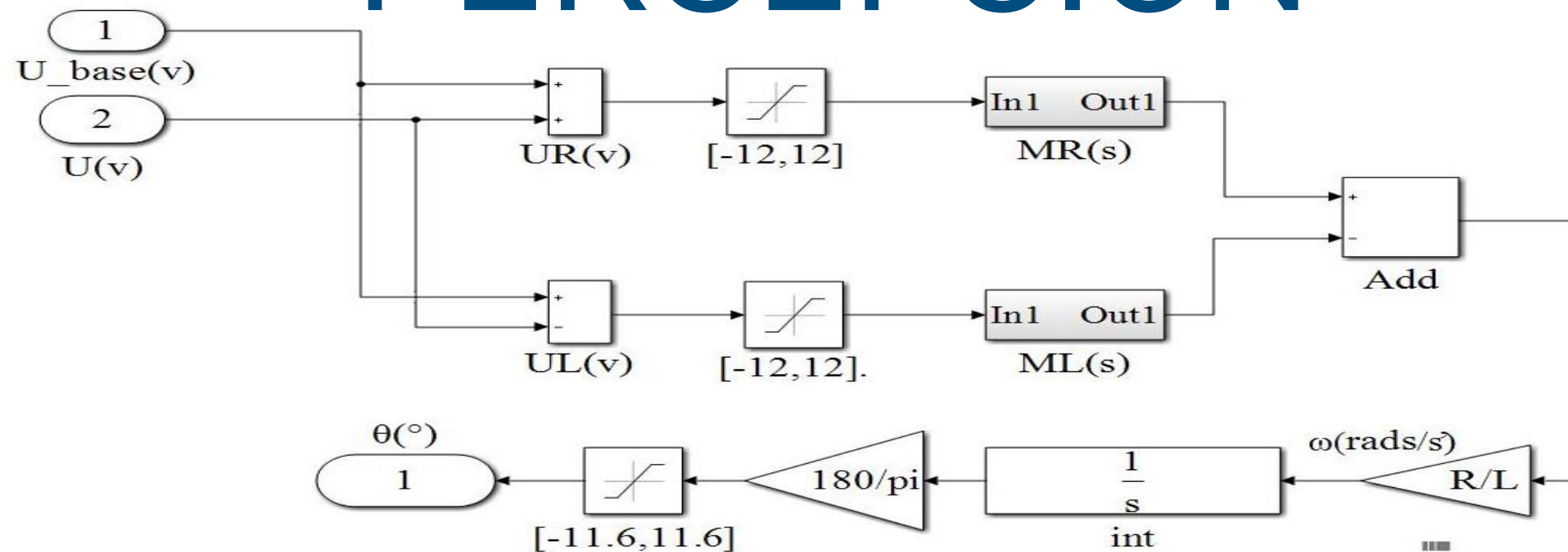


Figura 4.5 Representación de un robot móvil en configuración diferencial **Fuente:** Autor.

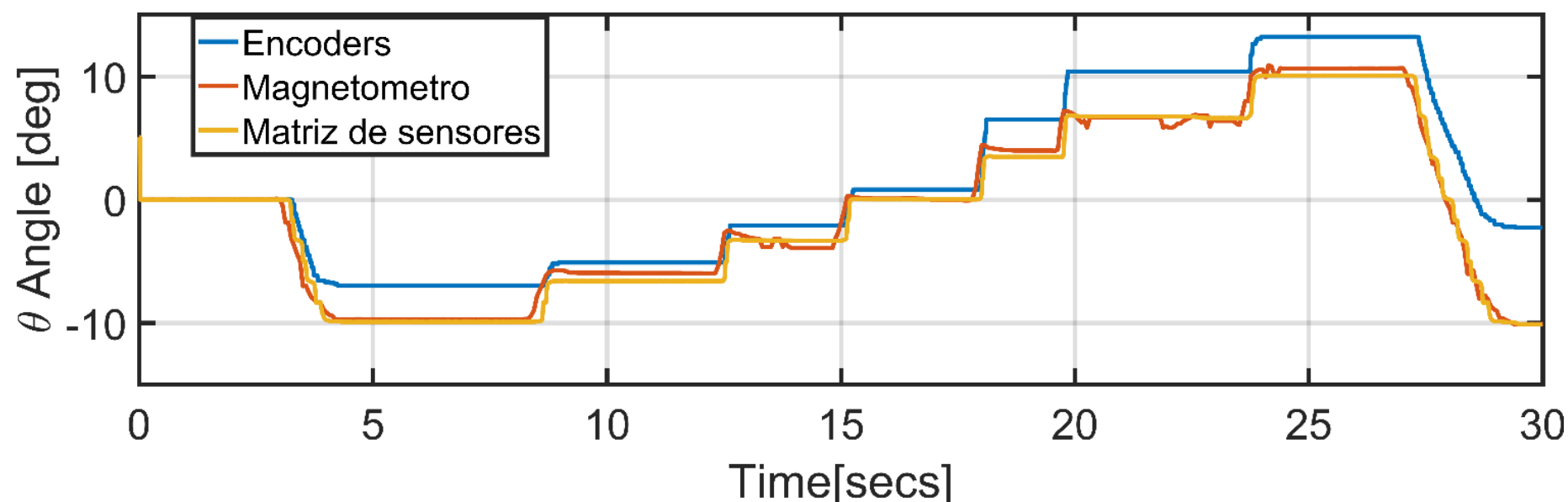


Figura 4.6 Angulo θ a partir de diferentes sensores.
Fuente: Autor.

ESTRATEGIAS DE CONTROL

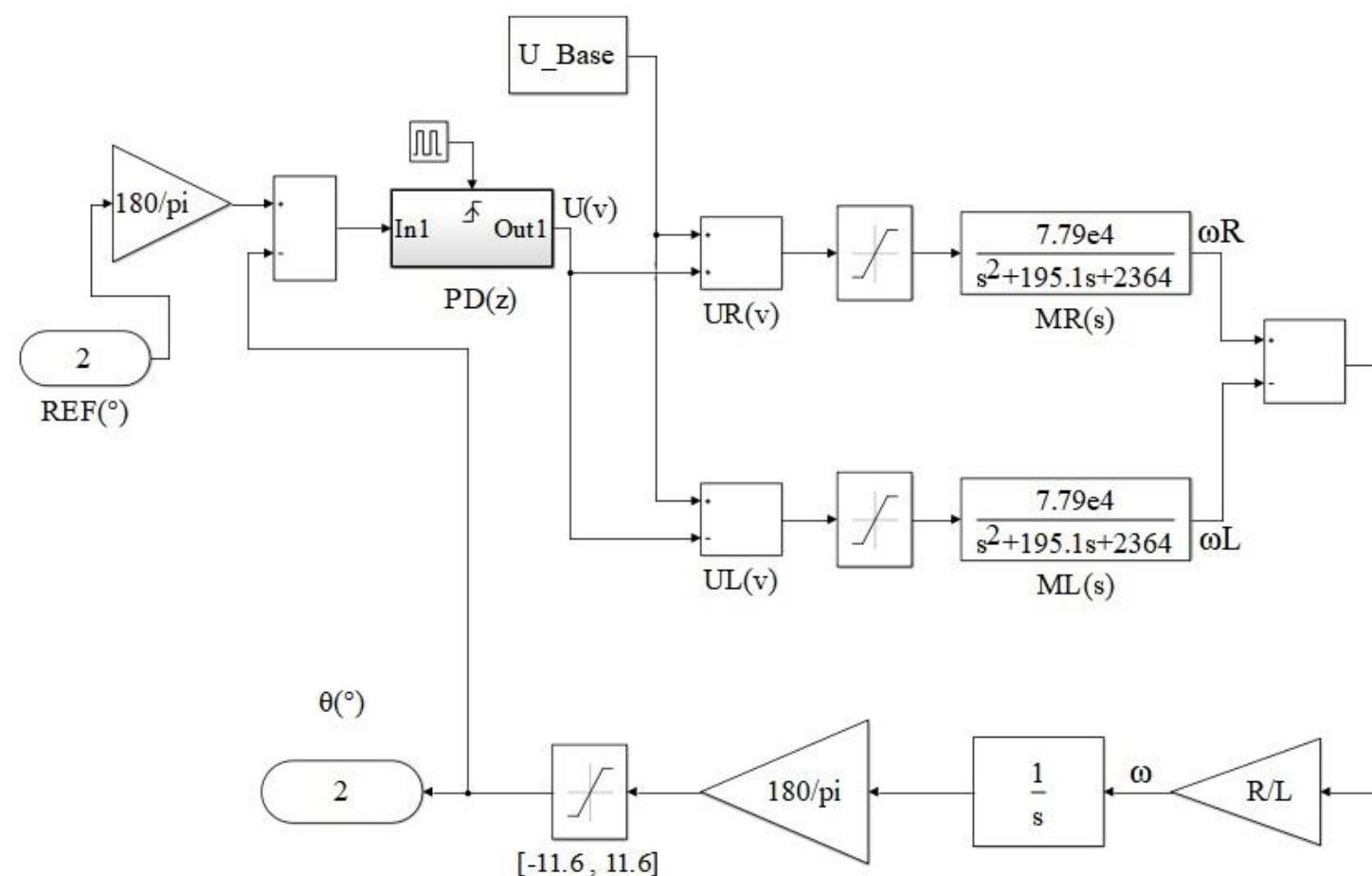


Figura 4.7 Diagrama de bloques estrategia PID convencional **Fuente:** Autor.

Algorithm 1. Implementation of algorithm to guarantee

```

    ▷ %comment: Fin
    % Calculate U[k] from PD controller
    U_base = U_Bref
    U_R[k] = U_base + U[k]/2
    U_L[k] = U_base - U[k]/2
    if (U_R[k] > U_Lim) or (U_R[k] < -U_Lim) then
        Δ1 = U_Lim - U_R[k]
        U_base = U_base + Δ1
    end if
    if (U_L[k] > U_Lim) or (U_L[k] < -U_Lim) then
        Δ2 = U_Lim - U_L[k]
        U_base = U_base + Δ2
    end if
    % Update motor actions
    U_R[k] = U_base + U[k]/2
    U_L[k] = U_base - U[k]/2
    
```

Figura 4.9 Seudocódigo algoritmo de garantía de control **Fuente:** [1].

ESTRATEGIAS DE CONTROL

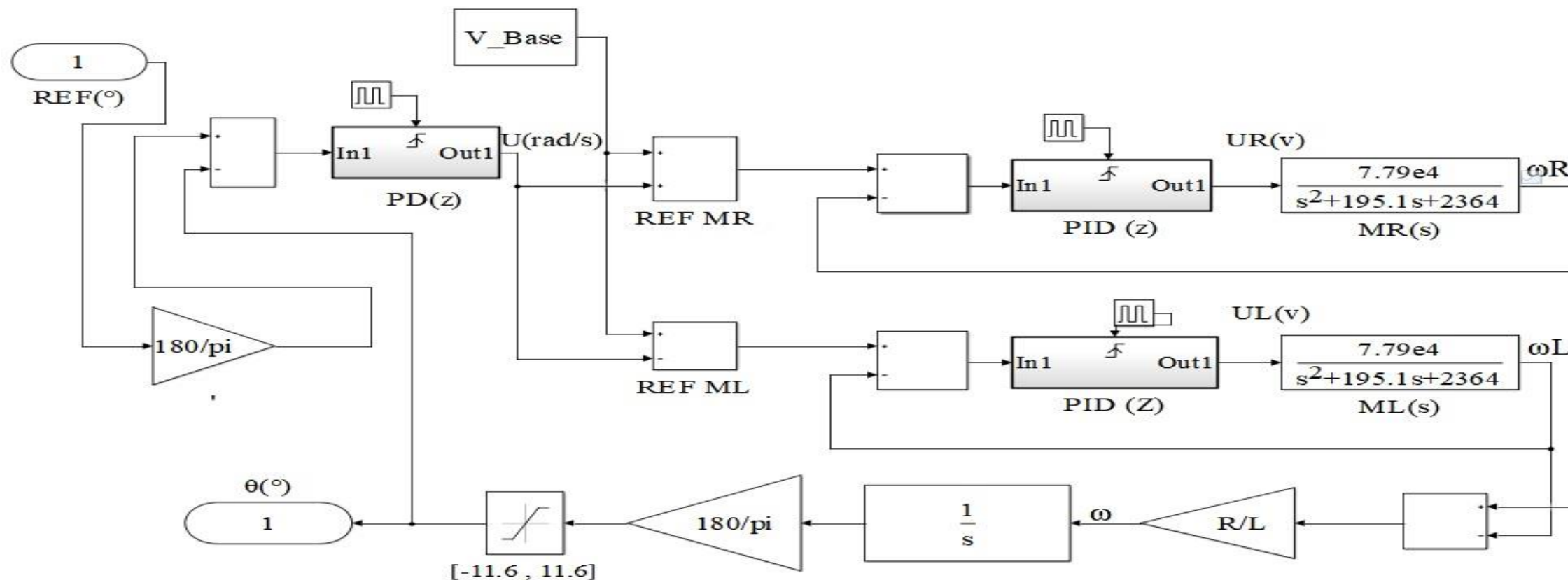
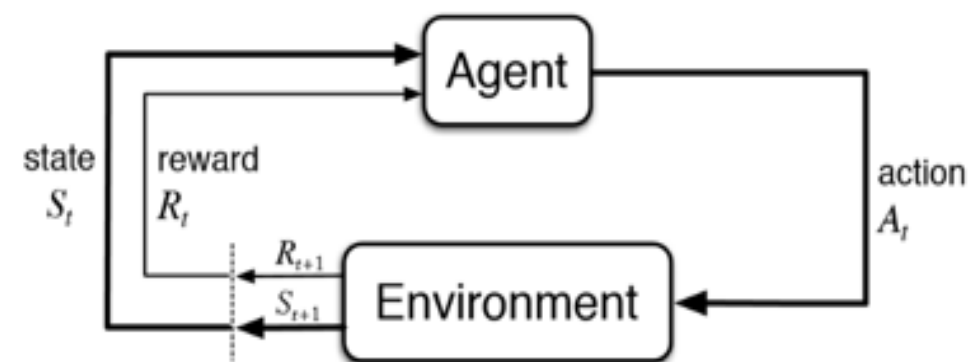


Figura 4.8 Diagrama de bloques control en cascada
Fuente: Autor.



$$Q[S_t, A_t] = (1 - \alpha) * Q[S_t, A_t] + \alpha * (R * \gamma * \text{Max}(Q[S_{t+1}, A_t]))$$

Figura 4.10 Q-Learning, Aprendizaje por refuerzo.
Fuente: Autor

IDENTIFICACION DE PARAMETROS

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{R}{2} \cos(\theta) & \frac{R}{2} \cos(\theta) \\ \frac{R}{2} \sin(\theta) & \frac{R}{2} \sin(\theta) \\ \frac{R}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} \dot{\phi}_R \\ \dot{\phi}_L \end{bmatrix}$$

Ecuación 3 Modelo propuesto [1]

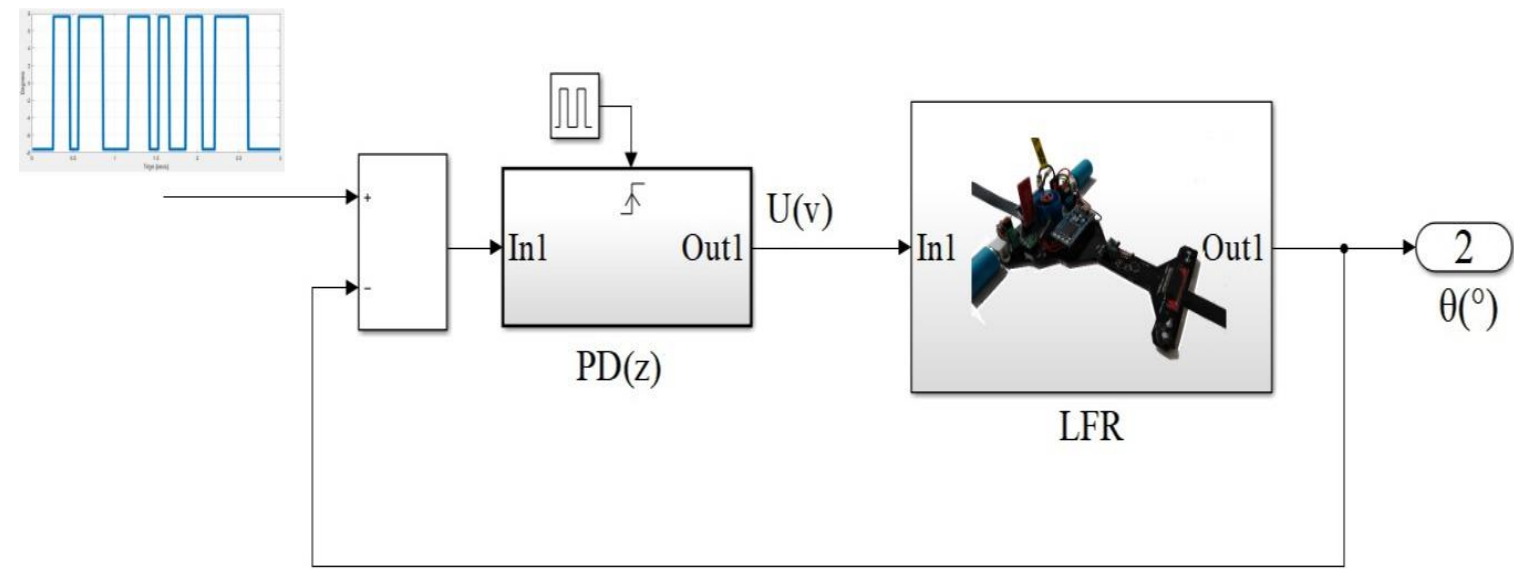


Figura 4.11 Ejercicio realizado para la identificación de parámetros. **Fuente:** Autor.

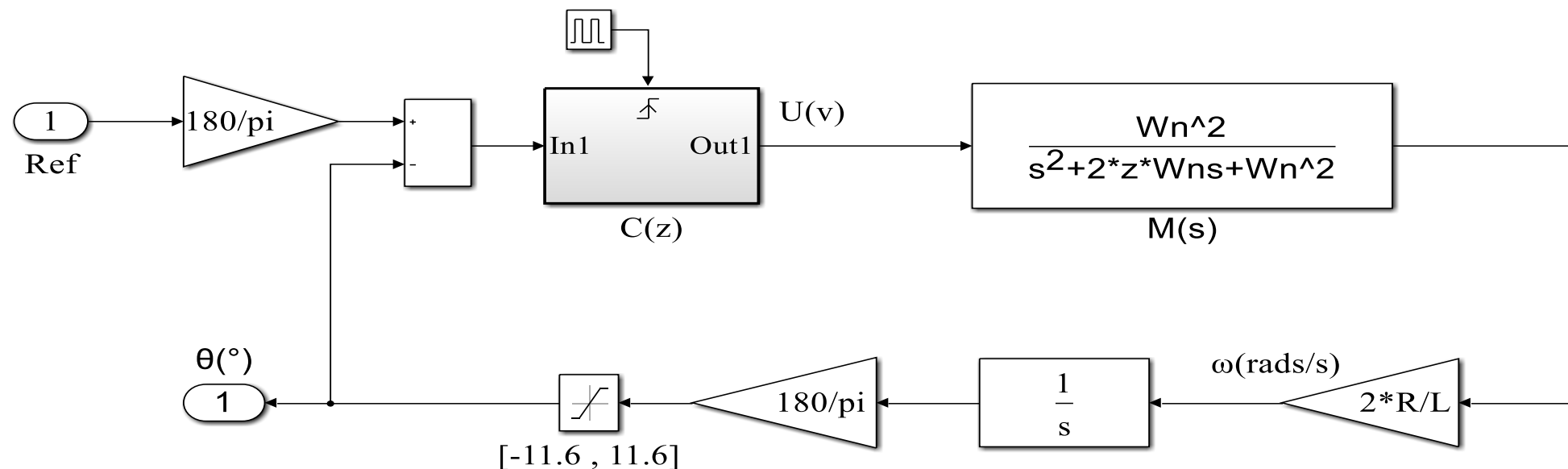


Figura 4.12 Modelo convencional reducido de un LFR
Fuente: [1].

IDENTIFICACION DE PARAMETROS

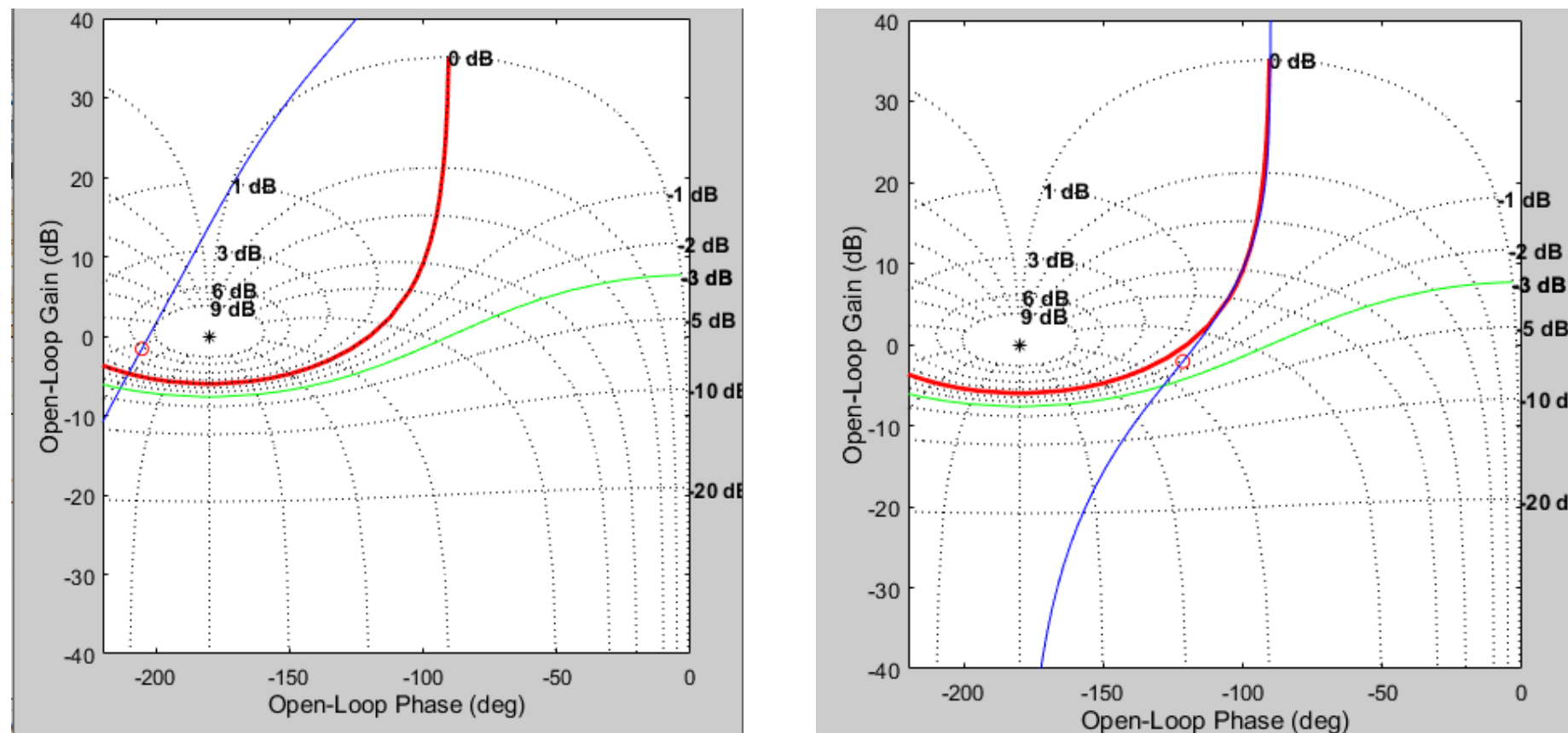
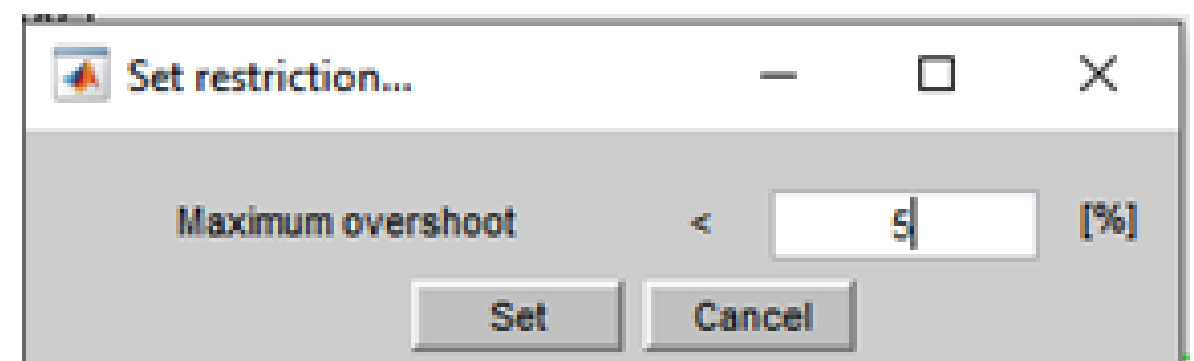
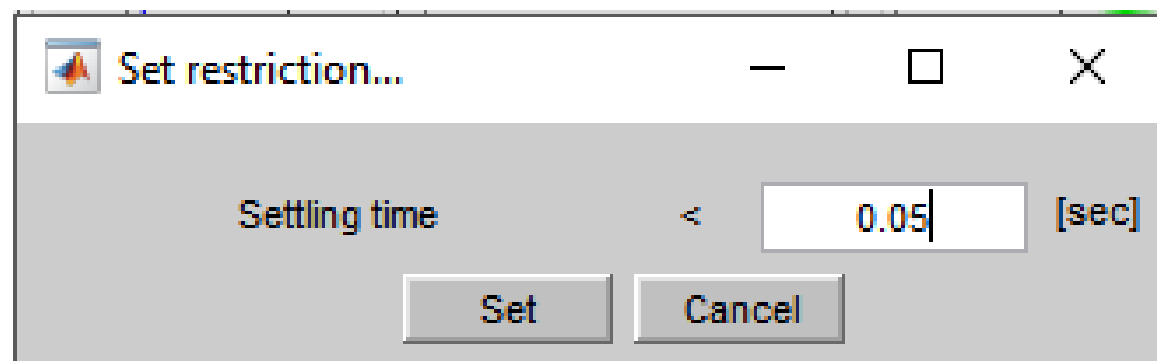


Figura 4.13 Resultado de modelo antes y después de ingresar controlador. **Fuente:** Autor



Resultados

Material: Fibra de vidrio
Dimensiones: 19.5x13.5 cm
Peso: 130g
Tiempo de autonomía: 3.7m



Figura 5.1 Vista lateral, trasera y superior del prototipo. **Fuente:** Autor.

Resultados

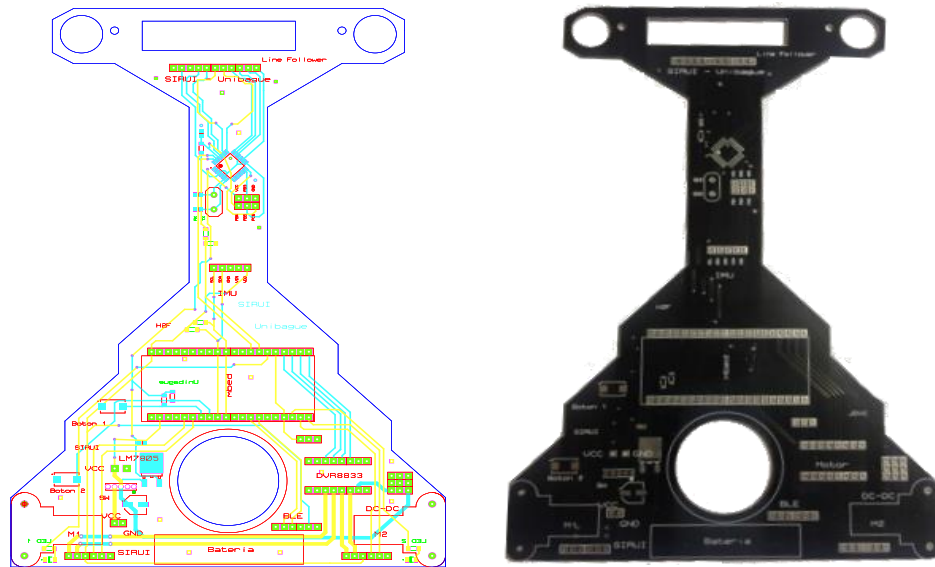


Figura 5.1 PCB (chasis) Fabricado.
Fuente: Autor.



Video 2. Proceso de ensamblaje. Fuente: Autor.

Material: Fibra de vidrio
Dimensiones: 19.5x13.5 cm
Peso: 130g
Tiempo de autonomía: 3.7m



Video 3. Prueba de funcionamiento durante Runibot 2019. Fuente: Autor.

Resultados

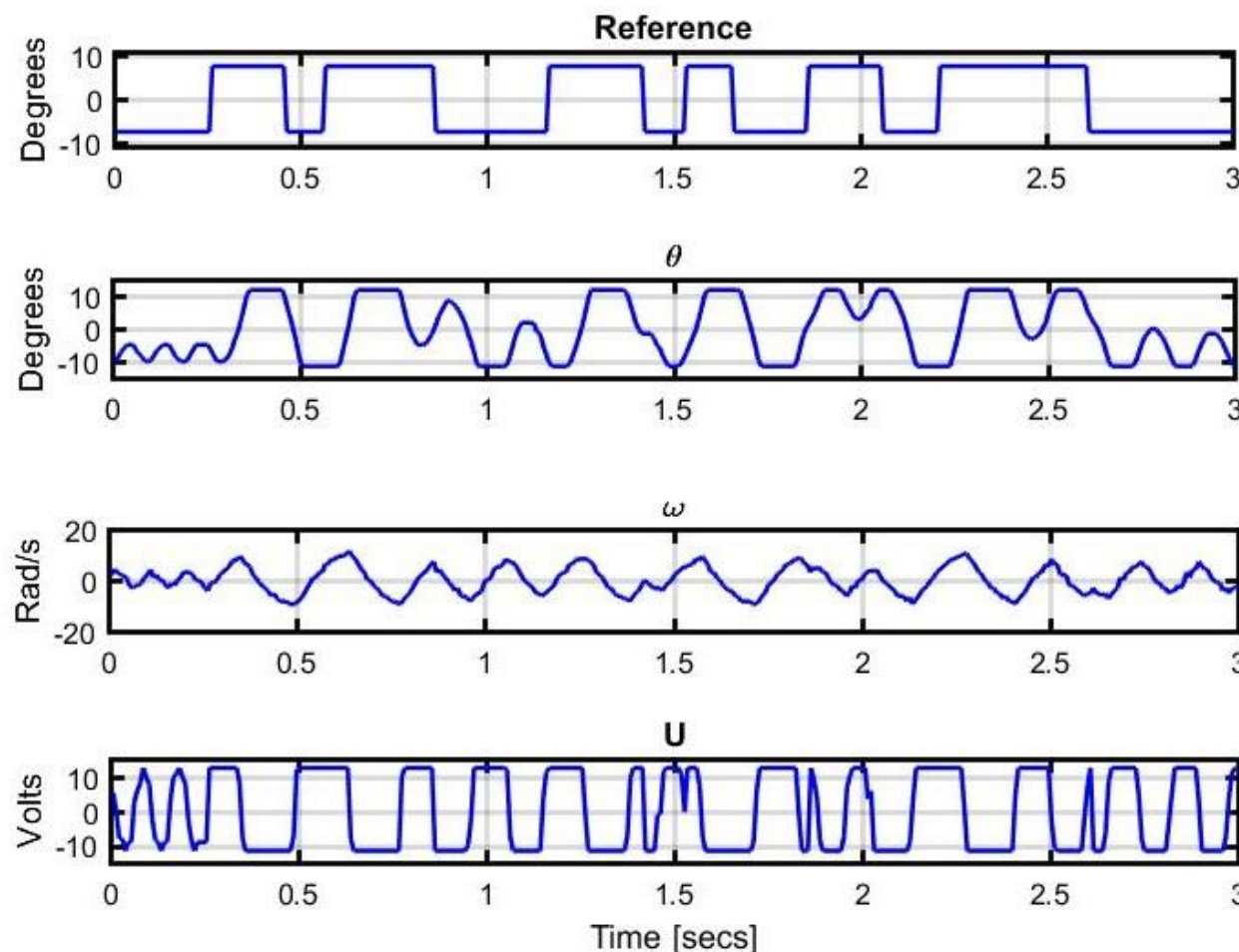


Figura 5.3 Señales de identificación de parámetros.
Fuente: Autor.

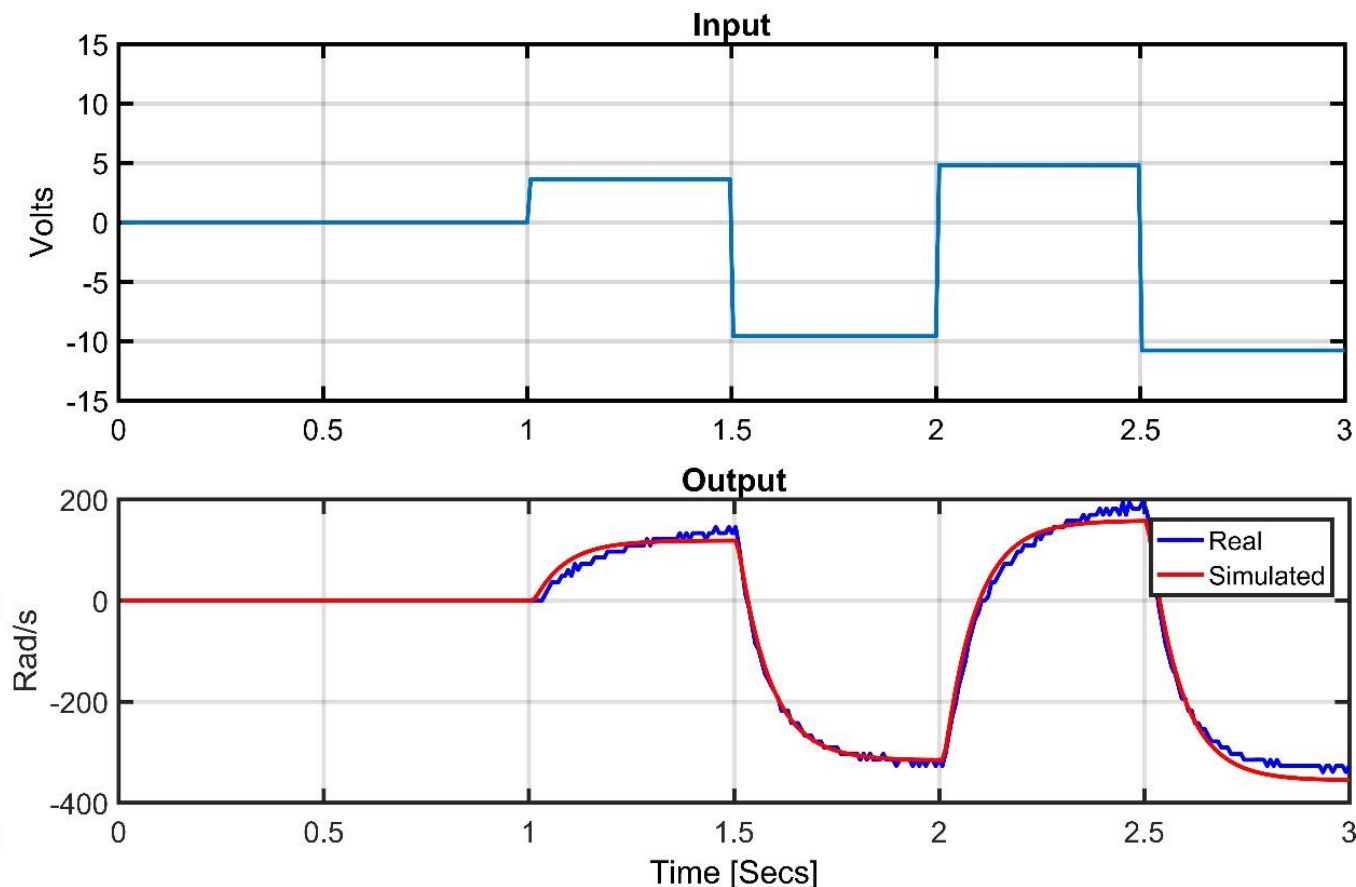
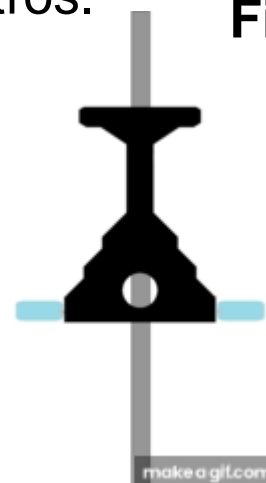


Figura 5.4 Resultados de identificación de $M(s)$ Con un ajuste del 86%. **Fuente:** Autor.

$$M(s) = \frac{\omega_n^2}{(s^2 + 2\zeta\omega_n s + \omega_n^2)}$$

Ecuación 3. Función de transferencia de 2do orden



$$M(s) = \frac{7.79 \times 10^4}{s^2 + 195.1s + 2364}$$

Ecuación 4. Función de transferencia Estimada.

Resultados

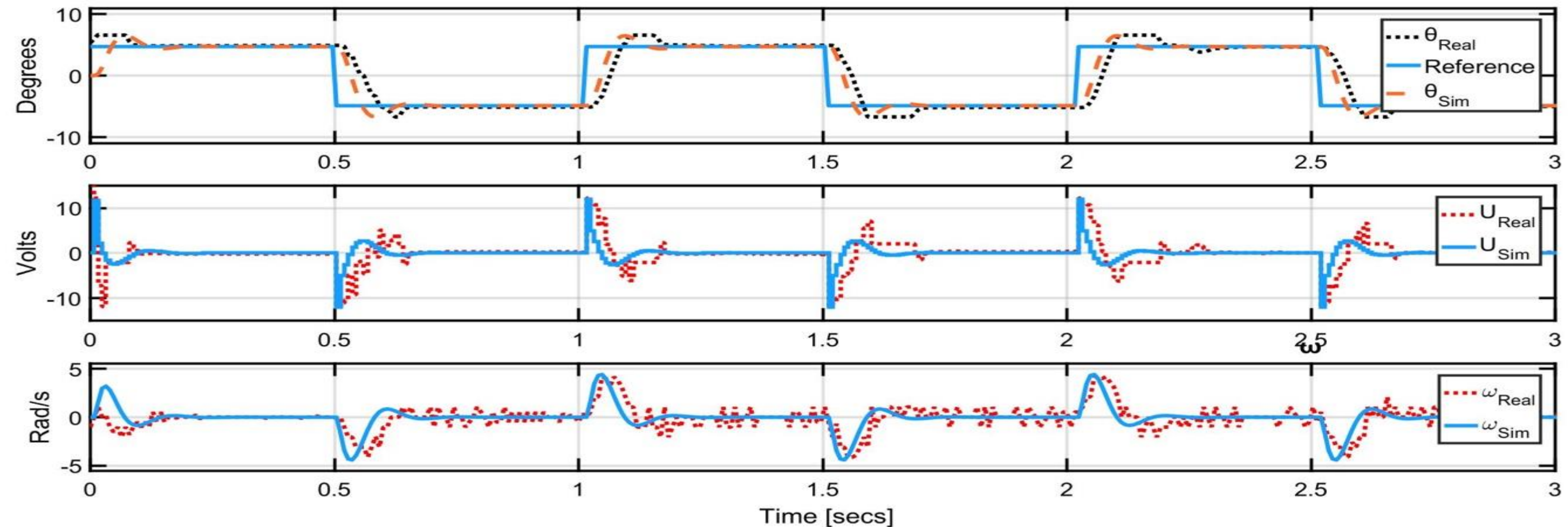


Figura 5.5 Validación del modelo convencional.

Fuente: Autor.

$$G(s) = \frac{\theta}{U} = \frac{2R}{L} * \frac{\omega_n^2}{s * (s^2 + 2\zeta\omega_n s + \omega_n^2)}$$

Ecuación 5. Función de transferencia
Del sistema.

$$G(s) = \frac{2.2954 \times 10^6}{s(s + 182.1)(s + 12.98)}$$

Ecuación 6. Función de transferencia
Del sistema estimada.

$$K_P = 0.1083 \quad K_D = 0.1125$$

Resultados

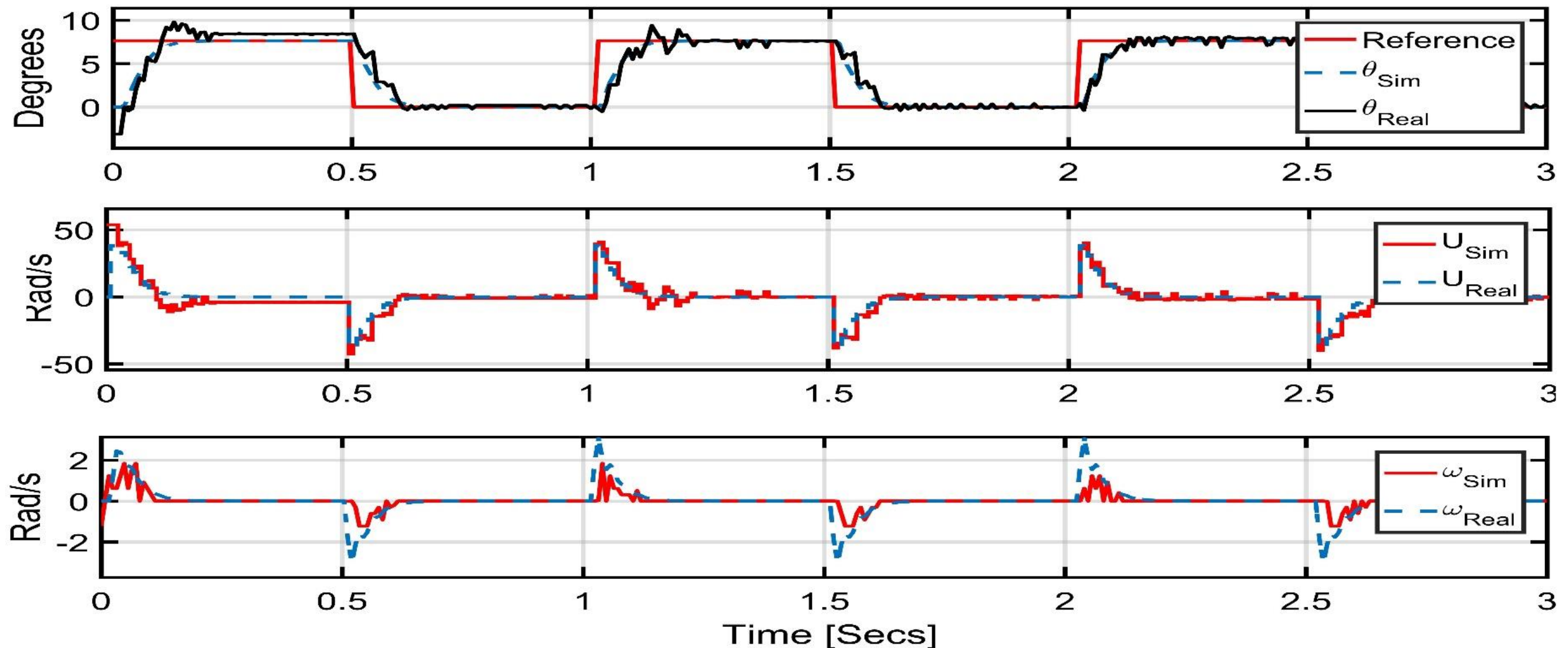


Figura 5.5 Validación del modelo en cascada.

Fuente: Autor.

$$K_p=1.4578, K_i=1.0413 \text{ y } K_D=1.4367$$

$$G(s) = \frac{\theta}{U} = \frac{2 * R}{Ls} M'(s) \quad M'(s) = \frac{Cm(s) * M(s)}{1 + Cm(s) * M(s)}$$

Ecuación 6. Función de transferencia

Del sistema en cascada.

$$K_p=18.35 \quad K_D=0.6$$

$$G(s) = \frac{3016.2 (s^2 + 184.9s + 1.131e04)}{s (s + 646.6) (s^2 + 162.7s + 1.075e04)}$$

Ecuación 6. Función de transferencia

Del sistema estimada.

Resultados

Tabla 1. Acciones y estados. Fuente: Autor.

Rango (°)	Estados (S_t)		Niveles de voltaje aplicados(v)		Acciones (A_t)	
			Motor Izquierdo (UL)	Motor Derecho (UR)		
$[-11.6336, -7.098]$	S_1	Izquierda	-7.92	7.92	A_1	Movers e Izquierda fuerte
$[-6.6098, -1.6847]$	S_2	Izquierda-Centro	-4.2	5.52	A_2	Movers e Izquierda suave
$[-1.5847, 1.5847]$	S_3	Centro	7.92	7.92	A_3	Adelante
$[1.6847, 6.6098]$	S_4	Derecha-Centro	5.52	, -4.2	A_4	Movers e Derecha suave
$[6.7098, 11.6336]$	S_5	Derecha	7.92	, -7.92	A_5	Movers e Derecha Fuerte

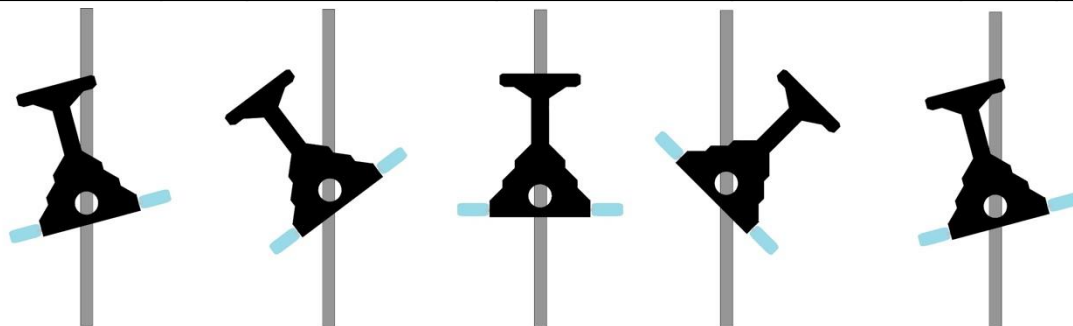


Figura 5.6 Proceso de entrenamiento. Fuente: Autor.

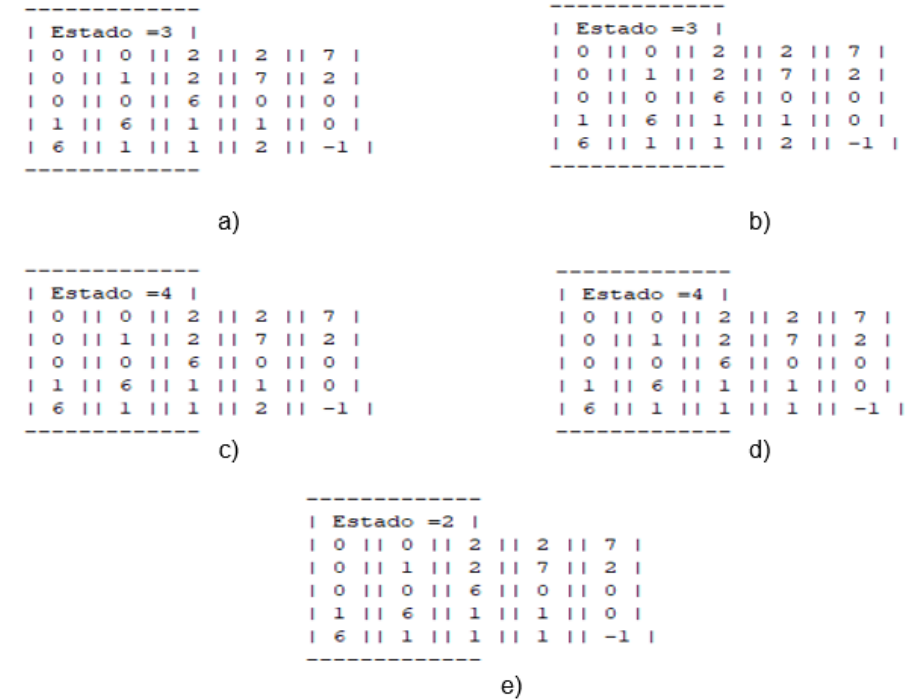


Figura 5.7 Convergencia de la matriz Q. Fuente: Autor.

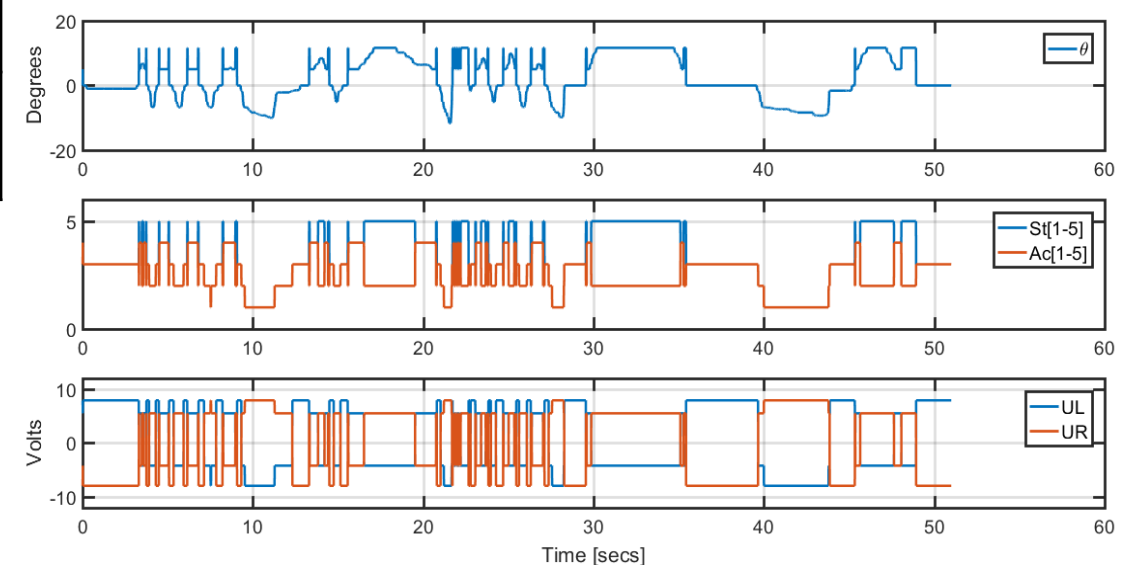


Figura 5.8 Resultados Q-Learning. Fuente: Autor.

Resultados

Tabla 2. Resultados adquiridos mientras se ponen a prueba las estrategias de control a la velocidad máxima de desplazamiento UBase y Vbase , respectivamente.

Fuente: Autor.

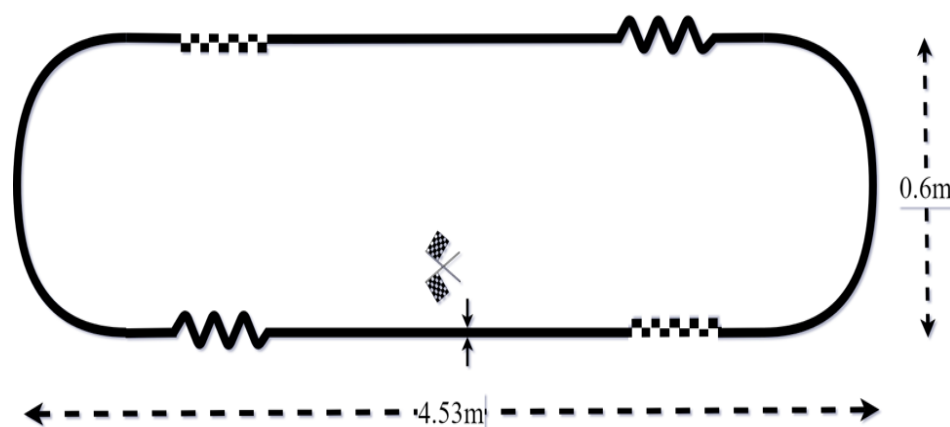


Figura 5.9 Circuito de pruebas.
Fuente: Autor.

Resultados				
Controlador	Base	STD (deg)	RMSE (deg)	Lap Time (s)
PID Simple	12v	~	~	inf
	8.4v	~	~	inf
	6v	4.04	4.20	7.47
	3,6v	2.51	2.51	28.37
PID con garantía de control	12v	4.43	5.03	4.64
	8.4v	4.62	4.76	5.68
	6v	4.25	4.41	7
	3,6v	2.76	2.77	17.85
Control en cascada	250rad/s	4,81	5.19	4.23
	220rad/s	4.67	4.87	5.04
	157rad/s	3.91	4.13	6.68
	94rad/s	3.05	3.20	14.3

Resultados

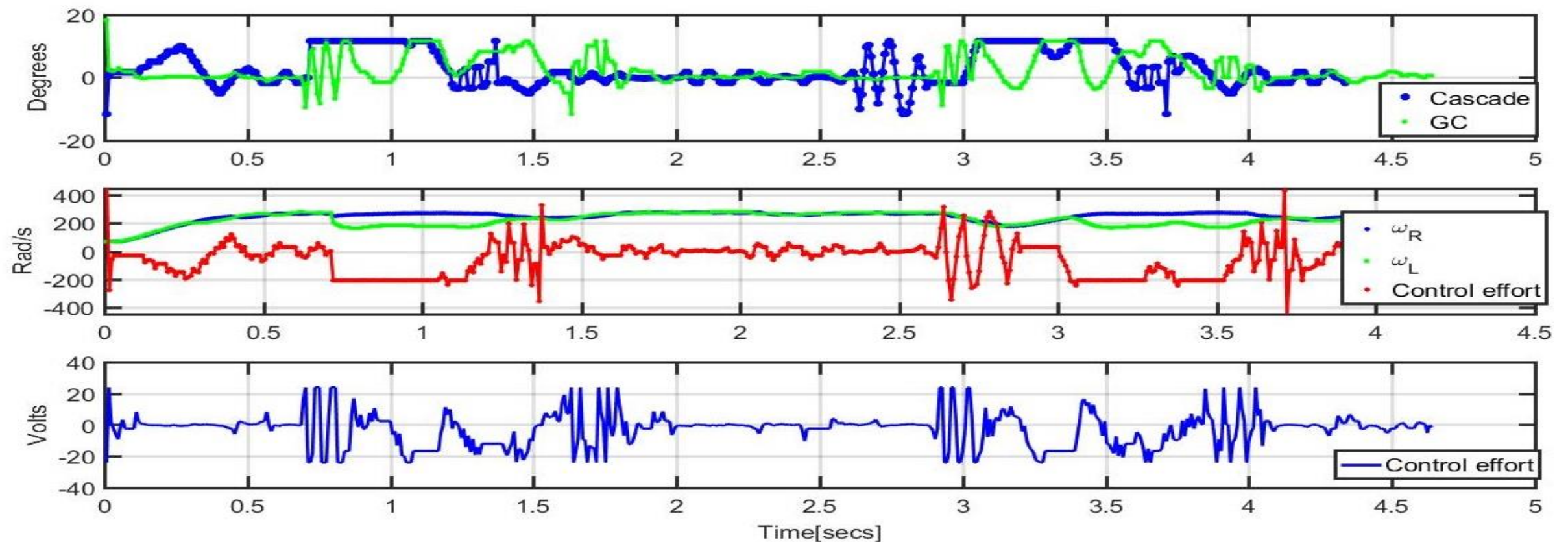
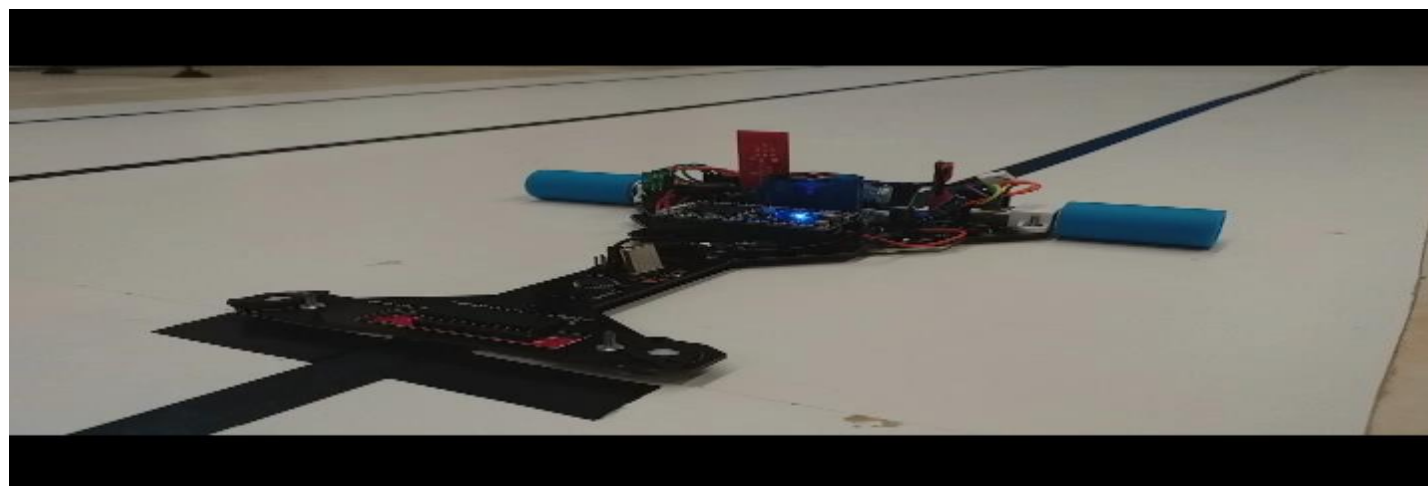


Figura 5.10 Resultados experimentales sobre el circuito de pruebas con la máxima velocidad de desplazamiento del LFR. **Fuente:** Autor.



Conclusiones

- La mejor forma de controlar un LFR es a través del sistema en cascada, teniendo en cuenta que fue la alternativa con mejor desempeño tanto en simulación como en su validación experimental, obteniendo el mejor tiempo de pista de 4.23 segundos con indicadores de estabilidad aceptables, respecto a las opciones de PD con garantía de control 4.64 segundos y 7.63 en la arquitectura convencional de control.
- La implementación de un controlador requiere de hardware adicional respecto a las estrategias convencionales, lo cual incrementa la complejidad del software, el peso y costo del robot.
- Actualmente un controlador PD convencional es suficiente para lograr un control óptimo de un LFR a altas velocidades, es por esto que muchos desarrolladores añaden rutinas propias adicionales, en busca de optimizar dicho controlador, en este trabajo se propuso el algoritmo de garantía de control como solución a este problema.
- El LFR es un sistema el cual permite implementar diferentes controladores para diferentes puntos de operación, que permitan optimizar los parámetros de estabilidad y velocidad de respuesta.

Recomendaciones

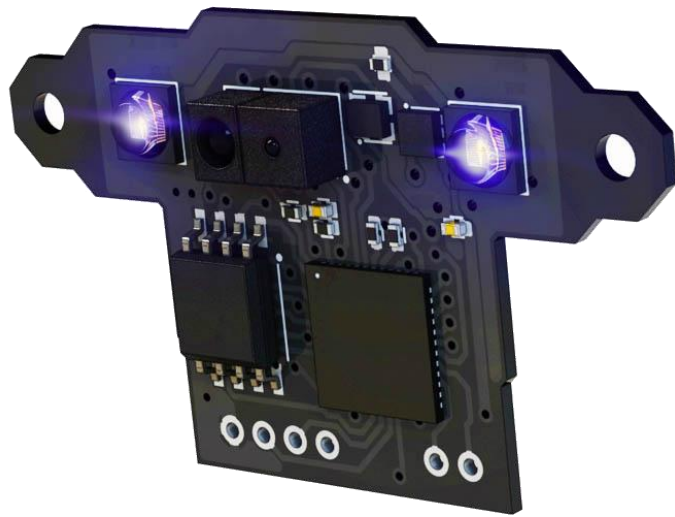


Figura 6.1 Sistema de visión de alta velocidad de
fotogramas. **Fuente:**
<https://hbfsrobotics.com/linefollower>.



Figura 6.1 Microcontroladores con arquitectura ARM.
Fuente: <https://www.arm.com/products/silicon-ip-cpu>.

Referencias

- [1] H. Murcia, J. D. Valenciano, and Y. Tapiero, “Development of a Line-Follower Robot for Robotic Competition Purposes,” *Appl. Comput. Sci. Eng.*, pp. 464–474, Oct. 2018.
- [2] Edgar Roberto Ramos., “Modelado, simulación y construcción de un robot móvil de ruedas tipo diferencial,” *Latin-American Journal of Physics Education*, Ciudad de Mexico, 2010.
- [3] H. F. Murcia and A. E. Gonzalez, “Performance comparison between PID and LQR control on a 2-wheel inverted pendulum robot,” in *2016 IEEE Colombian Conference on Robotics and Automation (CCRA)*, 2016, pp. 1–6.
- [4] M. Pakdaman and M. M. Sanaatiyan, “Design and Implementation of Line Follower Robot,” in 2009 Second International Conference on Computer and Electrical Engineering, 2009, pp. 585–590.
- [5] S. E. M. Bajestani and A. Vosoughinia, “Technical report of building a line follower robot,” in *2010 International Conference on Electronics and Information Engineering*, 2010, pp. V1-1-V1-5.
- [6] T. Fukao; H. Nakagawa; N. Adachi, “Adaptive tracking control of a nonholonomic mobile robot,” *IEEE Trans. Robot. Autom.*, vol. 16, pp. 609–605, 2000.

DISEÑO E IMPLEMENTACIÓN DE UN ROBOT SEGUIDOR DE LÍNEA DE COMPETENCIA PARA LA CATEGORÍA VELOCISTA

Aportes

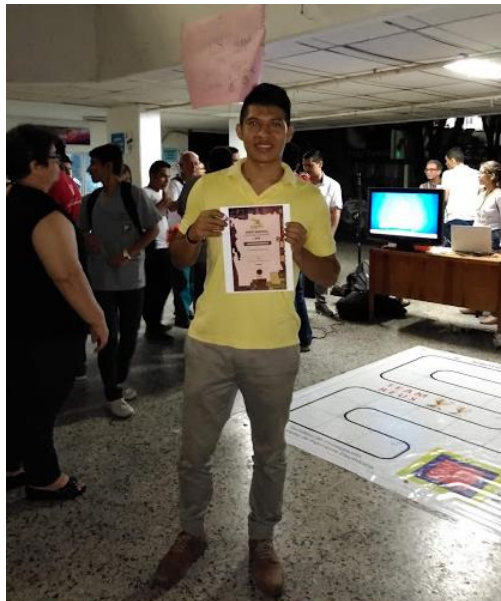


Figura 5.6 Primer puesto, RoboticPeople Fest Neiva 2018



Figura 5.6 Cuarto puesto, OMT. Tlaxcala, Mexico 2018



Figura 5.6 Primer puesto, RoboMatrix, Bogotá 2018



**DIRECCIÓN NACIONAL
DE DERECHO DE AUTOR**

Unidad Administrativa Especial
Ministerio del Interior

Figura 5.7 Software en proceso de obtención de registro por DIRECCIÓN NACIONAL DE DERECHOS DE AUTOR



Development of a Line-Follower Robot for Robotic Competition Purposes

Harold Murcia^(✉), Juan David Valenciano, and Yeison Tapiero

Facultad de Ingeniería, Grupo de Investigación D+TEC, Programa de Electrónica, Universidad Ibagué, Carrera 22 calle 67, Ibagué 730001, Colombia
harold.murcia@unibague.edu.co

Abstract. A fast line follower is an intelligent robot that must detect and follow a line drawn on a surface with possible changes of inclination. The robotics competitions demand that the robot go over a racetrack in the shortest possible time. The purpose of this paper is to study the line follower robot from the Control Engineering point of view to optimize its performance in standard races. In this paper we propose a SISO angle control scheme based on the relation between estimated line position and yaw robot angle. A sensitive position estimation respect to the line was implemented to improve the provided information interpretation from infrared array sensors respect to the conventional robots. Finally a suction turbine engine and a guarantee action algorithm were added to improve the angle controller response at high speeds and loss of grip on the wheels.

Keywords: Robotic competitions · Line follower robot
Embedded control

1 Introduction

Mobile robotics is one of the fields of robotics with more progress in recent decades. New mechanical configurations, control structures, unmanned vehicles developments and low-cost platforms have changed the way we knew about robotic systems. At university level, exist academic encounters which invite minds from around the world to compete in different events and have gained considerable popularity in recent years e.g.: RoboCup, Robot-challenge, all Japan Sumo, Mercury challenge and Robot-games. Colombia presents the same trend and the robotic competitions like Runibot and Robo-Matrix are increasing the number of participants each year. These spaces also compare efforts among universities allowing that participants to share experiences that not only help them strengthen their knowledge of the branches of engineering involved such as: Mechanical analytical properties of materials, electronic, instrumentation,

Our source implementation is freely available online and can be obtained from <https://github.com/HaroldMurcia/LineFollower.git>.

© Springer Nature Switzerland AG 2018
J. C. Figueroa-García et al. (Eds.): WEA 2018, CCIS 916, pp. 464–474, 2018.
https://doi.org/10.1007/978-3-030-00353-1_41

Figura 5.6 Paper de investigación presentado en WEA 2018.

Aportes

Durante el desarrollo de este proyecto, se obtuvieron diferentes resultados y productos, los cuales se presentan a continuación:

Participación en torneos de robótica junto con las muestras estudiantiles de la universidad:

- Torneo RoboMatrix en sus versiones 2016, 2017, 2018 I, 2018 II. Celebrado en la ciudad de Bogotá en la Corporación Universitaria Minuto de Dios y Universidad Piloto de Colombia. Obteniendo el primer lugar en la categoría velocista en las versiones 2018.
- Torneo Runibot en sus versiones 2016, 2017, 2018. Celebrado en la ciudad de Bogotá en las Universidades Católica, Cooperativa y Corporación Universitaria Minuto de Dios, obteniendo el cuarto puesto en la versión 2018.
- Torneo Robotic People Fest 2017 y 2018. Celebrado en las ciudades de Bogotá en el centro internacional de negocios Corferias y en la ciudad de Neiva en la Universidad Surcolombiana, obteniendo el primer lugar en su versión 2018. Cabe resaltar que durante la participación en el Robotic People Fest 2018, se logró una acreditación para asistir a las Olimpiadas Mexicanas de tecnología en la Ciudad de Tlaxcala, México en ese mismo año, apoyados por el programa de Ingeniería Electrónica de la Universidad de Ibagué.
- Muestra estudiantil del programa de Electrónica de la Universidad de Ibagué en sus versiones 2017B y 2018A. Además de las diferentes representaciones de semilleros del grupo D+TEC y participaciones como jurado en eventos de robótica.

Aportes

Participación en productos y eventos de carácter investigativo:

- Encuentro departamental de semilleros de investigación RedCOLSI nodo Tolima, Universidad Antonio Nariño sede Ibagué, noviembre de 2017.
- Encuentro departamental de semilleros de investigación RedCOLSI nodo Tolima, Universidad San Buenaventura sede Ibagué, marzo de 2018.
- En el ámbito investigativo, se logró la publicación de un artículo bajo el nombre de: *Development of a Line-Follower Robot for Robotic Competition Purposes*. Presentado en el evento WEA 2018..

Adicionalmente, se deja a disposición, un repositorio donde se encuentran los códigos fuente del proyecto los cuales cuentan con un registro de software con la Universidad de Ibagué en proceso, junto con los diagramas de conexiones y planos de PCB. Además de videos descriptivos que presentan el funcionamiento del robot, disponible en el siguiente enlace: https://github.com/HaroldMurcia/Line_Follower

Preguntas y Respuestas

