

# Diseño e Implementación de un Robot Velocista de Competencia sobre Plataforma FPGA

Herman Isaac Veriñaz Jadán <sup>(1)</sup>, Caril Ronnie Martínez Vera <sup>(2)</sup>, Ronald Ponguillo Intriago <sup>(3)</sup>  
Facultad de Ingeniería en Electricidad y Computación <sup>(1) (2) (3)</sup>  
Escuela Superior Politécnica del Litoral (ESPOL) <sup>(1) (2) (3)</sup>  
Campus Gustavo Galindo, Km 30.5 Vía Perimetral  
Apartado 09-01-5863. Guayaquil, Ecuador <sup>(1) (2) (3)</sup>  
hverinaz@espol.edu.ec (1) crmartin@espol.edu.ec (2) rponguil@espol.edu.ec (3)

## Resumen

*El presente trabajo consiste en el desarrollo de un robot velocista para competencias en el aspecto electrónico, mecánico y de software. El robot diseñado puede alcanzar una velocidad máxima de 2.23 m/s en el circuito de competición, que consiste en un camino cerrado formado por rectas y curvas, las cuales pueden tener un radio mínimo de 10 cm. El computador que controla el robot fue desarrollado sobre la FPGA EP4CE22F17C6N de la familia Cyclone IV de Altera. Para esto se utilizó la tarjeta de desarrollo DE0-Nano. El sistema usa el procesador definido en software NIOS II de Altera y posee periféricos para el manejo del robot.*

*El algoritmo que controla el robot es un sistema de control en cascada. Dos lazos secundarios controlan la velocidad de cada motor y un lazo primario controla cada referencia de acuerdo a la dirección y velocidad central del robot. Además se implementó en software un sistema de filtrado, para esto se usó el algoritmo de Filtrado de Kalman.*

**Palabras Claves:** Robot velocista, Seguidor de Línea, Robot Diferencial, FPGA, NIOS II, Filtro de Kalman.

## Abstract

*This paper presents the development of a Line Follower Robot for Robotrace Contests; it describes the electronics, mechanics and software design. This robot can reach top speed of 2.23 m/s in the competition circuit, which is a closed path that includes straight lines and curves with a minimum radius of 10 cm. The computer that controls the robot was designed on the Altera Cyclone IV FPGA (EP4CE22F17C6N); we used the DE0-nano Development board. The robot uses the Altera's soft-core processor NIOS II with specific cores for robot control.*

*The control algorithm proposed is a cascade control system. Two secondary loops controls the speed of each motor and a primary loop set each reference variable according with the desired direction and velocity for the robot. Also the system has filter blocks to improve the sensors readings; the algorithm used was the Kalman Filter.*

**Key words:** Robotracer, Line Follower, Differential Drive Robot, FPGA, NIOS II, Kalman filter.

## 1. Introducción

Tanto a nivel nacional como a nivel internacional existe una gran cantidad de torneos de robótica. A nivel nacional destaca el Concurso Ecuatoriano de Robótica (CER) y Robot Games Zero Latitud. Entre los principales a nivel internacional se puede nombrar: Robocup, RobotChallenge, All Japan Micromouse Contest, All Japan Robot SUMO Tournament.

Una de las categorías de competencia es la de robots velocistas. Los robots deben recorrer de forma autónoma en el menor tiempo posible un camino cerrado marcado con una línea. Existen además reglas adicionales que pueden diferir según la competencia.

Los robots más rápidos, dentro de su categoría, están en Japón, España, Polonia, Rumania y Letonia. En Latinoamérica cada vez se está mejorando más en este tipo de competencias, México es el país que destaca realizando buenas participaciones en torneos internacionales.

Estos eventos tienen como objetivo impulsar el desarrollo de diversas tecnologías relacionadas a la robótica. Además plantean un reto importante al enfrentar a los competidores a problemas reales que deben resolver usando ingeniería.

En el ámbito nacional el nivel de complejidad de los robots ha venido aumentando, aunque existe una monotonía en las soluciones que se presentan y aún se puede observar robots muy básicos. En general todavía no es común usar técnicas más profesionales.

En este proyecto no se busca presentar una solución óptima o concluyente para los problemas que se presentan en el desarrollo de un robot velocista. De hecho siempre se puede realizar mejoras en cada campo, sea electrónica, mecánica o software, de eso tratan las competencias.

En este trabajo se expone una solución específica, enfocada en el área de software y de electrónica, una implementación diferente a lo que se presenta actualmente en competencias nacionales, que permita desarrollar soluciones nuevas para robots velocistas y en general para robots de competencia ecuatorianos, de forma que se incremente el nivel de dificultad de torneos locales y la competitividad a nivel internacional.

## 2. Diseño e implementación

Para la construcción del robot es necesario trabajar en tres aspectos, el diseño físico o mecánico del robot, el diseño electrónico y el diseño de software. Primero se trata sobre el hardware, esto incluye el aspecto mecánico y electrónico del robot, luego se detalla los aspectos relacionados al software, empezando desde la creación de la computadora sobre la plataforma FPGA, el Soft-Core Processor junto a sus periféricos, hasta el diseño de los algoritmos de control y filtrado

usando MATLAB y su implementación sobre el procesador.

En la Figura 1 se muestra el diagrama de bloques del sistema descrito en esta sección.

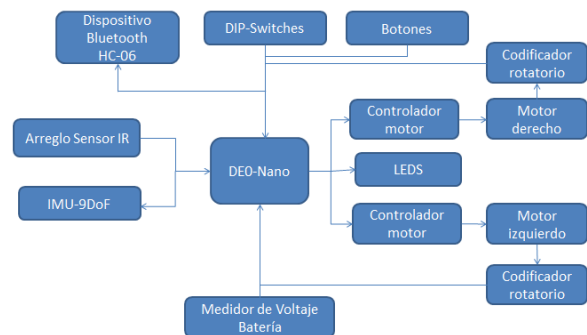


Figura 1. Diagrama de bloques del robot.

### 2.1. Hardware

#### 2.1.1. Mecánica

Este proyecto no se enfoca en el área de diseño mecánico, pero es necesario tener noción de la física que rige el comportamiento del robot de tal manera que el diseño del robot beneficie en lo posible el desempeño de los algoritmos de control y por consiguiente del robot.

Existen varias formas de locomoción para robots móviles, en el caso de los robots velocistas, las formas más usadas son la de tipo triciclo y la locomoción diferencial de dos y cuatro ruedas.

En este proyecto se usó la locomoción de tipo diferencial de dos ruedas, principalmente porque es eficaz y presenta la menor dificultad de implementación en el aspecto mecánico. En este tipo de locomoción se tiene dos ruedas fijas controladas, una a cada lado del robot, cada rueda es manejada de forma independiente, de esta manera el robot puede cambiar su dirección variando la velocidad de giro de cada rueda. Para balancear el robot se suele añadir ruedas adicionales, esféricas o ruedas orientables que giran en cualquier dirección.

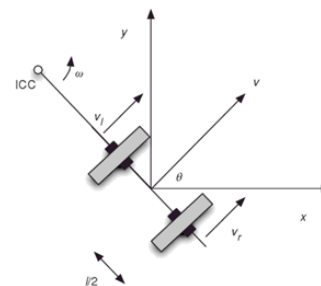


Figura 2. Cinemática diferencial.

Además se tomó a consideración ciertos detalles para el diseño físico del robot, los cuales se enlistan a continuación:

- Centro de gravedad bajo: Gracias a esto existe un menor riesgo de que la parte delantera del robot se levante alejándose de la pista, lo que puede originar errores en las lecturas de los sensores.
- Bajo momento de inercia: Aumenta la aceleración angular, lo que ayuda a que se desenvuelva mejor en las curvas.
- Masa del robot baja: Ayuda a mejorar la aceleración del robot.
- Ruedas: Son una parte importante, estas deben tener la máxima fricción posible, para ello se usó caucho de silicón. Las ruedas anchas mejoran la tracción hasta cierto punto, el radio depende de la velocidad que se quiera alcanzar, pero se debe tener en cuenta que un radio muy grande afecta otros factores, como la masa del robot y el momento de inercia de la carga sobre el motor.
- Unificación de chasis y circuito: El chasis de nuestro robot es la misma PCB, evitando montar la electrónica sobre un chasis aislado, así conseguimos tener menos puntos de fallo y un diseño más ligero.

Para finalizar esta sección, se menciona que los motores usados para el robot son FAULHABER 2224 SR06, estos motores tienen un voltaje nominal de 6V, alcanzan una velocidad angular sin carga de 8200 RPM y tienen un torque de parada de 21.2 mNm, además varios de los parámetros están detallados en la hoja de datos del motor [1].

### 2.1.2. Electrónica

La mayoría de componentes electrónicos usados corresponden a módulos desarrollados que se encuentran distribuidos en 2 placas diferentes, los cuáles a través del circuito adecuado se comunican con la tarjeta DE0-Nano entre los principales elementos tenemos:

- Módulo inalámbrico Bluetooth HC06.
- 2 Controladores para motores TB6612FNG Dual Motor Driver Carrier[2].
- Un regulador de 5V D24V50F5 con una corriente de salida de hasta 5A[3].
- Regulador de 3.3V LM1117-33.
- Botoneras, divisor de voltaje para medir el voltaje de la batería.
- Puertos de alimentación adicionales, para adaptar algún tipo de sensor requerida para una tarea en específico.

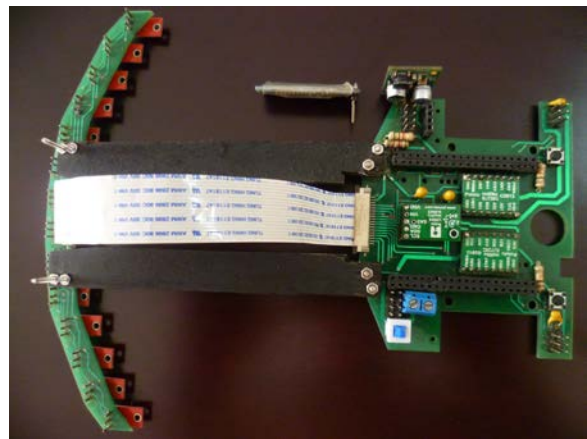
Además contamos con 3 componentes sensoriales como son:

- Un arreglo de que se compone de 14 sensores ópticos que distinguen el negro del blanco, y en este caso nos sirve para identificar la posición del

robot con respecto a la pista, para ello se usan sensores individuales QTR-1A de Pololu[4].

- Codificadores rotatorios o encoders, con los cuales podemos saber la velocidad de los motores en todo momento y poder controlarla, éstos vienen integrados en cada motor.
- Unidad de medición inercial o IMU, compuesta por un giroscopio, un acelerómetro y un magnetómetro, correspondiente al MinIMU-9 v3 de Pololu[5].

Cabe recalcar que es posible adaptar sensores adicionales, para alguna variante existente en este tipo de competencias.



**Figura 3. Circuito electrónico final en PCB, con los respectivos componentes.**

En la Figura 3 se aprecian los diferentes elementos que forman partes del diseño electrónico. La parte delantera corresponde a la placa que contiene el arreglo de sensores, la cual está conectada con el resto de componentes a través de un cable plano flexible.

Ambas placas fueron diseñadas a través de las diferentes herramientas proporcionadas por PROTEUS.

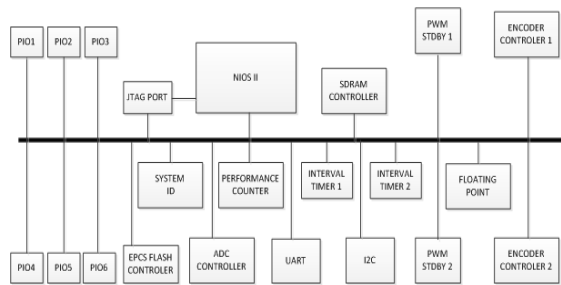
Para terminar este apartado, cabe mencionar que el robot está energizado mediante una batería de polímero de litio, comúnmente llamada LiPo, ya que son las ideales para este tipo de aplicaciones debido a su tamaño, capacidad de carga, y tasa de descarga, se cuenta con dos de éstas, de 1000 mAh y 300mAh, la primera es usada para pruebas, y la segunda para competencia, esto se debe a que la batería de 300mAh, tiene menor peso, y dura mucho menos, pero ese tiempo es suficiente para el que se necesita en una competencia.

## 2.2. Software

### 2.2.1. Creación de la computadora sobre FPGA

La creación de la computadora se la realizó a través de la herramienta Qsys de Quartus II.

Para construir la computadora se tomó como base la computadora DE0-nano Basic Computer [6] y se realizó algunas modificaciones para adaptarla al proyecto [7] [8]. En la Figura 4 se muestra el diagrama del sistema con cada módulo usado.



**Figura 4. Diagrama de bloques de la computadora sobre la FPGA.**

### 2.2.2. Controlador de velocidad

Para definir el movimiento del robot se necesita especificar su velocidad angular y la velocidad lineal del centro de masa.

Entonces, basta con controlar la velocidad lineal en el punto de contacto de cada rueda, para controlar la velocidad angular y lineal del robot.

Por esta razón, lo que primero se hizo fue diseñar un controlador de velocidad para cada motor. Estos controladores son a su vez manejados por un controlador externo que modifica su referencia según la velocidad angular y lineal que el robot deba tener.

Para identificar la función de transferencia del motor se almacenó los datos en la memoria SDRAM de la DE0-Nano, luego se transmitieron inalámbricamente con el módulo Bluetooth para ser analizados con MATLAB.

### 2.2.3. Algoritmo de filtrado

Las señales que pasan por el proceso de filtrado son las señales que provienen de los codificadores rotatorios, la señal que viene de los sensores infrarrojos y la del giroscopio.

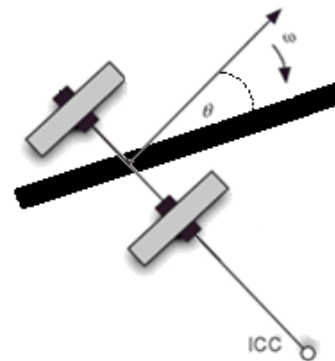
Lo que se hace es filtrar independientemente la señal de cada codificador rotatorio. La señal de los sensores infrarrojos y la del giroscopio se las pasa juntas por otro filtro que además fusiona ambos datos. Siendo el algoritmo de Filtrado de Kalman el que se implementó en ambos casos, debido al gran impacto que tiene este en la estimación de datos reales.

Para la implementación del algoritmo de filtrado en los codificadores rotatorios se necesita un modelo matemático expresado en matrices de estado. El modelo que se necesita debe tener como entrada la señal de voltaje que va a los motores y como salida la velocidad angular del eje del motor. Con esto se logra

una apreciación más real de la velocidad angular de cada motor.

Para filtrar las señales provenientes de los sensores infrarrojos y el giroscopio se usa también un filtro de Kalman, pero es usado de forma diferente a como se implementó con los codificadores rotatorios. Esto se explica a continuación.

Los sensores infrarrojos se usan para medir la separación angular entre la línea de la pista y la parte central delantera del robot, esto se observa en la Figura 5 el ángulo en mención es representado con  $\theta$ .



**Figura 5. Ángulo de separación entre el robot y la pista.**

Lo que se hace en software es leer el valor de cada uno de los sensores infrarrojos y en base a estas lecturas una función devuelve un número proporcional al ángulo que se desea medir.

Los sensores infrarrojos se pueden leer de forma analógica o digital. Si se usa el valor analógico, la medición es muy sensible a cualquier cambio en la iluminación del ambiente y se debe usar el convertidor analógico digital, esto vuelve más lento el proceso de adquisición. Pero la ventaja es que la señal analógica puede ser muestreada a cualquier rapidez y facilita el manejo por parte del controlador.

Si se usa el valor digital, la señal que se obtiene es más robusta frente al ruido ambiental, el proceso de leer los sensores se ejecuta rápidamente, pero el usar valores digitales hace que sea menos apropiada para ser manejada por el controlador, pues se introduce un error de cuantificación.

La solución que se implementó fue complementar los valores digitales de los sensores con la velocidad angular medida por el giroscopio, la unión de ambos datos da como resultado una señal analógica que es mejor manejada por el controlador.

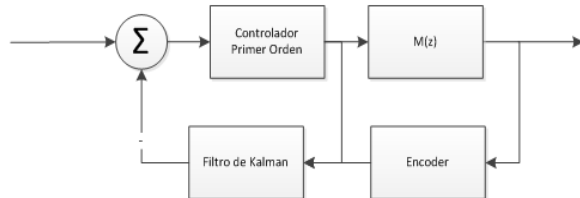
### 2.2.4. Controlador principal

El controlador principal es el encargado de definir la velocidad a la que debe girar cada rueda, es decir fija la referencia del controlador de velocidad de cada motor. Para tomar esta decisión se basa en la posición

angular y en la velocidad lineal que necesita tener el robot.

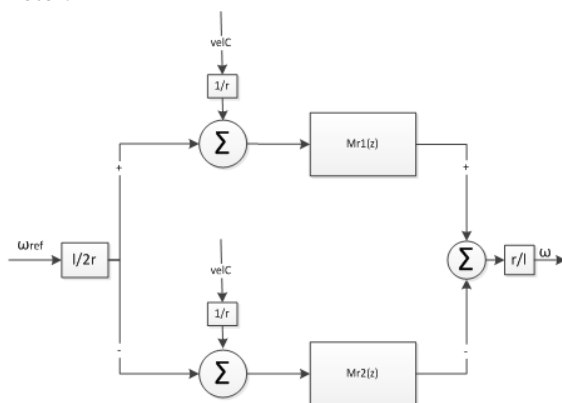
### 2.2.5. Diseño final del robot

En la Figura 6 se muestra el sistema realimentado para cada motor. El bloque  $M(z)$  representa la función de transferencia del motor, esta función tiene como entrada el voltaje de alimentación del motor y como salida la velocidad angular del motor.



**Figura 6. Sistema de Realimentación para Control de Velocidad del Motor.**

En el código del programa la interrupción ocurre cada 0.5ms, entre cada interrupción se ejecuta el algoritmo de filtrado con parámetro de entrada la velocidad angular medida por los codificadores y el último valor de PWM enviado al motor. El algoritmo de Kalman con estos parámetros realiza una estimación de la velocidad angular del motor, este valor es restado del valor de referencia para obtener la señal de error y con esta señal como parámetro de entrada se ejecuta el controlador. El controlador en base al error medido decide el nuevo valor de PWM y el lazo continúa cuando ocurre la siguiente interrupción. Este sistema es implementado para cada motor.

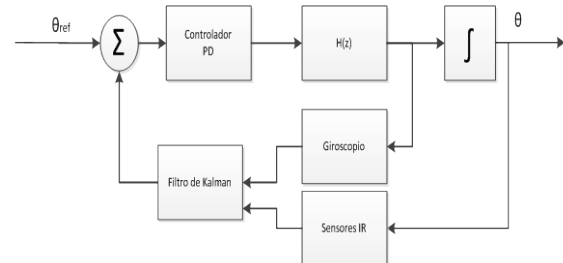


**Figura 7. Función de transferencia entre la Velocidad Angular de Referencia y Real del Robot.**

En la Figura 7 se muestra la función de transferencia que tiene como entrada el valor de referencia de la velocidad angular del robot y como salida la velocidad angular del robot. EL bloque  $Mr1$  representa el sistema de realimentación mostrado en la Figura 6 para un motor y el bloque  $Mr2$  representa

el sistema para el otro motor. A la salida de los bloques  $Mr1$  y  $Mr2$  la velocidad angular será similar al valor de entrada, ya que estos son sistemas realimentados y fueron diseñados para esto.

Todo el sistema anterior es resumido en un bloque llamado  $H$  y es representado junto al sistema principal de realimentación en la Figura 8.



**Figura 8. Sistema de Realimentación para el control de la Posición angular.**

El sistema de la Figura 8 sirve para controlar la posición angular del robot. En software lo que se hace es, en cada interrupción de 1.5ms, ejecutar las ecuaciones del algoritmo de filtrado con parámetro de entrada la lectura de la velocidad angular del giroscopio y la lectura de la posición angular de los sensores infrarrojos. El algoritmo fusiona ambos datos y hace una estimación de la posición angular, este valor es restado de la referencia para obtener la señal de error que entra al controlador.

El controlador en base al error medido decide el nuevo valor de referencia para el bloque  $H$ . A la salida de este bloque se tienen la velocidad angular del robot y luego su integral que es la posición angular, estos valores son medidos nuevamente en la siguiente interrupción y el lazo se repite.

La realización del proyecto implicó la construcción de dos modelos, en la Figura 9 se muestra el primer prototipo construido.



**Figura 9. Primer prototipo Construido.**

En este diseño el PCB fue hecho de baquelita, este material es poco resistente por lo que se tuvo que agregar partes de aluminio para sostener el chasis, además los motores se colocaron a cada lado del



robot, esto ocasionaba un incremento del momento de inercia. Para alcanzar la velocidad de 2 m/s se tuvo que usar ruedas de 2.1 cm de radio ya que los engranes que se disponían daban una relación de 8:1 y no permitían una mayor velocidad con un radio menor.

Para el diseño final el circuito fue impreso en FR-4 a doble lado, su tamaño se redujo respecto al primer diseño.



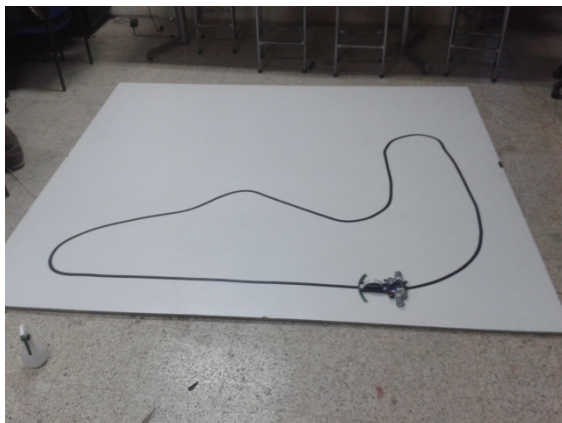
**Figura 10. Diseño Final del Robot**

En la Figura 10 se muestra el diseño final del robot. Las placas se unen con dos partes de madera, los circuitos se conectan usando cable plano flexible. Las ruedas tienen un radio de 1.5 cm, esta reducción del radio respecto al primer diseño fue posible cambiando la relación del sistemas de engranajes a 48:10.

Además se utilizó el software SolidWorks para diseñar los soportes que sostienen los motores y los aros donde van las llantas del robot

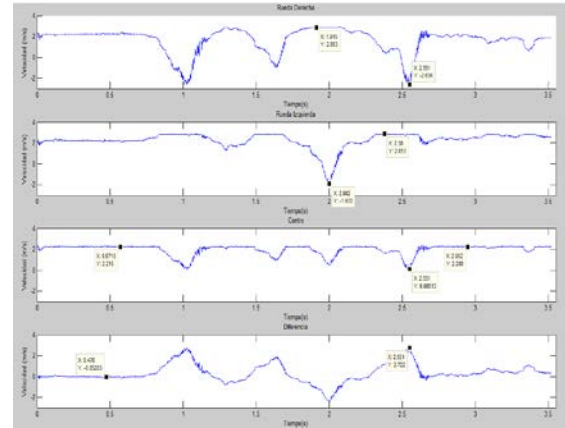
### 3. Pruebas y resultados

En la Figura 11 se puede observar el recorrido del robot en una pista de competencia.



**Figura 11. Pruebas del robot velocista en pista.**

Esta pista está formada por curvas y rectas que ideales para hacer pruebas dicha pista tiene una longitud total de 5.84 m, obteniendo los resultados que se muestran en la Figura 12.



**Figura 12. Diversas velocidades del robot durante su recorrido en la pista.**

Se puede apreciar cinco gráficas las cuales están ordenadas de la siguiente manera: En la primera gráfica se muestra la velocidad de la rueda derecha durante el recorrido, en la segunda se puede observar la velocidad durante el recorrido de la rueda izquierda, ambos datos se obtienen de los codificadores rotacionales del motor que corresponde al lado respectivo, como tercera gráfica tenemos la velocidad con la que se desplaza el centro de masa del robot a lo largo de la trayectoria lo cual es el promedio de las dos mediciones anteriores, por último se muestra la diferencia de velocidades que existe entre las velocidades a ambos lados que, que es en lo que se basa la locomoción de este tipo de robot con configuración diferencial y es la medida de la velocidad angular del robot multiplicada por una constante de proporcionalidad

Entre los principales resultados obtenidos sobre esta pista tenemos que el tiempo total que se tomó el robot en recorrerla es de aproximadamente 3.5 segundos, por lo que la velocidad media del robot a lo largo de la trayectoria es de 1.67 m/s aproximadamente.

Finalmente considerando la velocidad en el centro de masa del robot velocista, tenemos que este alcanza su máxima velocidad en los tramos rectos, donde su valor es de 2.23 m/s aproximadamente y esta velocidad disminuye su valor en las curvas.

Hay que tomar en consideración que estos valores son exclusivos de la trayectoria que sigue esta pista y que pueden variar dependiendo del recorrido de la misma, y los límites se establecerán de acuerdo a la curva con el menor radio de curvatura posible, así como del ángulo del arco de circunferencia que esta tenga.

Bajo estos datos se aprecia que el robot velocista alcanza velocidades muy buenas para este tipo de competencias lo cual consigue que de manera implícita un aumento en el nivel de las competencias de robótica nacionales.

## 4. Conclusiones

- El uso de la FPGA permite diseñar un microcontrolador a la medida, integrando en un solo chip los módulos necesarios para control del robot.
- El uso del Filtro de Kalman en las señales leídas desde los codificadores rotatorios permitió mejorar el control de velocidad de los motores.
- El uso del Filtro de Kalman permitió combinar la información obtenida del arreglo de sensores infrarrojos y del giroscopio para realizar una estimación de la separación angular y de su derivada.
- La realimentación en cascada permite estabilizar la posición del robot combinando datos de velocidad de las ruedas y la posición angular del robot.
- La adquisición de datos permite analizar en detalle el comportamiento del robot, ya sea respecto a los algoritmos de control, filtrado o estrategias implementadas.
- Matlab permite trabajar en mejoras del código haciendo simulaciones con los datos adquiridos.
- La tarjeta electrónica y en general el diseño de hardware del robot se debe hacer enfocándose en disminuir la masa del robot, mantener el centro de gravedad bajo y disminuir el momento de inercia alrededor del centro de curvatura.
- La elección de las ruedas es un punto importante en la construcción física del robot, se debe considerar material, radio y ancho.

## 5. Agradecimientos

Agradecemos a nuestros compañeros y profesores que son parte del Club de Robótica de la ESPOL "Robota" por facilitar materiales que fueron de utilidad el desarrollo del robot velocista en mención.

## 6. Referencias

- [1] FAULHABER, DC-Micromotors Precious Metal Commutation Series 2224...SR, [https://fmcc.faulhaber.com/resources/img/EN\\_2224\\_SR\\_DFF.PDF](https://fmcc.faulhaber.com/resources/img/EN_2224_SR_DFF.PDF).
- [2] POLOLU, TB6612FNG Dual Motor Driver Carrier, <https://www.pololu.com/product/713>.
- [3] POLOLU, Pololu 5V, 5A Step-Down Voltage Regulator D24V50F5, <https://www.pololu.com/product/2851>.
- [4] POLOLU, QTR-1A Reflectance Sensor, <https://www.pololu.com/product/958>.
- [5] POLOLU, MinIMU-9 v3 Gyro, Accelerometer, and Compass (L3GD20H and LSM303D Carrier), <https://www.pololu.com/product/2468>.
- [6] Altera, RS232 UART for Altera DE-Series Boards, [ftp://ftp.altera.com/up/pub/Altera\\_Material/11.1/U](ftp://ftp.altera.com/up/pub/Altera_Material/11.1/U)

niversity\_Program\_IP\_Cores/Communications/RS232.pdf, 2011.

- [7] Altera, NIOS II Hardware Development Tutorial, [http://www.altera.com/literature/tt/tt\\_nios2\\_hardware\\_tutorial.pdf](http://www.altera.com/literature/tt/tt_nios2_hardware_tutorial.pdf), 2011.
- [8] Altera, Basic Computer System for the Altera De0-Nano, [ftp://ftp.altera.com/up/pub/Altera\\_Material/12.0/Computer\\_Systems/DE0-Nano/DE0-Nano\\_Basic\\_Computer.pdf](ftp://ftp.altera.com/up/pub/Altera_Material/12.0/Computer_Systems/DE0-Nano/DE0-Nano_Basic_Computer.pdf), 2010.