

INSTITUTO TECNOLÓGICO SUPERIOR DE CAJEME

**“DISEÑO DE UN CONTROLADOR DIFUSO DE
NAVEGACIÓN PARA UN ROBOT TRANSPORTADOR DE
MATERIALES AUTÓNOMO INTELIGENTE.”**

**“TESIS PARA OBTENER EL GRADO DE MAESTRO
EN INGENIERÍA MECATRÓNICA”**

**PRESENTA
ING. ULISES BALDERRAMA REY**

**DIRECTOR DE PROYECTO
DR. FLAVIO MUÑOZ BELTRÁN**

CD. OBREGÓN, SON.

MARZO DE 2019

Fecha

04/02/2019

No. de Registro

MPIMC-036

Dr. Francisco Javier Ochoa Estrella
Coordinador del Programa de la Maestría en Ingeniería Mecatrónica
PRESENTE

A través de este documento, me permito comunicarle que el alumno

Ulises Balderrama Rey

cuyo trabajo de innovación o tesis:

DISEÑO DE UN CONTROLADOR DIFUSO DE NAVEGACIÓN PARA UN ROBOT TRANSPORTADOR DE MATERIALES AUTÓNOMO INTELIGENTE

Cumple satisfactoriamente, con los Lineamientos para la Operación de Estudios de Posgrado en el Sistema Nacional de Educación Superior Tecnológica, por lo que ha sido aceptado para la obtención del grado:

MAESTRO EN INGENIERÍA MECATRÓNICA

Sin más por el momento, quedo de Usted.

Flavio Muñoz B.

Flavio Muñoz Beltrán

04/02/2019

Nombre y Firma del Director de Tesis

Fecha

Director de Tesis

DICTAMEN DE ACEPTACION - COMISIÓN REVISORA

PRESIDENTE

Flavio Muñoz Beltrán

Nombre

Flavio Muñoz B.

Firma

21/01/2019

Fecha

MIEMBRO 1

Juan Enrique Palomares Ruiz

Nombre

Juan E. Palomares

Firma

28/01/2019

Fecha

MIEMBRO 2

José Guadalupe Castro Lugo

Nombre

José Gpe.

Firma

01/02/2019

Fecha

DEDICATORIA

A MIS PADRES: Por todo lo que me dieron en vida.

A MI NOVIA: Por su constante apoyo en esta etapa tan importante de mi vida.

AGRADECIMIENTOS

AL CONACYT: Por su apoyo, sin el cual no hubiera sido posible cursar ni aprovechar esta maestría.

A MIS MAESTROS: Por su paciencia, su conocimiento y experiencia que nos brindaron.

A MI ASESOR: Por el apoyo y dedicación en la elaboración de este proyecto.

A ITESCA: Por brindarme la oportunidad de estudiar en su sede Cajeme

RESUMEN

Actualmente, los vehículos autónomos son cada vez más utilizados en la realización de diversas tareas; razón por la cual el diseño e implementación de estrategias de control para estos sistemas es de vital importancia a fin de mejorar su eficiencia y desempeño. Aunque las variables de interés a supervisar son diversas, destacan dos de ellas que es necesario controlar en todo vehículo autónomo: la velocidad y la dirección.

De todas las diferentes estrategias de control para llevar a cabo la supervisión de la velocidad y la dirección, se eligió el control difuso porque no precisa de conocer el modelo matemático del sistema para controlarlo de manera precisa muy similar a los métodos de control no difuso. El objetivo general de este trabajo es diseñar un controlador difuso que controle la velocidad y dirección de un vehículo sin necesidad de un operador.

En el caso del control de velocidad, el diseño considera la variable de entrada de detección de obstáculos ya que al detectar un objeto en su trayectoria y este es susceptible de entrar en colisión debe de disminuir su velocidad hasta llegar a cero o en estado de frenado.

Para el control de la dirección que permita mantener al vehículo en la trayectoria de la ruta, se consideraron tres variables, las cuales detectan la desviación hacia la izquierda, derecha y si mantiene en el centro, nombradas como SDER, SCENT y SDER. Estas operan sobre la variable de salida de dirección que arroja valores que permiten definir si se mantiene centrado o se corrige la dirección.

Los conjuntos difusos y la base de reglas de este controlador fueron diseñados con el fin de emular y mejorar el tiempo de respuesta de un conductor humano. Ambos controladores se validaron y evaluaron a nivel de simulación en un ambiente **Matlab®**, utilizando el software **Fuzzy Logic Toolbox™**. El modelo de inferencia

difusa que se utilizó fue el de Mandami, debido a que sus características lo hacen apropiado para realizar un controlador que puede ser fácilmente programado. Como método defucificador se eligió el del centroide debido a su practicidad.

Una vez implementada su programación se obtuvieron resultados satisfactorios en todos los casos propuestos que involucraron pruebas con referencias de pérdida de dirección y detección de obstáculos.

Se concluye de los resultados obtenidos en este trabajo que el control difuso constituye una estrategia de control efectiva para ser implementado en vehículos autónomos.

ABSTRACT

At present, the autonomous vehicles are increasingly used in the performance of various tasks; this is the reason why the design and implementation of control strategies for these systems is of vital importance to improve their efficiency and performance. Although the variables of interest to be monitored are diverse, two of them stand out that it is necessary to control in all autonomous vehicles: speed and direction.

Of all the different control strategies to carry out the monitoring of speed and direction, fuzzy control was chosen because it does not need to know the mathematical model of the system to control it in a very similar way to non-diffuse control methods. The general objective of this work is to design a fuzzy controller that controls the speed and direction of a vehicle without the need of an operator.

In the case of speed control, the design considers the input variable of obstacle detection because when detecting an object in its trajectory and this is susceptible to collision it must decrease its speed until it reaches zero or in a braking state.

To control the direction to keep the vehicle in the path of the route, three variables were considered, which detect the deviation to the left, right and if kept in the center, named as SDER, SCENT and SDER. These operate on the output variable address that yields values that allow to define if it stays centered or the address is corrected.

The fuzzy sets and rulebase of this driver were designed to emulate and improve the response time of a human driver. Both controllers were validated and evaluated at simulation level in a Matlab® environment, using the Fuzzy Logic Toolbox™ software. The diffuse inference model that was used was that of Mandami, because its characteristics make it appropriate to perform a controller that can be easily programmed. The centroid method was chosen as the decontamination method due to its practicality.

Once its programming was implemented, satisfactory results were obtained in all the proposed cases that involved tests with references to loss of direction and detection of obstacles.

It is concluded from the results obtained in this work that fuzzy control constitutes an effective control strategy to be implemented in autonomous vehicles.

ÍNDICE GENERAL

RESUMEN	V
ABSTRACT	VII
ÍNDICE DE FIGURAS	XI
1.- INTRODUCCIÓN	143
1.1.-	
Antecedentes.....	143
1.2.- Planteamiento del problema	15
1.3.- Justificación	15
1.4.- Objetivo general.....	16
1.5.- Objetivos específicos	16
1.6.- Estado del Arte	17
2.- MARCO TEÓRICO	199
2.1 Robótica Inteligente.....	199
2.2 Visión Artificial	199
2.3 Machine Learning.....	200
2.4 Aprendizaje por refuerzo	200
2.5 Lógica difusa	211
2.5.1 Teoría de conjuntos difusos.....	222
2.5.1.1 Conjuntos Clásicos	232
2.5.1.2 Conjuntos Difusos.....	244
2.5.1.3 Funciones de Pertenencia	255
2.5.1.4 El Controlador Difuso	277
2.5.1.5 Fusificación.....	288
2.5.1.6 Base de Conocimiento.....	299
2.5.1.7 Inferencia	300
2.5.1.8 Defusificación	300
2.5.1.9 Definición de características del modelo.....	3131
2.6 Selección de Sensores	322
2.7 Selección de Actuadores.....	344

2.8 Selección de Electrónica de control355

3.- MÉTODO	377
3.1 Ubicación y función de sensores / actuadores	388
3.2 Diseño de la ruta	411
3.2 Diseño propuesto del robot	422
3.5 Diseño de control	433
3.6 Diseño del controlador difuso	444
4. Software utilizado	500
5.- CONCLUSIONES Y RECOMENDACIONES	633
6.- BIBLIOGRAFÍA	644
7.- ANEXOS	655

ÍNDICE DE FIGURAS

Figura 1.1.- Modelo cinemático del trabajo de Franyelit Suarez.....	17
Figura 1.2.-Trailer autónomo OTTO de Uber en pleno funcionamiento.	18
Figura 2.1.- Ejemplo de Conjuntos Difusos.	24
Figura 2.2.- Función de membresía para un conjunto difuso triangular.	25
Figura 2.3.- Función de membresía para un conjunto difuso trapezoidal.	25
Figura 2.4.- Operaciones Fuzzy.	27
Figura 2.5.- Estructura de un modelo difuso.....	28
Figura 2.6.- Ejemplo de fusificación de una variable de entrada (velocidad).	29
Figura 2.7.- base del conocimiento para frenar un vehículo.....	29
Figura 2.8.- Método del centroide.....	30
Figura 2.9. - Aplicación del sensor QRD1114 en la detección de líneas.....	33
Figura 2.10. - Aplicación del sensor HC-SR04 en la detección de objetos.	33
Figura 2.11. - Microinterruptor SEN-FDCM y funcionamiento.	34
Figura 2.12.- Motorreductor del tipo elevavinas y puente H BTS7960.	35
Figura 2.13. – Arduino Mega 2560.	36
Figura 3.1.- Ubicación de sensores de pruebas y presencia de paquete.....	39
Figura 3.2.- Ubicación de sensor y punto de control del sensor de obstáculo, en las áreas de partida(punto A) y llegada(punto B).....	40
Figura 3.3.- Trayectoria de la ruta dentro de las instalaciones del ITESCA.	41
Figura 3.4.- Líneas para mantener dentro de ruta al robot.....	41
Figura 3.5.- Función del robot.	42
Figura 3.6.- Diagrama de flujo.	43
Figura 3.7. – Ciclo de diseño del controlador difuso.....	44
Figura 3.8.- Funciones de membresía de SOI, SOD y SCENT.....	46
Figura 3.9.- Funciones de membresía de SDO.....	46
Figura 3.10.- Funciones de membresía de DIRECCION.....	47
Figura 3.11.- Funciones de membresía de VELOCIDAD.....	48
Figura 4.1. - Modelo FIS.....	50

Figura 4.2. – Funciones de membresía de la entrada SIZQ.....	51
Figura 4.3. – Funciones de membresía de la entrada SCENT.....	51
Figura 4.4. – Funciones de membresía de la entrada SDER.....	52
Figura 4.5. – Funciones de membresía de la entrada SDO.....	52
Figura 4.6. – Funciones de membresía de la salida DIRECCION.....	53
Figura 4.7. – Funciones de membresía de la salida VELOCIDAD.....	53
Figura 4.8. Gráfica del comportamiento de las entradas SCENT, SDER y la salida DIRECCION.....	55
Figura 4.9. Gráfica del comportamiento de las entradas SCENT, SIZQ y la salida DIRECCION.....	56
Figura 4.10. Gráfica del comportamiento de las entradas SIZQ, SDER y la salida DIRECCION.....	56
Figura 4.11.- Gráfica del comportamiento de la entrada SDO y la salida VELOCIDAD.....	56
Figura 4.12.- El vehículo va sobre la línea negra.....	57
Figura 4.13.- El vehículo tiende a ir a la izquierda saliéndose de la trayectoria.....	58
Figura 4.14.- El vehículo tiende a ir a la derecha saliéndose de la trayectoria.....	59
Figura 4.15.- Un obstáculo ha sido detectado por SDO.....	60
Figura 4.16.- El vehículo se está acercando al obstáculo.....	60
Figura 4.17.- El vehículo está cerca del obstáculo.....	61
Figura 4.18.- El vehículo está muy cerca del obstáculo.....	61
Figura 4.19.- El vehículo está muy próximo del obstáculo.....	62

1.- INTRODUCCIÓN

Los robots móviles son sistemas mecatrónicos capaces de moverse en un entorno con cierto grado de autonomía. La navegación autónoma se asocia a la disponibilidad de sensores externos que capturan información del entorno a través de imágenes visuales o mediciones de distancia o proximidad. Los sensores más comunes utilizados en la conducción de robots son los sensores de distancia que son capaces de detectar obstáculos y de medir la distancia a los objetos cercanos a la trayectoria del robot. Cuando los robots autónomos avanzados navegan dentro de ambientes interiores, tienen que ser dotados de la capacidad de moverse por los pasillos, de seguir las paredes, de dar vuelta a las esquinas y de entrar en áreas abiertas de las habitaciones (Suarez, 2018).

1.1.- Antecedentes

Los robots se desarrollan rápidamente desde entornos industriales donde se encuentran fijos a sus lugares de trabajo a máquinas cada vez más complejas capaces de realizar tareas desafiantes en nuestro entorno diario. Los robots industriales tradicionales utilizados en las plantas de fabricación donde el medio ambiente puede ser altamente controlado suelen ser más o menos estacionarios. Por el contrario, los robots móviles son los sistemas mecatrónicos que pueden operar en entornos sin restricciones y tienen la capacidad de moverse libremente.

Pueden funcionar de manera autónoma en un entorno parcialmente desconocido e impredecible sin la necesidad de dispositivos de guía físicos o electromecánicos (robot móvil autónomo). Alternativamente, los robots móviles pueden confiar en dispositivos de guía que les permiten recorrer una ruta de navegación predefinida en un espacio relativamente controlado (vehículo guiado automatizado).

1.2.- Planteamiento del problema

Actualmente la manera de realizar entregas de mensajería, así como de materiales en las instituciones educativas se realiza de forma tradicional, lo realiza un ser humano, sin aprovechar otras maneras como es el uso de robots autónomos. Los cuales pueden cumplir eficientemente esta tarea y permiten el ahorrar costos a través de no tener un puesto de trabajo que la realice o que se desplace de sus actividades a personal que no tengan en su perfil de trabajo la tarea de entrega de paquetes y/o sobres.

El Instituto Tecnológico Superior de Cajeme es una institución que al igual que las demás instituciones educativas necesita diariamente mantener comunicados los distintos departamentos que la componen generando muchas veces un problema de comunicación y entrega de los distintos equipos que son solicitados en otras áreas del plantel debido a la disponibilidad del personal encargado de cumplir con dichas tareas. Otro factor agravante que se presenta en las instituciones de algunos estados como sonora, son las altas temperaturas que se alcanzan en temporada de verano lo cual hace aún más incómodas estas actividades.

1.3.- Justificación

Todas las instituciones académicas o científicas requieren de un sistema interno de mensajería, paquetería o traslado de materiales, la realización de este proyecto dará soporte al Instituto Tecnológico Superior de Cajeme en la realización de algunas labores en el manejo de información y comunicación entre los distintos departamentos del plantel. Por lo cual el desarrollar un sistema autónomo de entrega de paquetes no solo ahorraría costos de personal sino además coloca a la institución en la vanguardia del uso de la tecnología para automatizar sus procesos.

1.4.- Objetivo general

Diseñar un controlador difuso que puede ser implementado en un robot de navegación autónoma y que sea capaz de desempeñar las labores de mensajería y transporte dentro del plantel de ITESCA.

1.5.- Objetivos específicos

- Desarrollo del controlador mediante lógica difusa.
- Propuesta de diseño de un controlador difuso que pueda ser utilizado en un robot de entregas donde controle el movimiento y conducción de este.

1.6.- Estado del Arte

Actualmente la investigación en sistemas de inferencia difusos se ha aplicado a vehículos autónomos a fin de que realicen tareas que tradicionalmente se llevaba a cabo con operadores humanos. Existen de muy variada índole, así como de alcance; como: la aplicación a robots seguidores de línea que no tienen otro propósito más que demostrar la eficacia de las técnicas de control difuso. Un ejemplo de este caso es el proyecto llevado por **Franyelit Suarez** en su artículo representa la fabricación de un robot de seguidor de línea con algoritmo de control híbrido basado en lógica difusa. Para el control de posición del robot se implementó un control PD+I difuso y para el control de velocidad se utilizó un controlador PI con realimentación odométrica, el robot en estudio es un robot seguidor de línea tipo tracción diferencial (Suarez, 2018).

Desde un punto de vista matemático podemos considerar el robot móvil de tipo tracción diferencial como un robot no holonómico y sus ecuaciones cinemáticas, dinámicas y de control son muy complejas. De la consideración anterior la lógica difusa es una buena alternativa para el desarrollo de controladores basado en inteligencia artificial y la experticia del proceso a controlar (Suarez, 2018). Donde utilizó Arduino y LabView para implementarlo, parte de su trabajo que es el modelo cinemático puede apreciarse en la figura 1.2.

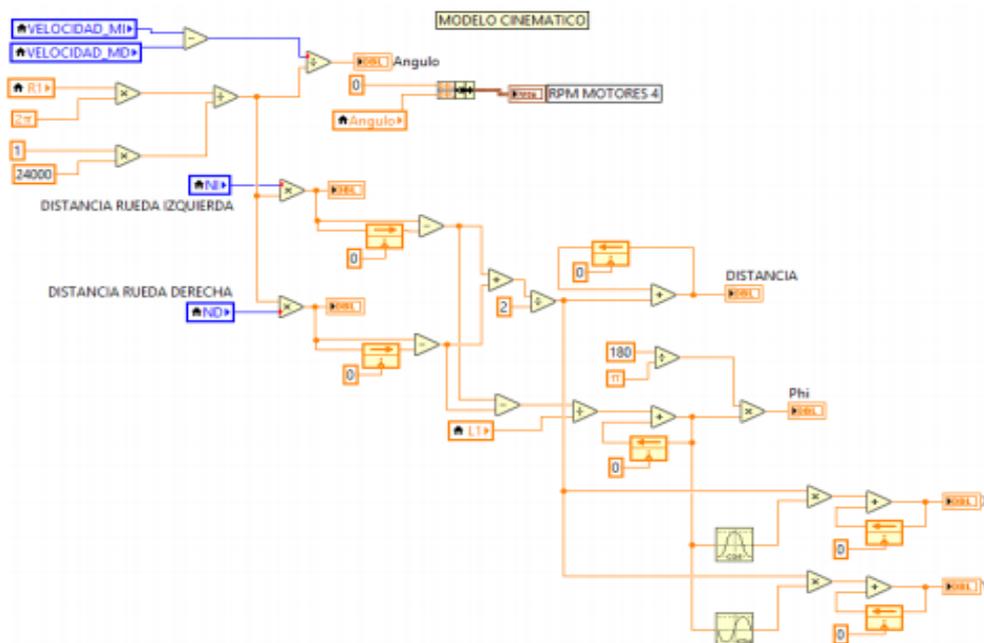


Figura 1.1.- Modelo cinemático del trabajo de Franyelit Suarez.

Sin embargo existen trabajos aún más ambiciosos como el presentado en el artículo: “Modelamiento y control de un sistema de tráiler autónomo” de **Wilder Medina** (Medina, 2018), donde propone una solución mediante un sistema de control de robot móvil tipo tráiler aplicado a una empresa que se encarga del transporte de material, y se demuestra que es una solución aplicable y viable. Se definió un modelo matemático que consiste en integrar dos técnicas de control: lineal LQR y lógica difusa. Se muestran resultados favorables del sistema de control y seguimiento de trayectoria, y se señalan las mejoras que se tienen al integrar ambas técnicas de control y que el sistema es capaz de responder correctamente a fin de guiar al robot por cualquier tipo de trayectoria (Medina, 2018), un ejemplo exitosa de aplicación de este tipo de sistemas es el tráiler autónomo llamado OTTO de Uber, figura 1.3.



Figura 1.2.-Trailer autónomo OTTO de Uber en pleno funcionamiento.

2.- MARCO TEÓRICO

Los Robots Inteligentes Autónomos son la nueva generación y están situados en su entorno, adoptan comportamientos, razonan, evolucionan y actúan como seres vivos. El fundamento de estas investigaciones es la Ciencia Cognitiva Corporizada y la Nueva Inteligencia Artificial. Su finalidad: desarrollar robots inteligentes y autónomos que razonan, se comportan, evolucionan y actúan como las personas.

2.1 Robótica Inteligente

La robótica inteligente autónoma es un enorme campo de estudio multidisciplinario, que se apoya esencialmente sobre la ingeniería (mecánica, eléctrica, electrónica e informática) y las ciencias (física, anatomía, psicología, biología, zoología, etología, etc.). Se refiere a sistemas automáticos de alta complejidad que presentan una estructura mecánica articulada gobernada por un sistema de control electrónico y características de autonomía, fiabilidad, versatilidad y movilidad.

En esencia, los robots inteligentes autónomos son sistemas dinámicos que consisten en un controlador electrónico acoplado a un cuerpo mecánico. Así, estas máquinas necesitan de adecuados sistemas sensoriales (para percibir el entorno en donde se desenvuelven), de una precisa estructura mecánica adaptable (a fin de disponer de una cierta destreza física de locomoción y manipulación), de complejos sistemas efectores (para ejecutar las tareas asignadas) y de sofisticados sistemas de control (para llevar a cabo acciones correctivas cuando sea necesario. (Tapia Fernández, 2009)

2.2 Visión Artificial

Visión Artificial. - La visión artificial trata de simular el proceso del sentido de la visión de los seres vivos, según el cual un objeto es captado mediante los receptores de la retina y es transformado en impulsos nerviosos que son procesados por el cerebro. La visión artificial consiste en la captación de imágenes en línea mediante cámaras basadas en matrices de sensores sensibles a la luz (CCD o CMOS), el posterior

tratamiento de estas mediante técnicas de análisis de imagen y la actuación sobre el proceso (control de producto) o el producto (control de calidad) (Morales, 2011).

2.3 Machine Learning

A diferencia de los programas informáticos clásicos, en los que la tarea se formaliza como una secuencia predefinida de instrucciones precisas, los algoritmos de ML basan sus decisiones en la información extraída de los datos. Esto es especialmente importante en los casos donde no es factible especificar la tarea explícitamente, o las especificaciones estáticas no son lo suficientemente robustas. Una de las definiciones clásicas de aprendizaje automático es proporcionada por Mitchell (Tom, 1955).

Definición: Se dice que un programa de computadora aprende de la experiencia E con respecto a alguna clase de tareas T y la medida de rendimiento R si su desempeño en tareas en T , medido por R , mejora con la experiencia E . Considere una tarea de clasificación de imágenes: se proporciona un conjunto de datos de imágenes con un gato o un perro en cada imagen. Para algunas de las imágenes, se dan las etiquetas, es decir, se sabe si hay un gato o un perro en la imagen, y para otro conjunto de imágenes, las etiquetas son desconocidas. La tarea T es inferir las etiquetas para las imágenes sin ellos la experiencia E consiste en el conjunto de imágenes emparejadas con sus etiquetas conocidas, también comúnmente denominadas conjunto de entrenamiento. El conjunto de imágenes con etiquetas desconocidas generalmente se llama conjunto de prueba. La medida de rendimiento R mide que tan bien se generaliza el conocimiento extraído del conjunto de entrenamiento para hacer predicciones sobre las etiquetas de las imágenes en el conjunto de prueba. (Pedro, 2010)

2.4 Aprendizaje por refuerzo

El aprendizaje por refuerzo consiste en aprender que acciones realizar, dado el estado actual del ambiente, con el objetivo de maximizar una señal de recompensa numérica, lo cual requiere de un mapeo de situaciones a acciones. El sistema que aprende debe descubrir por si solo cuales son las acciones que le dan a ganar más. En los casos más

interesantes y difíciles, las acciones pueden afectar no solo la recompensa inmediata sino también la siguiente situación, y de esta forma afectar las siguientes recompensas. (Cetina, 2012).

2.5 Lógica difusa

La palabra “fuzzy” proviene del inglés y significa “difusa” y se puede tener significados como como “borroso, impreciso, confuso, vago”, mas esta definición de la palabra “difusa” como un adjetivo no tiene una relación directa con la teoría de lógica difusa ya que, si bien los fenómenos que analiza pueden ser difusos, la propia teoría es precisa. En ingeniería casi todas las teorías que caracteriza lo que ocurre en la realidad son de una manera aproximada.

La teoría difusa se fundamenta en la teoría de los conjuntos difusos la cual fue introducida por el profesor Lotfi Asker Zadeh en 1965 (Zadeh, 1965), docente de la Universidad de California en el área de Ingeniería Eléctrica y Ciencias de la Computación en Berkeley. La primera vez que se utilizó la palabra “fuzzy” en la literatura técnica y se describe con la teoría matemática de conjuntos clásicos fue en el artículo publicado por Zadeh, donde también nos muestra la posibilidad de trabajar de forma matemática con datos imprecisos, como lo hacen los humanos.

Siendo una extensión de la teoría de conjuntos clásicos la teoría de los conjuntos difusos establece la pertenencia de clase de un objeto con un cierto grado, sin haber líneas de separación abruptas entre una clase de objetos y otros; en el caso clásico un elemento puede pertenecer o no a un cierto conjunto. En la teoría de conjuntos difusos, el elemento tiene un grado de pertenencia en el cual puede ir desde un grado de pertenencia máximo hasta un grado de pertenencia mínimo, que sería la no pertenencia.

En la realidad todo fenómeno que se tenga que procesar arroja datos con incertidumbre, imprecisión, ambigüedad, vaguedad o información incompleta. La teoría difusa viene ayudar en la comprensión y tratamientos de dichos datos.

Y fue en los inicios de los 60, cuando Zadeh pensó que la teoría del control clásico había puesto demasiado énfasis en la precisión y por consiguiente no podría manejar los sistemas complejos con incertidumbre. Zadeh, comentó: “necesitamos un tipo de matemática sumamente diferente, la matemática de cantidades difusas o borrosas que no sean describibles en términos de una distribución probabilística” (Zadeh, From Circuit Theory to System Theory, 1962), es por esta razón, que propone la teoría de los conjuntos difusos.

El incremento del uso del control difuso se ha dado por el éxito práctico al utilizarlo en equipos electrodomésticos y en procesos industriales. Hay que tomar en cuenta que la lógica difusa se puede aplicar en procesos demasiados complejos, cuando no existe un modelo de solución simple o un modelo matemático preciso. Así como también se necesite usar el conocimiento de un experto que utiliza conceptos ambiguos o imprecisos. De la misma manera se puede aplicar cuando ciertas partes de un sistema a controlar son desconocidas y no pueden medirse de forma confiable y cuando el ajuste de una variable puede producir el desajuste de otras. ***Sin embargo, no es recomendable utilizar la lógica difusa cuando algún modelo matemático ya soluciona eficientemente el problema, cuando los problemas son lineales o cuando no tienen solución.***

2.5.1 Teoría de conjuntos difusos

La lógica difusa permite tratar con información que no es exacta o con un alto grado de imprecisión a diferencia de la lógica convencional la cual trabaja con información precisa. El problema principal surge de la poca capacidad de expresión de la lógica clásica.

2.5.1.1 Conjuntos Clásicos

Los conjuntos clásicos surgen por la necesidad del ser humano de clasificar objetos y conceptos. Estos conjuntos pueden definirse como un conjunto bien definido de elementos o mediante una función de pertenencia μ que toma valores de 0 o 1 de un universo en discurso para todos los elementos que pueden o no pertenecer al conjunto (Ross, 1995). Un conjunto clásico se puede definir con la función de pertenencia mostrada en:

$$\mu = \begin{cases} 0 & \text{si } x \notin A \\ 1 & \text{si } x \in A \end{cases} \quad 2.1$$

2.5.1.2 Conjuntos Difusos

La necesidad de trabajar con conjuntos difusos surge del hecho que existen conceptos que no tienen límites claros. Un conjunto difuso se encuentra asociado por un valor lingüístico que está definido por una palabra, etiqueta lingüística o adjetivo. En los conjuntos difusos la función de pertenencia puede tomar valores del intervalo entre 0 y 1, y la transición del valor entre cero y uno es gradual y no cambia de manera instantánea como pasa con los conjuntos clásicos (Ross, 1995). Un conjunto difuso en un universo en discurso puede definirse como lo muestra:

$$A = \{x, \mu_A(x) | x \in U\} \quad 2.2$$

Donde $\mu_A(x)$ es la función de pertenencia de la variable x , y U es el universo en discurso. Cuando más cerca este la pertenencia del conjunto A al valor de 1, mayor será la pertenencia de la variable x al conjunto A (Ross, 1995), esto se puede ver en la figura 2.1.

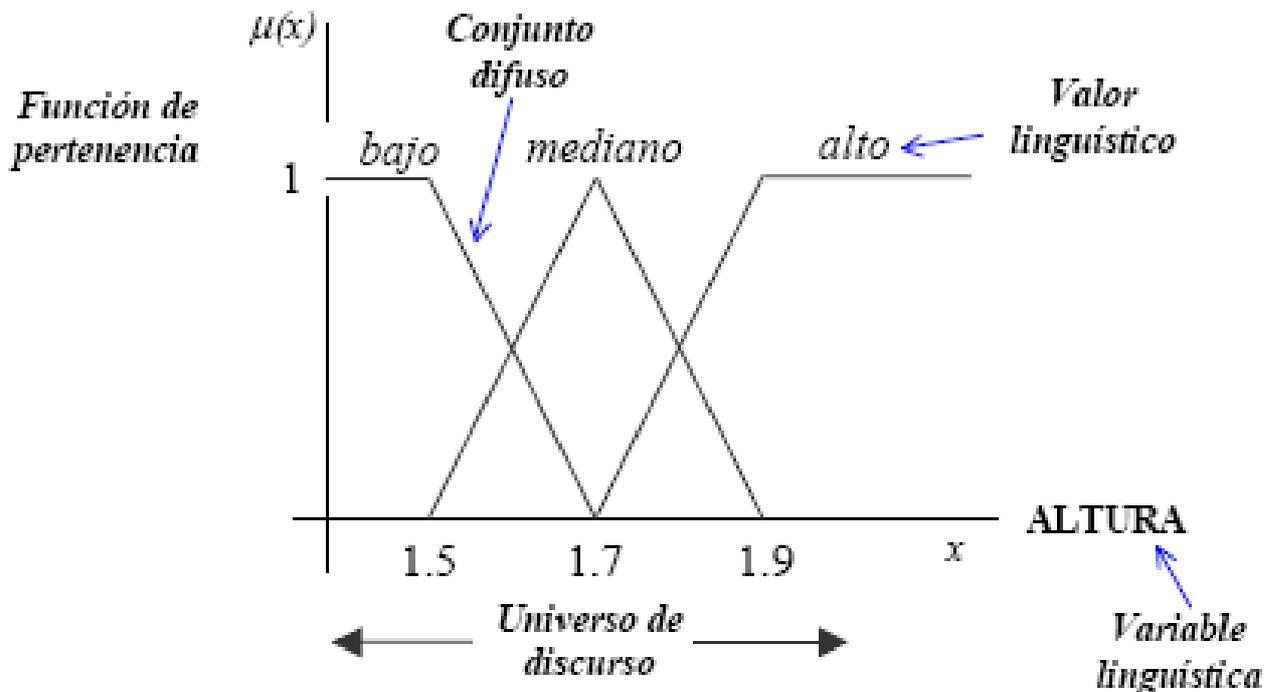


Figura 2.1.- Ejemplo de Conjuntos Difusos.

2.5.1.3 Funciones de Pertenencia

Aun cuando cualquier función puede ser válida para definir un conjunto difuso, existen ciertas funciones que son más comúnmente utilizadas por su simplicidad matemática, entre éstas se encuentran las funciones de tipo triangular, mostrado en la figura 2.2, trapezoidal mostrado en la figura 2.3, entre otras.

$$\mu(x) = \begin{cases} 0 & \text{para } x \leq a \\ \frac{x - a}{m - a} & \text{para } a < x \leq m \\ \frac{b - x}{b - m} & \text{para } m < x \leq b \\ 0 & \text{para } x > b \end{cases}$$

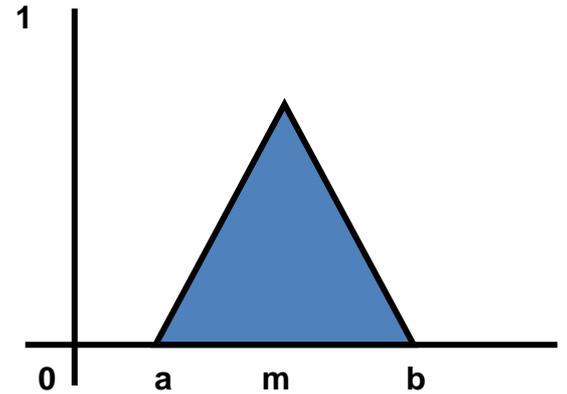


Figura 2.2.- Función de membresía para un conjunto difuso triangular.

$$\mu(x) = \begin{cases} 0 & \text{para } x \leq a \\ \frac{x - a}{b - a} & \text{para } a < x \leq b \\ 1 & \text{para } b < x \leq c \\ \frac{d - x}{b - c} & \text{para } c < x \leq d \\ 0 & \text{para } x > d \end{cases}$$

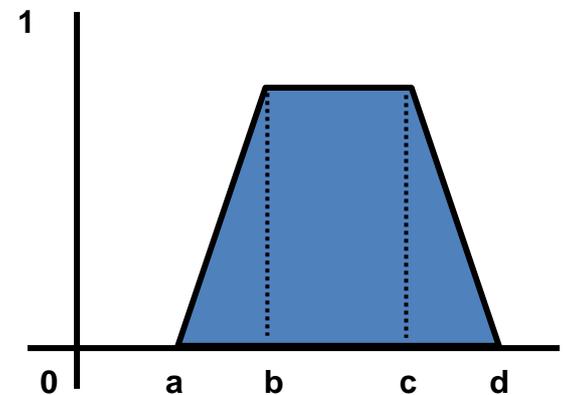


Figura 2.3. - Función de membresía para un conjunto difuso trapezoidal.

2.5.1.4 Operaciones Fuzzy

A los subconjuntos se les puede aplicar determinados operadores o bien se puede realizar operaciones entre ellos. Al aplicar un operador sobre un solo conjunto se obtendrá otro conjunto, lo mismo sucede cuando se realiza una operación entre conjuntos.

Las operaciones lógicas se utilizan en controladores y modelos difusos, son necesarias en la evaluación del antecedente de reglas (y otras etapas) que más adelante veremos.

Se definen a continuación 3 operaciones básicas a realizar sobre conjuntos, estas operaciones son complemento, unión e intersección. Sean las etiquetas A y B las que identifican a dos conjuntos borrosos asociados a una variable lingüística x, las operaciones se definen como:

- Complemento $\mu_{\bar{A}}(x) = 1 - \mu_A(x)$
- Unión. Operador lógico OR de Zadeh (max) $\mu_{A \cup B}(x) = \max [\mu_A(x), \mu_B(x)]$
- Intersección. Operador lógico AND de Zadeh (min) $\mu_{A \cap B}(x) = \min [\mu_A(x), \mu_B(x)]$

Hay muchas definiciones para las operaciones lógicas, algunas otras definiciones que normalmente también se utilizan son:

- Operador lógico AND del producto $\mu_{A \cap B}(x) = \mu_A(x) \times \mu_B(x)$
- Operador lógico OR de Lukasiewicz $\mu_{A \cup B}(x) = \max [\mu_A(x) + \mu_B(x), 1]$

Una representación gráfica de lo anterior se muestra en la figura 2.4.

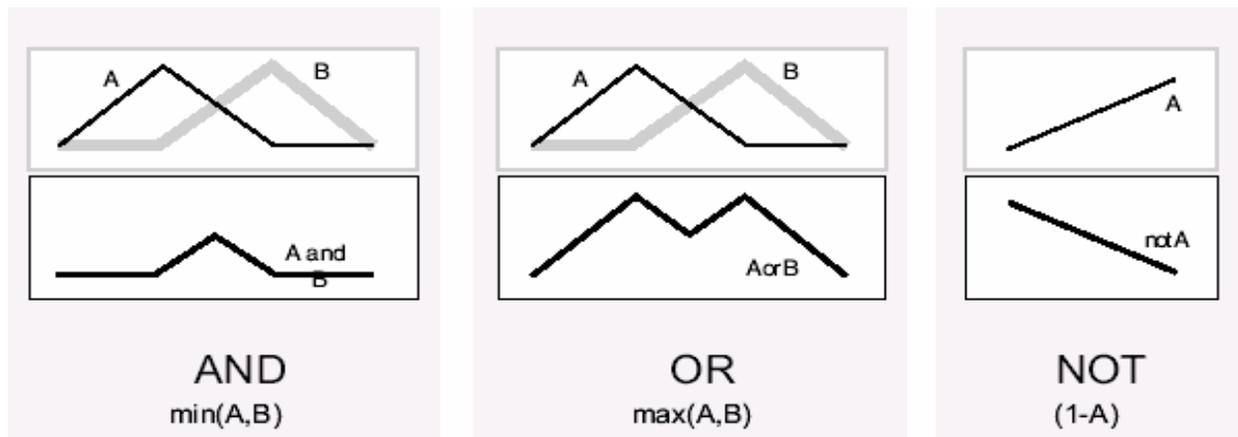


Figura 2.4.- Operaciones Fuzzy.

2.5.1.4 El Controlador Difuso

La lógica difusa se aplica principalmente en sistemas de control difuso que utilizan expresiones ambiguas para formular reglas que controlen el sistema. Un sistema de control difuso trabaja de manera muy diferente a los sistemas de control convencionales. Estos usan el conocimiento experto para generar una base de conocimientos que dará al sistema la capacidad de tomar decisiones sobre ciertas acciones que se presentan en su funcionamiento (Ross, 1995). Los sistemas de control difuso permiten describir un conjunto de reglas que utilizaría una persona para controlar un proceso y a partir de estas reglas generar acciones de control. El control difuso puede aplicarse tanto en sistemas muy sencillos como en sistemas cuyos modelos matemáticos sean muy complejos. La estructura de un controlador difuso se muestra en la figura 2.5.

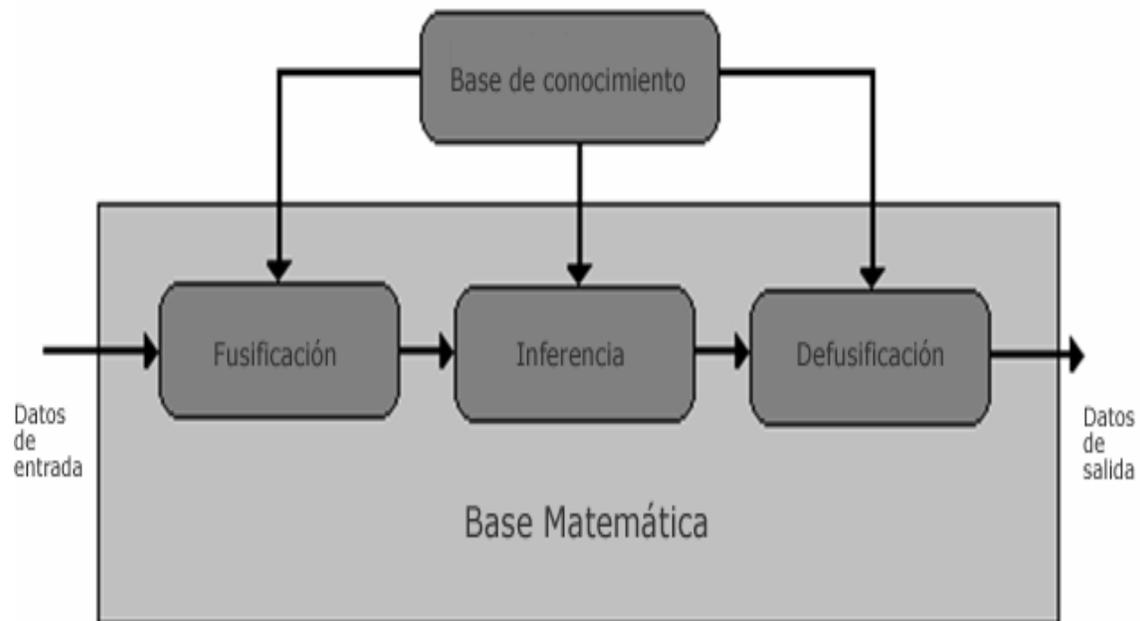


Figura 2.5.- Estructura de un modelo difuso.

2.5.1.5 Fusificación

La fusificación tiene como objetivo convertir valores *crisp* o valores reales en valores difusos. En la fusificación se asignan grados de pertenencia a cada una de las variables de entrada con relación a los conjuntos difusos previamente definidos utilizando las funciones de pertenencia asociadas a los conjuntos difusos (Helmick, 2004). En otras palabras, se trata de procesamiento matemático que toma elementos del universo de discurso (variables de entrada) y las transforma en un valor numérico en cada función de membresía a las cuales pertenece, figura 2.6.

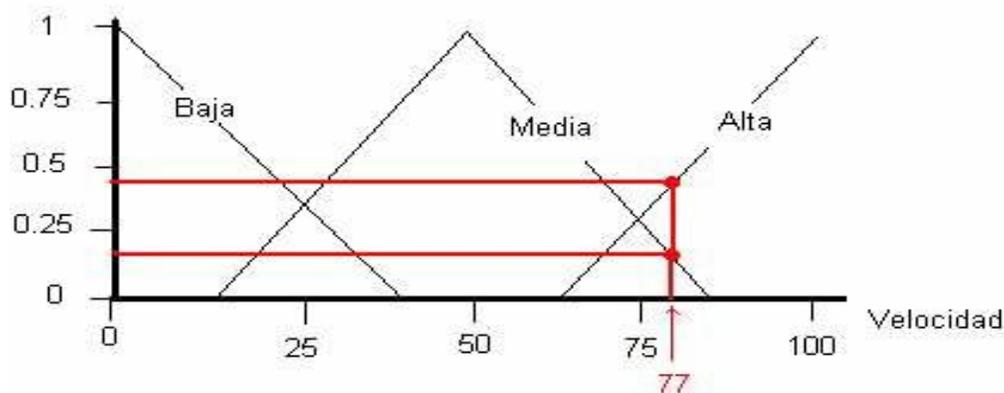


Figura 2.6.- Ejemplo de fusificación de una variable de entrada (velocidad).

2.5.1.6 Base de Conocimiento

La base de conocimiento contiene el conocimiento asociado con el dominio de la aplicación y los objetivos del control. En esta etapa se deben definir las reglas lingüísticas de control que realizarán la toma de decisiones que decidirán la forma en la que debe actuar el sistema (Melin, 2007). Se realiza con el apoyo de un experto del cual nos serviremos de su experiencia del control del proceso que lo expresara en forma lingüística y podemos establecer las reglas que serán evaluadas de forma difusa. Un ejemplo sería el frenado de un vehículo donde el experto nos platicaría como podemos frenarlo en base a la distancia y la velocidad, de ahí obtenemos la base del conocimiento, figura 2.7, que son los datos con los que formaremos las reglas.

DESACELERACION		VELOCIDAD		
		LENTA	MODERADA	RAPIDA
DISTANCIA	MUY CERCA	MODERADO	BASTANTE	TODO
	CERCA	POCO	MODERADO	BASTANTE
	LEJOS	NADA	POCO	MODERADO

Figura 2.7.- base del conocimiento para frenar un vehículo.

2.5.1.7 Inferencia

La inferencia relaciona los conjuntos difusos de entrada y salida para representar las reglas que definirán el sistema. En la inferencia se utiliza la información de la base de conocimiento para generar reglas mediante el uso de condiciones, por ejemplo: *si caso1 y caso2, entonces acción1* (Melin, 2007). Volviendo con el ejemplo de frenado entonces una regla sería: ***SI la velocidad es rápida Y distancia es muy cerca ENTONCES desacelerar todo***

2.5.1.8 Defusificación

La defusificación realiza el proceso de adecuar los valores difusos generados en la inferencia en valores *crisp*, que posteriormente se utilizarán en el proceso de control. En la defusificación se utilizan métodos matemáticos simples como el método del Centroide, Método del Promedio Ponderado y Método de Membresía del Medio del Máximo (Melin, 2007).

Por ejemplo, si optamos por utilizar el método del centroide tendremos que formar un polígono con los datos obtenidos de la pertenencia de la entrada a una o varias funciones de membresía y después encontrar su centro de gravedad el cual será el valor numérico defusificado que podremos aplicar al control, figura 2.8.

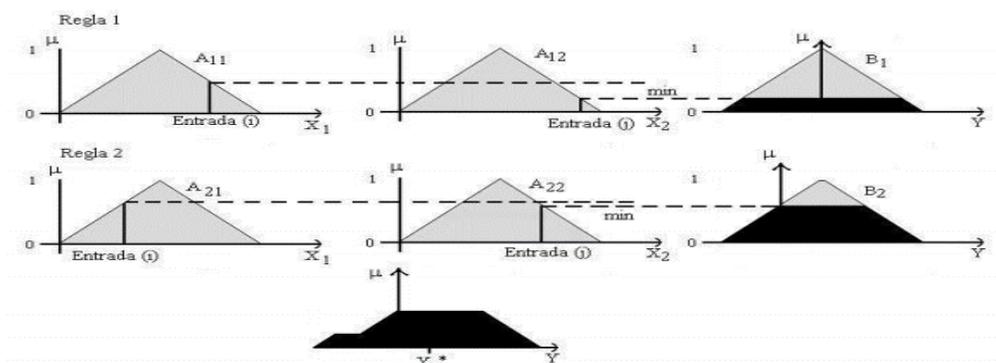


Figura 2.8.- Método del centroide.

2.5.1.9 Definición de características del modelo

En esta etapa se definen las características funcionales y operacionales del modelo el cual es del tipo Mamdani, aquí la tarea del diseñador consiste en definir (aunque se carezca de un modelo matemático del sistema):

- Los datos de entrada al sistema.
- Las transformaciones básicas que se aplicarán a los datos.
- Los datos de salida del sistema.

Definición de conjuntos difusos. - Para el proceso de definición de los conjuntos difusos en sistemas de control, primero se identifican, se nombran las variables de entrada, así como de salida y se establecen sus rangos,

Definición de las reglas. - las reglas de control engloban el conocimiento del sistema y los objetivos de control. Cada regla tiene un estado del sistema en su premisa y una acción de control sugerida en su conclusión.

Las reglas de control difusas conectan los valores de entrada con las propiedades de la salida del modelo. Están expresadas como proposiciones condicionales:

**SI [ESTADO DEL PROCESO]
ENTONCES [ACCION DE CONTROL]**

Cuyo número de reglas se obtiene multiplicando el número de términos difusos de las variables de entrada. Y su base de reglas utiliza una matriz que cubre todas las posibles combinaciones de las entradas.

Defusificación. - Es el proceso de transformar los valores difusos en “*crisp*” (que son aquellos que puede entender el computador). Se utilizan para ello métodos matemáticos simples como el del centroide (ecuación 3.1), el cual calcula el centro de gravedad del polígono que se crea con el método de composición de membresía del máximo mínimo (ecuación 3.2).

$$y = \frac{\sum_i b_i \int u(i)}{\sum_i \int u(i)} \quad 2.3$$

$$\mu_{P \circ Q} = \max \min [\mu_P, \mu_Q] \quad 2.4$$

Simulación y ajuste del sistema. - Cuando el modelo difuso ha sido construido, el proceso de simulación y desarrollo del modelo comienza. La cual puede realizarse en varios paquetes computacionales disponibles en el mercado. Cuando los resultados de la simulación o las pruebas físicas no son satisfactorios se realizan ajustes en las descripciones de los conjuntos difusos o en las acciones de control sugeridas por las reglas, hasta afinar el desempeño del control.

2.6 Selección de Sensores

Los sensores que se proponen son los siguientes:

Sensor QRD1114.- Es del tipo óptico y cuya función es detectar la cantidad de luz reflejada por una superficie. El sensor está conformado por un LED emisor infrarrojo, el cual genera luz en el espectro infrarrojo, este emisor hace rebotar la señal sobre la superficie que se encuentra sensando mientras que un optotransistor es el encargado de recibir la luz reflejada. Este sensor posee un filtro de luz con la finalidad de evitar perturbaciones por luz ambiente (College, 2012).

Una de las aplicaciones del sensor es en seguidores de línea. Que tienen la función de seguir una línea de color negro o blanco sin salirse del camino marcado y en el menor tiempo posible, esta característica se muestra en la siguiente figura 2.9.

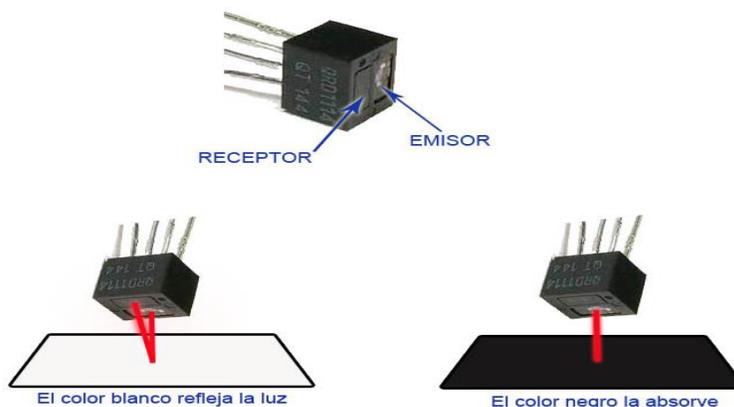


Figura 2.9. - Aplicación del sensor QRD1114 en la detección de líneas.

Los anteriores sensores serán implementados para que realicen la función de los etiquetados como: **SC**, **S1**, **S2** y **S3**.

Sensor Ultrasónico HC-SR04.- El Sensor Ultrasónico HC-SR04 puede llegar a medir una distancia desde los 2 Cm hasta los 450 Cm en óptimas condiciones, esto se logra gracias a la emisión de un sonido imperceptible por el oído humano, sin embargo, el sensor detecta cuánto tarda la señal en regresar y en base a ello puede calcular la proximidad con un objeto, figura 2.10. Esta característica será aprovechada para detectar obstáculos en la trayectoria del robot.



Figura 2.10. - Aplicación del sensor HC-SR04 en la detección de objetos.

Los anteriores sensores serán implementados para que realicen la función de los etiquetados como: **S4** y **S5**.

Microinterruptor o final de Carrera Modelo SEN-FDCM. - También conocido como "interruptor de límite", son dispositivos eléctricos, neumáticos o mecánicos situados al final del recorrido o de un elemento móvil, como por ejemplo una cinta transportadora, con el objetivo de enviar señales que puedan modificar el estado de un circuito. Se accionan cuando se oprime un pequeño pulsador, una palanca o un pequeño rodillo, figura 2.11.



Figura 2.11. - Microinterruptor SEN-FDCM y funcionamiento.

Su implementación será para que realicen la función de los etiquetados como: **SIF**, **SDF**, **SIT**, **SDT**, **SCF** y **SCT**.

2.7 Selección de Actuadores

Para lograr el desplazamiento del robot se debe de implementar los actuadores que permitan su movimiento y su cambio de dirección. Para ello se opta por aconsejar el uso de reductores de velocidad o denominados motorreductores.

Los cuales constan con uno o varios pares de engranajes que ajustan la velocidad y la

potencia mecánica de aparatos y máquinas que funcionan con un motor, y que precisan que la velocidad del motor se adapte a la velocidad para que funcione perfectamente la máquina. Un ejemplo es el usado en los elevallunas eléctricos de los automóviles los cuales tienen un alto torque y pueden ser controlados a través de un puente H de potencia como el BTS7960 de 43A figura 2.12.



Figura 2.12.- Motorreductor del tipo elevallunas y puente H BTS7960.

Los anteriores actuadores serán implementados para que realicen la función de los etiquetados como: **M1**, **M2**, **M3**, **M4**, **MD1** y **MD2**.

2.8 Selección de Electrónica de control.

Se propone el uso de una placa de desarrollo Arduino que es una plataforma abierta que facilita la programación de un microcontrolador. Los cuales cuentan con sensores para escuchar el mundo físico y salidas digitales/análogas (pwm) que permiten activar actuadores para interactuar con el mundo físico (J. P. , 2009).

Cuenta además con tres interesantes ventajas:

1. Una facilidad de programación al contar con su entorno de desarrollo y un lenguaje sencillo.

2. Puede ser usado como tarjeta de adquisición de datos.
3. Es compatible con **MATLAB®** y puede ser programado directamente en este software.

El modelo propuesto es el **Arduino Mega2560** se muestra en la siguiente figura 2.13, cuyas principales características son:

- Posee 54 pines digitales que funcionan como entrada/salida; 16 entradas análogas,
- Un cristal oscilador de 16 MHz,
- Una conexión USB, un botón de reset y una entrada para la alimentación de la placa.
- Posee un convertidor usb-serie, por lo que sólo se necesita conectar el dispositivo a la computadora utilizando un cable USB como el que utilizan las impresoras.



Figura 2.13. – Arduino Mega 2560.

3.- MÉTODO

La lógica de control a utilizar es del tipo difusa, la cual es una rama de la inteligencia artificial que le permite a una computadora analizar información del mundo real entre una escala de lo falso y lo verdadero. Dado que una computadora entiende el mundo en base a ceros y unos, trabaja con datos discretos que pueden tomar solo un valor dado y no aceptan ambigüedades, no entiende que es poco, medio, mucho, más o menos y otras sentencias que no arrojan un dato preciso, sino que depende de una valoración dada por alguien a quien se denomina experto (Mamdani, Application of fuzzy algorithms for control of simple dynamic plant, 1974). Por ello se hace uso de la lógica difusa que permite darle un valor numérico dentro de un rango a tales sentencias ya que poco podría ser de 0 a 4, medio de 3 a 7 y mucho de 6 a 1 y se les conoce como variables lingüísticas. De esta manera si el dato a analizar es 2 se puede decir que es poco, sin embargo, se puede dar el caso de que algunos valores no estén tan claros en que ponderación caben y es aquí donde se introduce el término “grado de pertenencia” a través de funciones que lo determinan. Estas funciones se denominan de pertenencia y se denotan por el símbolo μ las cuales tienen un valor de 0 a 1 en el eje de las ordenadas y puede tomar valores en el eje de las abscisas que están determinados por los elementos totales a los cuales se hace referencia y lo cual se conoce como universo discurso

Por lo tanto, en los sistemas de control basados en lógica difusa se han de combinar variables de entrada, definidas en términos de conjuntos difusos, por medio de grupos de reglas que producen uno o varios valores de salida.

3.1 Ubicación y función de sensores / actuadores.

La función de los sensores se describe a continuación

- **S1, S2 Ys3.-** Se propone colocarlos en el centro del área del chasis y en posición horizontal, su función es mantener en ruta al robot al detectar que se sale de la línea negra indicando si fue a la izquierda y a la derecha.
- **S4 y S5.-** Ubicados uno al frente y el otro en la parte trasera del chasis, su función es detectar si un objeto obstruye la trayectoria del robot.

La función de los actuadores se describe a continuación:

- **M1, M2, M3 Y M4.-** Son los motores de desplazamiento los encargados de imprimir movimiento al robot que le permitan desplazarse hacia adelante, atrás e incluso frenarse.
- **MD1 y MD2.-** Son los motores de dirección y su función es de mover hacia la izquierda o derecha el eje donde se encuentran dos motores de desplazamiento.

Una vez que se definió la función los actuadores el siguiente paso sería el colocar sensores que nos aporten información adicional a los ya instalados. Lo cual nos permitirá usarla en la función de prueba del dispositivo así como si existe un paquete a entregar, en la figura 3.1 se muestra la ubicación de los sensores y actuadores. La etiqueta de identificación y función que realizan es la siguiente:

- **SIF y SIT.-** Estos sensores identifican el giro completo a la izquierda de la parte frontal y trasera del chasis.
- **SDF y SDT.-** Estos sensores identifican el giro completo a la derecha de la parte frontal y trasera del chasis.
- **SCF y SCT.-** Estos sensores identifican la posición central de la parte frontal y trasera del chasis.
- **SC.-** Este sensor identifica si existe un paquete presente.

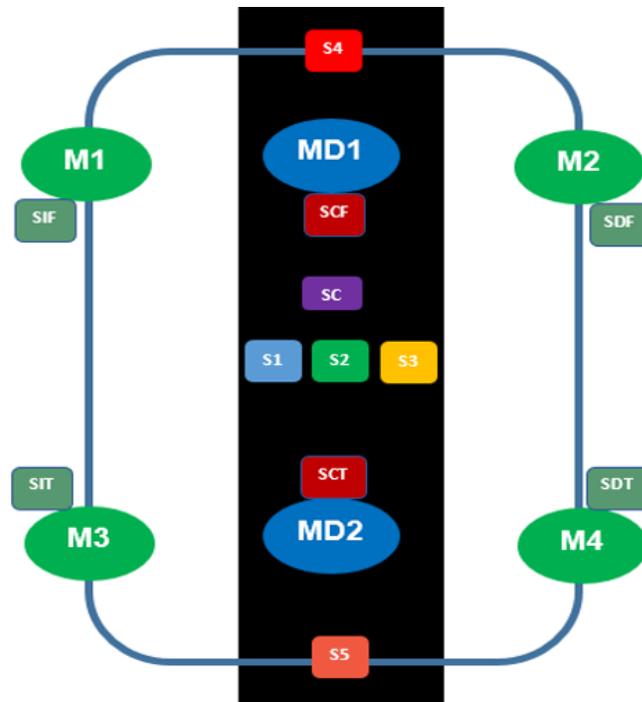


Figura 3.1.- Ubicación de sensores de pruebas y presencia de paquete.

Otro punto a tener en cuenta es que el vehículo debe saber si está en el punto de partida y a final de la ruta de reparto, es necesario que pueda identificar en que punto se encuentra y que tarea va realizar. Las cuales serían:

- **Entrega.-** Debe de identificar si va realizar una entrega.
- **Pedir el paquete.-** Debe de identificar la presencia de un paquete.
- **Entregar el Paquete.-** Llegar al punto de entrega o B, comunicar que llegó y solicitar que recogan el paquete.
- **Volver a la base.-** Una vez entregado el paquete debe retornar a la base o punto A.
- **Identificar en que punto se encuentre.-** Debe de identificar si esta en el punto A o B.
- **Debe de seguir una ruta preestablecida.-** Se marca una trayectoria que va del punto A o B, la cual debe de seguir sin desviarse.
- **Identificar obstáculos.-** Debe de reconocer que existe un obstáculo, detenerse y proseguir hasta que deje de estar presente.

- **Debe de realizar auto prueba.-** Debe de solicitar que se posicione en el punto A o B y realizar un auto diagnóstico de sensores y actuadores.
- **Comunicarse con el usuario.-** Debe de desplegar información pertinente como pedir paquetes, mostrar falla, solicitar que recojan el paquete.

Lo anterior lo realizará mediante un programa que combine técnicas convencionales de programación e inteligencia artificial con el uso de un controlador difuso para mantenerse en ruta.

Para identificar el punto se deben de colocar en un área de cada uno, un sensor y una obstrucción para el sensor de obstáculo, donde:

- **SPA.-** sensor que indique que estoy en el punto A.
- **SPB.-** sensor que indique que estoy en el punto B.

Como se muestra en la siguiente figura 3.2:

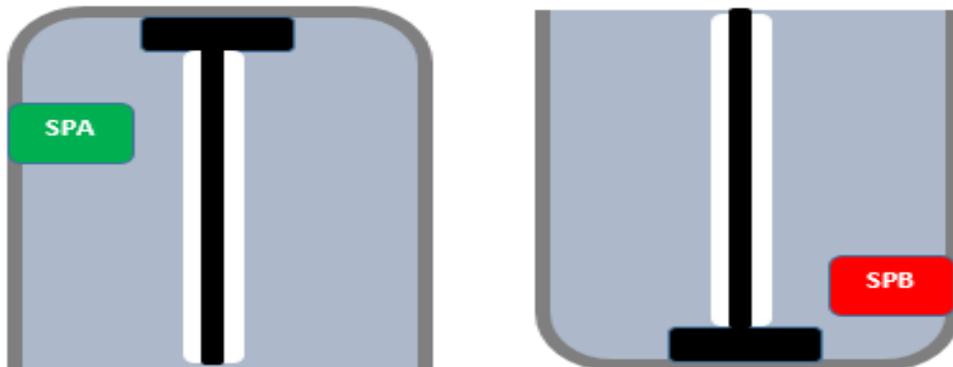


Figura 3.2.- Ubicación de sensor y punto de control del sensor de obstáculo, en las áreas de partida (punto A) y llegada (punto B).

3.2 Diseño de la ruta

La ruta de trabajo del robot está dentro de las instalaciones del ITESCA, la cual sigue una trayectoria que lo guiará del punto A al punto B y viceversa, la cual se puede apreciar en la figura 3.3.

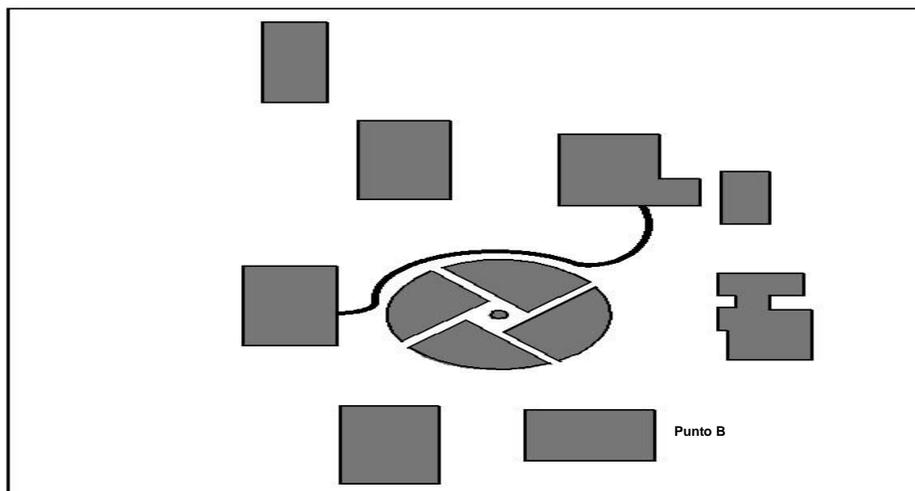


Figura 3.3.- Trayectoria de la ruta dentro de las instalaciones del ITESCA.

Se definió que la trayectoria se “dibuja” mediante una línea blanca que contiene una línea negra que mantendrá al robot dentro de ruta, figura 3.4, donde:

Línea Central. - Su función es la de control principal.

Línea Blanca. - Indica si se ha salido de la línea central

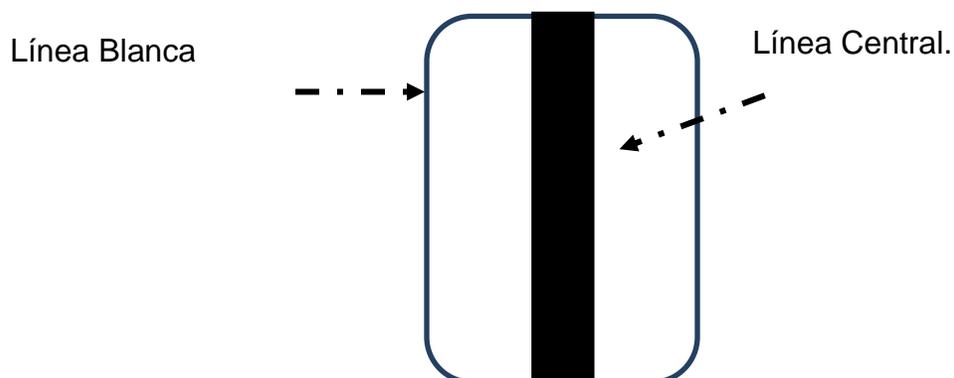


Figura 3.4.- Líneas para mantener dentro de ruta al robot.

3.2 Diseño propuesto del robot

El robot debe ser capaz de seguir una línea describiendo una trayectoria del punto A al punto B, Detectar obstáculos y mantener una velocidad constante. Un esquema básico se aprecia en la figura 3.5.

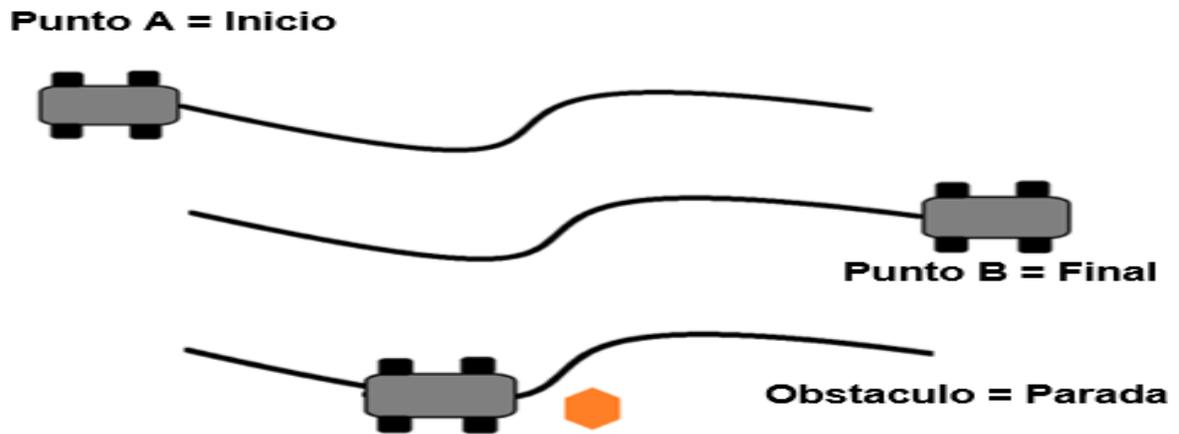


Figura 3.5.- Función del robot.

3.5 Diseño de control.

Las funciones que realizara el vehiculo son comandadas a traves de código de programación que incluye un controlador difuso,cuyo digrama de flujo se muestra a continuacion en las siguiente figura:

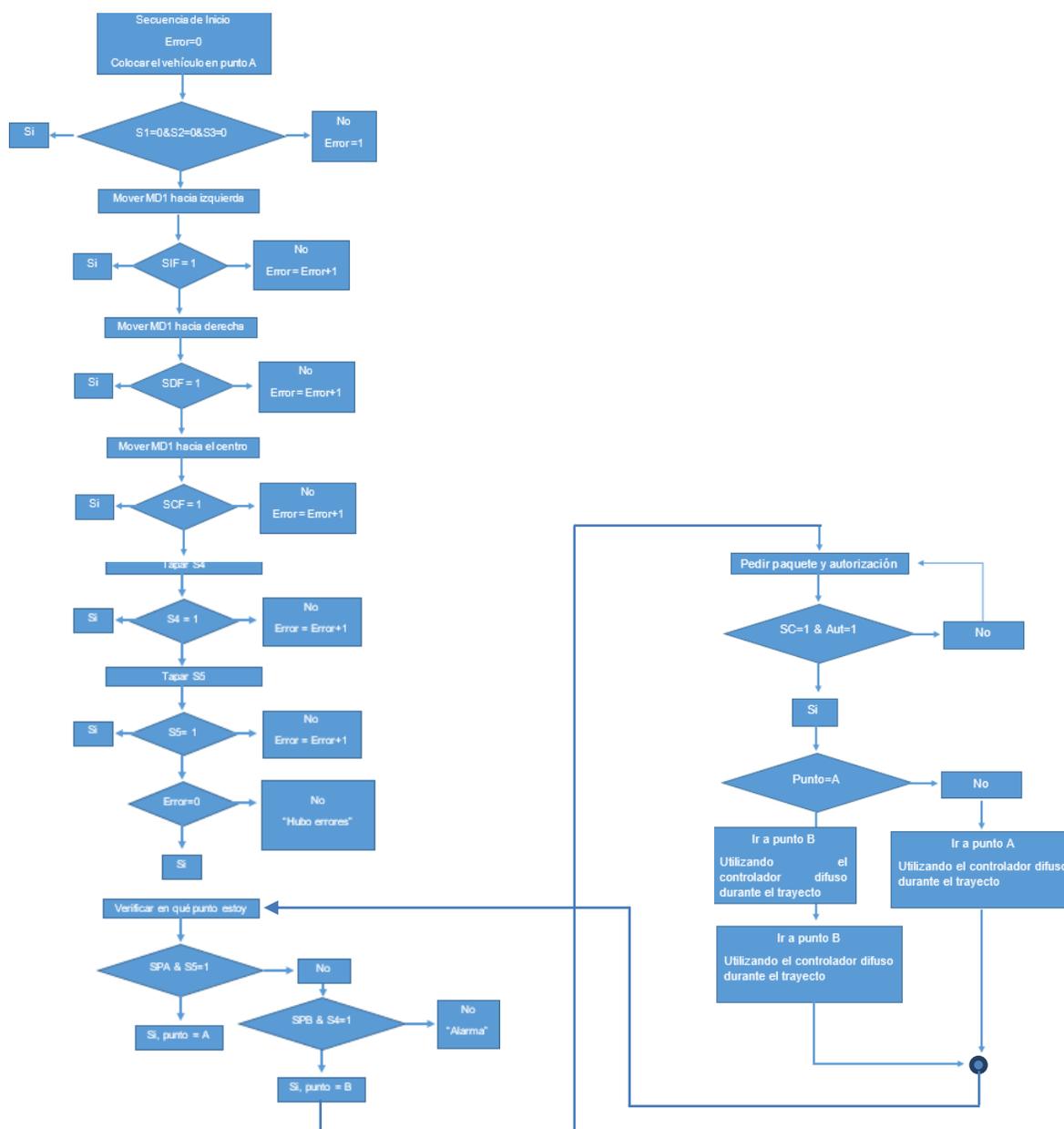


Figura 3.6.- Diagrama de flujo.

3.6 Diseño del controlador difuso

Para lograr el objetivo deseado de controlar a través de lógica difusa el robot se siguió el siguiente ciclo para su diseño, figura 3.7:

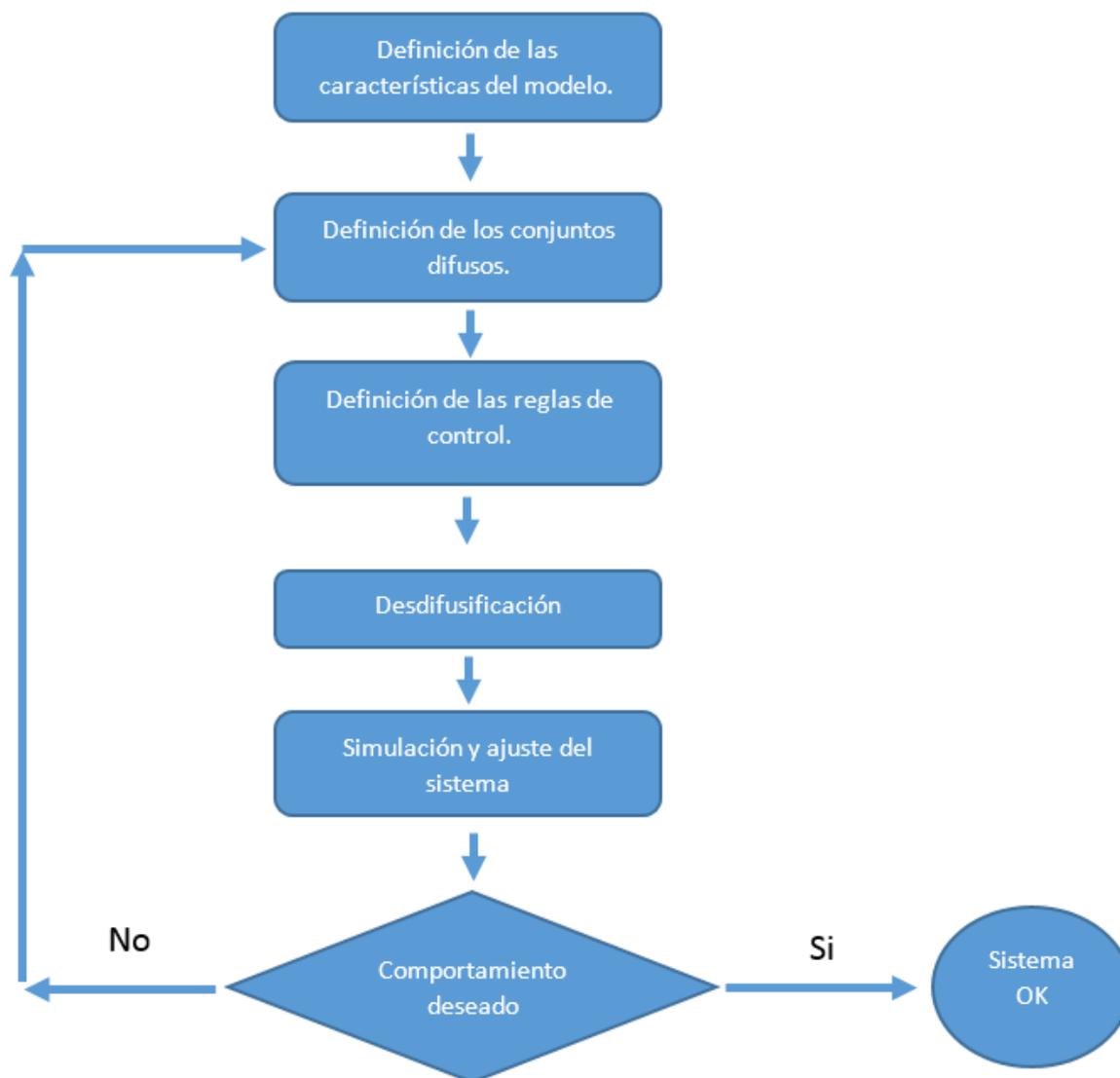


Figura 3.7. – Ciclo de diseño del controlador difuso.

Programación del controlador difuso. - Se inicia identificando los siguientes elementos:

Entradas. – Las variables que se miden son datos que resultan de procesar las señales provenientes de los sensores ópticos y ultrasónicos. Las cuales se han etiquetado como:

- **SOI.-** Señal de posición moviéndose a la izquierda.
- **SOD.-** Señal de posición moviéndose a la derecha.
- **SCENT.-** Señal de posición centrada.
- **SDO.-** Señal de detección de obstáculos.

Universo discurso de la entrada. – Los niveles de intensidad para SOI, SOD y SCENT va de 0 a 1 y el de SDO va de 0 a 100.

Funciones de membresía de la entrada. – El universo discurso se ha dividido en las siguientes funciones de membresía, figura 3.8, para SOI, SOD y SCENT son las siguientes:

- **N.-** Negro, función trapezoidal cuyos parámetros son: -0.6 -0.11 0.05 0.1.
- **CN.-** Casi Negro, función triangular cuyos parámetros son: 0.05 0.15 0.25.
- **GO.-** Gris Oscuro, función triangular cuyos parámetros son: 0.2 0.3 0.4.
- **G.-** Gris, función triangular cuyos parámetros son: 0.35 0.5 0.65.
- **GC.-** Gris Claro, función triangular cuyos parámetros son: 0.6 0.7 0.8.
- **CB.-** Casi Blanco, función triangular cuyos parámetros son: .0.75 0.85 0.95.
- **B.-** Blanco, función trapezoidal cuyos parámetros son: 0.9 0.95 1.055 1.375

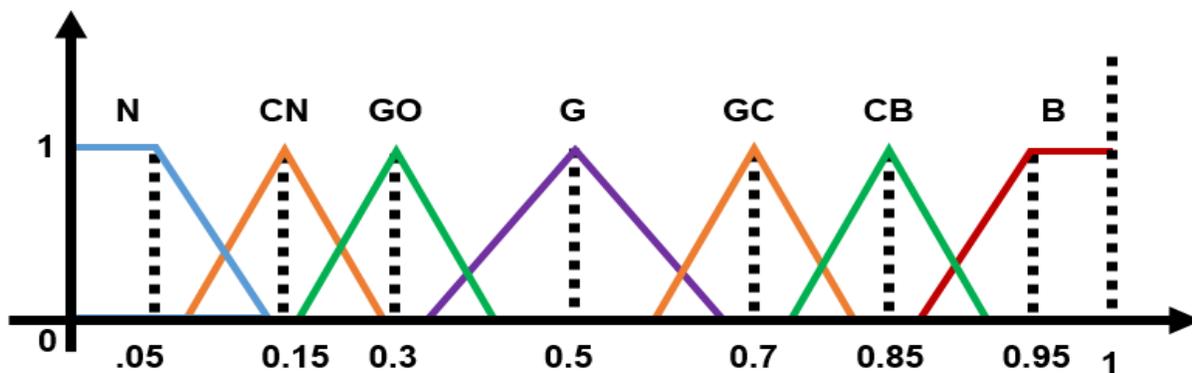


Figura 3.8. - Funciones de membresía de SOI, SOD y SCENT.

Para la variable SDO El universo discurso se ha dividido en las siguientes funciones de membresía, figura 3.9:

:

- **SO.-** Sin Objeto, función trapezoidal cuyos parámetros son: -36.7 -4.7 5 15.
- **OD.-** Objeto Distante, función triangular cuyos parámetros son: 10 20 30.
- **ACER.-** Objeto Acercándose, función triangular cuyos parámetros son: 25 35 45.
- **OMD.-** Objeto a Media Distancia, función triangular cuyos parámetros son: 35 50 65.
- **OCER.-** Objeto Cerca, función triangular cuyos parámetros son: 60 70 80.
- **OMCER.-** Objeto Muy Cerca, función triangular cuyos parámetros son: 75 85 95.
- **OINM.-** Objeto Inminente, función trapezoidal cuyos parámetros son: 85 95 104 136.

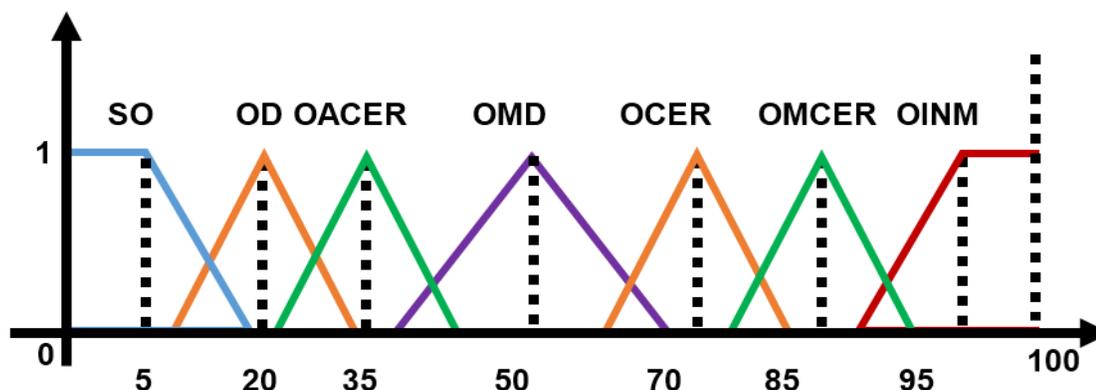


Figura 3.9.- Funciones de membresía de SDO.

Salidas.- Estas variables son datos que han de procesarse para controlar los actuadores. Las cuales se han etiquetado como: **Dirección y Velocidad.**

Universo discurso de la salida. – Los rangos de valores para Dirección van de -1 a 1 y los de Velocidad va de 0 a 100.

Funciones de membresía de las variables de salida.- El universo discurso de Dirección se ha dividido en las siguientes funciones de membresía, figura 3.10:

- **I.-** Izquierda, cuya función es trapezoidal cuyos parámetros son: -1.72 -1.08 -0.8 -0.6.
- **CI.-** Casi Izquierda, cuya función es triangular cuyos parámetros son: -0.7 -0.5 - 0.3.
- **C.-** Centro, cuya función es trapezoidal cuyos parámetros son: -0.4 -0.2 0.2 0.4.
- **CD.-** Casi Derecha, cuya función es triangular cuyos parámetros son: 0.3 0.5 0.7.
- **D.-** Derecha, cuya función es trapezoidal cuyos parámetros son 0.6 0.8 1.094 1.734.

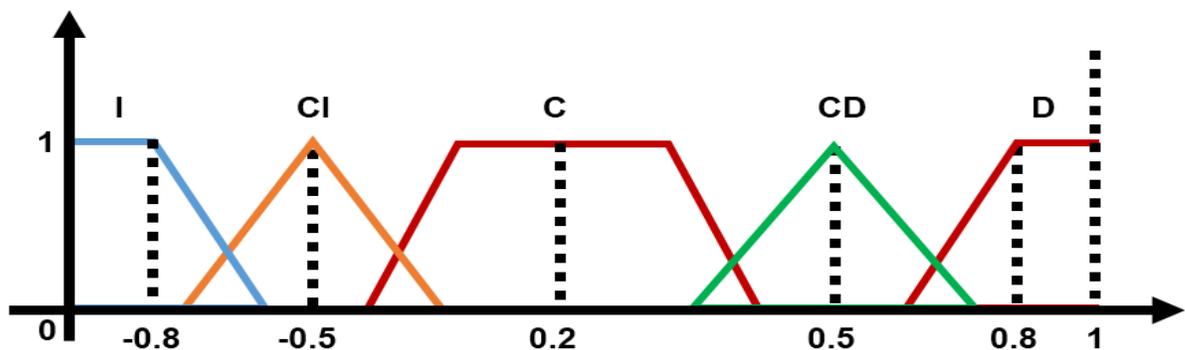


Figura 3.10. - Funciones de membresía de DIRECCION.

Para Velocidad el universo discurso se ha dividido en las siguientes funciones de membresía figura 3.11:

- **Frenado.**- Frenado, cuya función es trapezoidal cuyos parámetros son: -35.82 - 3.824 10 20.
- **VELMBAJA.**- Velocidad Muy Baja, cuya función es triangular cuyos parámetros son: 15 25 35.
- **VELBAJA.**- Velocidad Baja, cuya función es triangular cuyos parámetros son: 25 35 45.
- **VEMED.**- Velocidad Media, cuya función es triangular cuyos parámetros son: 40 50 60.
- **VELALTA.**- Velocidad Alta, cuya función es triangular cuyos parámetros son: 55 65 75.
- **VELMALTA.**- Velocidad Muy Alta, cuya función es triangular cuyos parámetros son: 70 77 85.
- **VELMAX.**- Velocidad Máxima, cuya función es trapezoidal cuyos parámetros son: 80 90 104 136.

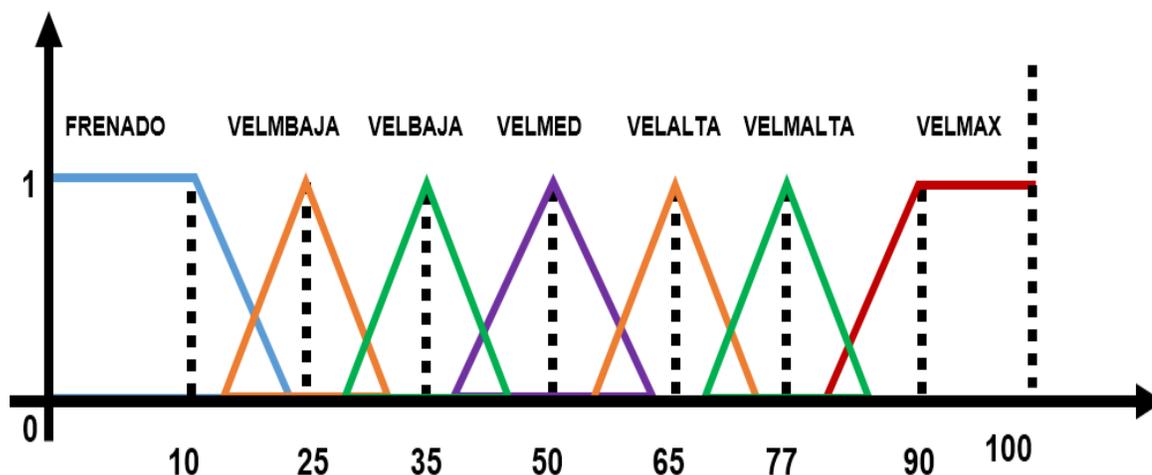


Figura 3.11.- Funciones de membresía de VELOCIDAD.

Variables lingüísticas. - Las reglas **si evento₁ y evento₂ y ... evento entonces acción**, teniendo presente la ubicación y función de los sensores, variables de entrada, los actuadores, variables de salida. Se puede establecer las reglas **“si...entonces...”** algunas de ellas son:

1. **Si S1 no detecta la línea negra y S2 si lo hace y S3 también entonces gira un poco a la derecha.**
2. **Si S1 no detecta la línea negra y S2 no detecta la línea negra y S3 si lo hace entonces gira a la derecha.**
3. **Si S4 encuentra un objeto lejano entonces disminuye un poco la velocidad.**
4. **Si S4 encuentra un objeto a media distancia entonces disminuye a velocidad media.**
5. **Si S4 encuentra un objeto inminente entonces la velocidad es igual cero.**

4. Software utilizado

Para su programación se hizo uso del software **MATLAB®**, en el cual se eligió un modelado de un sistema de inferencia difusa **FIS** por sus siglas en inglés: **Fuzzy Inference System**. La inferencia difusa es el proceso de formular asignaciones de entrada / salida utilizando una lógica difusa. El software **Fuzzy Logic Toolbox™** es una aplicación para crear sistemas difusos Mamdani y Sugeno donde el usuario puede definir el número de entradas y el método de defuzzificación (Gilat A. , 2005).

Donde se realizaron los siguientes pasos:

1.- Se añadieron las entradas y se etiquetaron como SISQ, SDER, SCENT y SDO, figura 4.2.

2.- Se eligió el controlador de Mamdani.

3.- Se añadieron las entradas y se etiquetaron como DIRECCION y VELOCIDAD.

4.- Se eligió el método del centroide para la defuzzificación.

5.- Quedando el diseño mostrado en la figura 4.1:

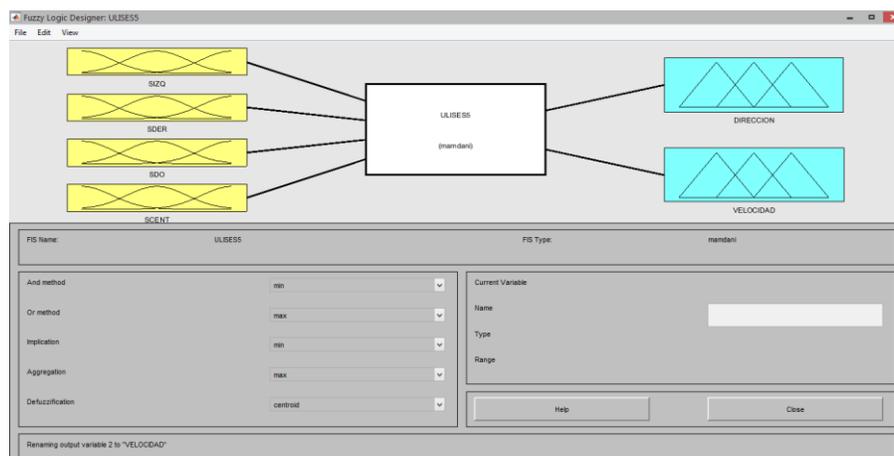


Figura 4.1. - Modelo FIS

6.- Se definió las funciones de membresía de la entrada SIZQ figura 4.2.

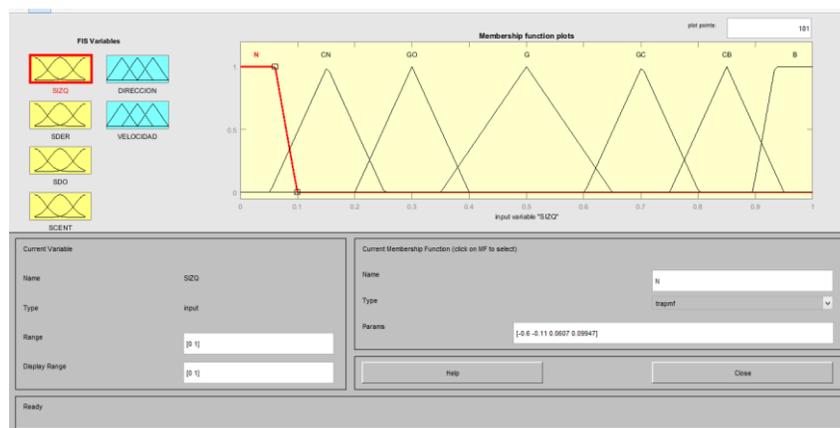


Figura 4.2. – Funciones de membresía de la entrada SIZQ.

7.- Se definió las funciones de membresía de la entrada SCENT, figura 4.3.

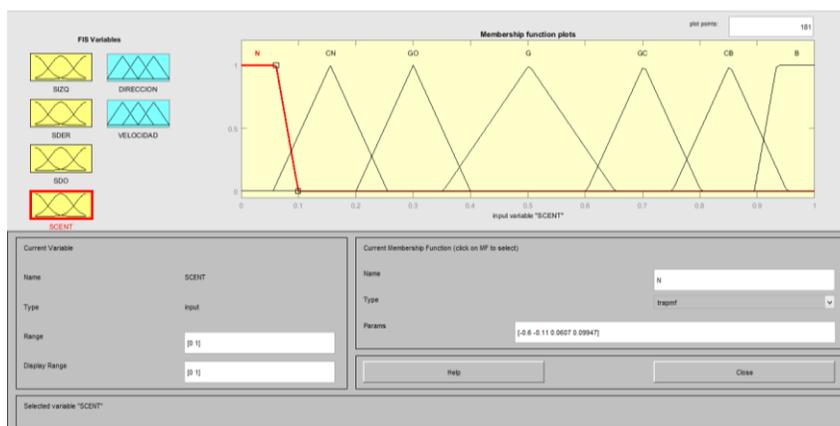


Figura 4.3. – Funciones de membresía de la entrada SCENT.

8.- Se definió las funciones de membresía de la entrada SDER, figura 4.4.

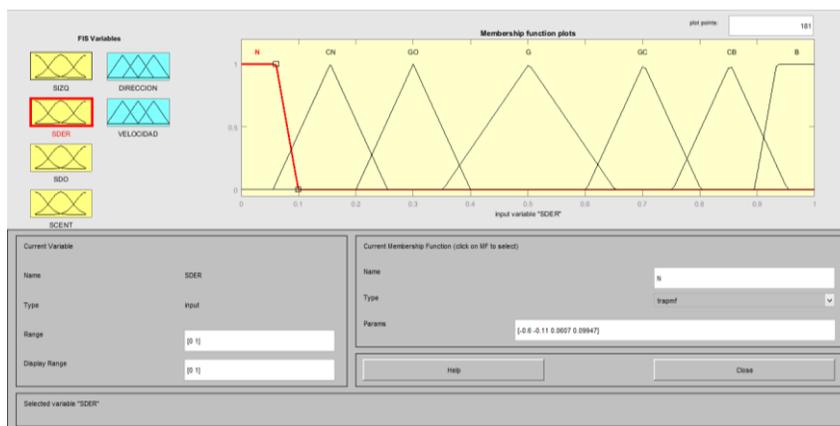


Figura 4.4. – Funciones de membresía de la entrada SDER.

9.- Se definió las funciones de membresía de la entrada SDO, figura 4.5.

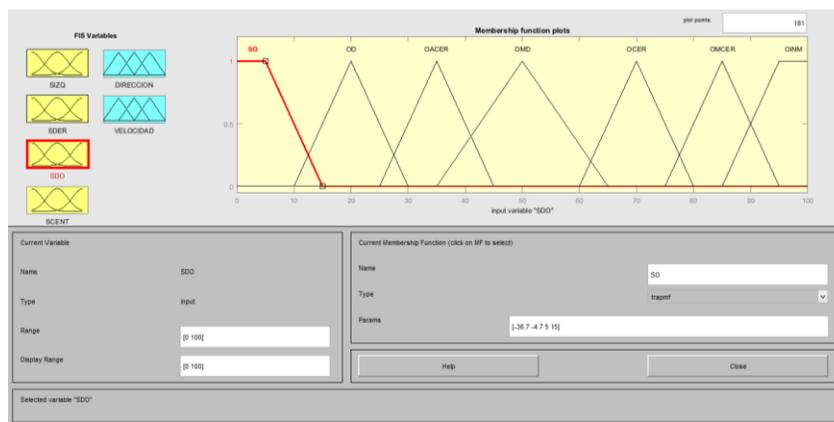


Figura 4.5. – Funciones de membresía de la entrada SDO.

10.- Se definió las funciones de membresía de la salida DIRECCION, figura 4.6.

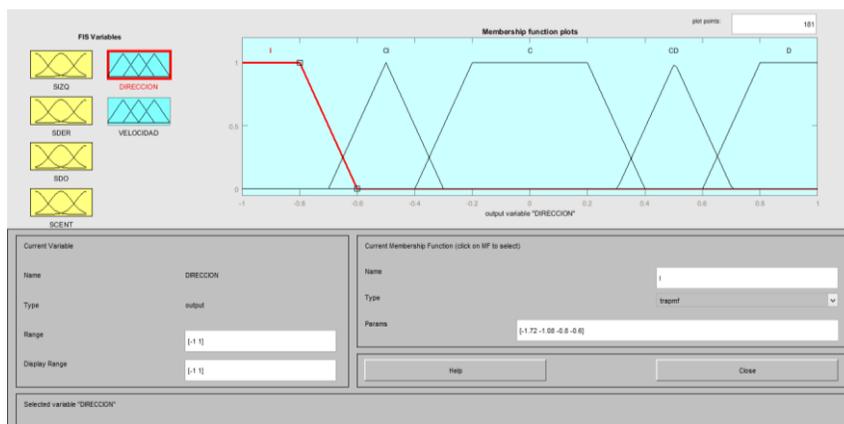


Figura 4.6. – Funciones de membresía de la salida DIRECCION.

11.- Se definió las funciones de membresía de la salida VELOCIDAD, figura 4.7.

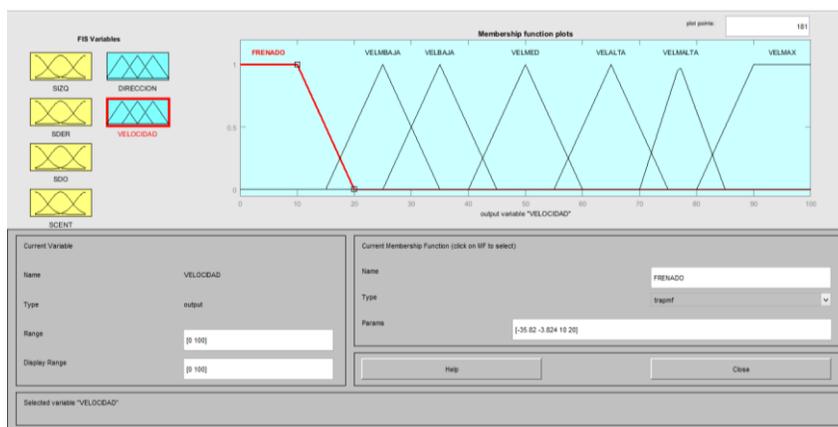


Figura 4.7. – Funciones de membresía de la salida VELOCIDAD.

12.- Se definió un conjunto de reglas **si...entonces** de 36 enunciados que se muestran en la siguiente tabla:

Numero	Regla "si...entonces..."
1	If (SIZQ is N) and (SDER is N) and (SCENT is N) then (DIRECCION is C)
2	If (SIZQ is CN) and (SDER is N) and (SCENT is N) then (DIRECCION is CD)
3	If (SIZQ is G) and (SDER is N) and (SCENT is N) then (DIRECCION is CD)
4	If (SIZQ is GC) and (SDER is N) and (SCENT is N) then (DIRECCION is D)
5	If (SIZQ is CB) and (SDER is N) and (SCENT is N) then (DIRECCION is D)
6	If (SIZQ is B) and (SDER is N) and (SCENT is N) then (DIRECCION is D)
7	If (SIZQ is CN) and (SDER is CN) and (SCENT is N) then (DIRECCION is CD)
8	If (SIZQ is GO) and (SDER is CN) and (SCENT is N) then (DIRECCION is D)
9	If (SIZQ is G) and (SDER is CN) and (SCENT is N) then (DIRECCION is D)
10	If (SIZQ is GC) and (SDER is CN) and (SCENT is N) then (DIRECCION is D)
11	If (SIZQ is CB) and (SDER is CN) and (SCENT is N) then (DIRECCION is D)
12	If (SIZQ is B) and (SDER is CN) and (SCENT is N) then (DIRECCION is D)
13	If (SIZQ is B) and (SDER is CN) and (SCENT is CN) then (DIRECCION is D)
14	If (SIZQ is G) and (SDER is CN) and (SCENT is CN) then (DIRECCION is D)
15	If (SIZQ is GC) and (SDER is CN) and (SCENT is CN) then (DIRECCION is D)
16	If (SIZQ is CB) and (SDER is CN) and (SCENT is CN) then (DIRECCION is D)
17	If (SIZQ is B) and (SDER is CN) and (SCENT is CN) then (DIRECCION is D)
18	If (SIZQ is GO) and (SDER is GO) and (SCENT is GO) then (DIRECCION is D)
19	If (SIZQ is G) and (SDER is GO) and (SCENT is GO) then (DIRECCION is D)
20	If (SIZQ is GC) and (SDER is GO) and (SCENT is GO) then (DIRECCION is D)
21	If (SIZQ is CB) and (SDER is GO) and (SCENT is GO) then (DIRECCION is D)
22	If (SIZQ is B) and (SDER is GO) and (SCENT is GO) then (DIRECCION is D)
23	If (SIZQ is GC) and (SDER is GC) and (SCENT is GO) then (DIRECCION is D)
24	If (SIZQ is CB) and (SDER is GC) and (SCENT is GO) then (DIRECCION is D)
25	If (SIZQ is B) and (SDER is GC) and (SCENT is GO) then (DIRECCION is D)
26	If (SIZQ is B) and (SDER is G) and (SCENT is GO) then (DIRECCION is D)
27	If (SIZQ is B) and (SDER is G) and (SCENT is GO) then (DIRECCION is D)

28	If (SIZQ is B) and (SDER is B) then (DIRECCION is D)
29	If (SIZQ is B) and (SDER is B) then (DIRECCION is D)
30	If (SIZQ is B) and (SDER is B) then (DIRECCION is D)
31	If (SDO is OINM) then (VELOCIDAD is FRENADO)
32	If (SDO is OMCER) then (VELOCIDAD is VELBAJA)
33	If (SDO is OCER) then (VELOCIDAD is VELBAJA)
34	If (SDO is OACER) then (VELOCIDAD is VELBAJA)
35	If (SDO is OMD) then (VELOCIDAD is VELMED)
36	If (SDO is SO) then (VELOCIDAD is VELMAX)

Tabla 4.1– Reglas si...entonces del modelo FIS.

13.- Se graficaron los comportamientos de las salidas respecto a las salidas que se muestran en las siguientes figuras 4.8-11:

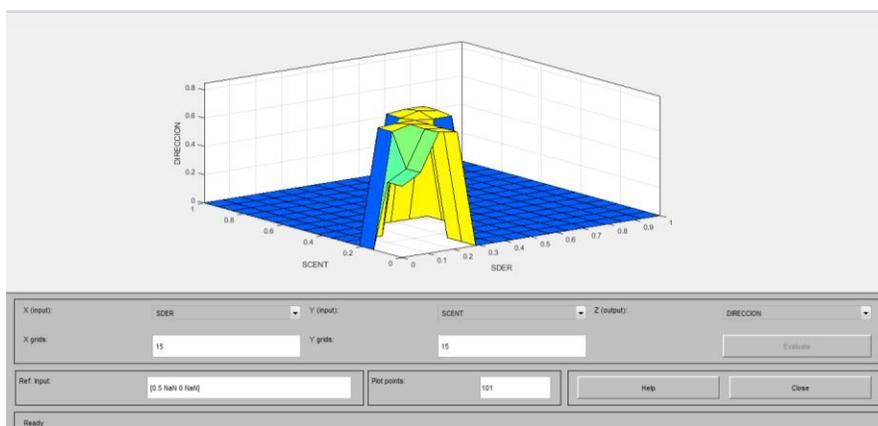


Figura 4.8. Gráfica del comportamiento de las entradas SCENT, SDER y la salida DIRECCION.

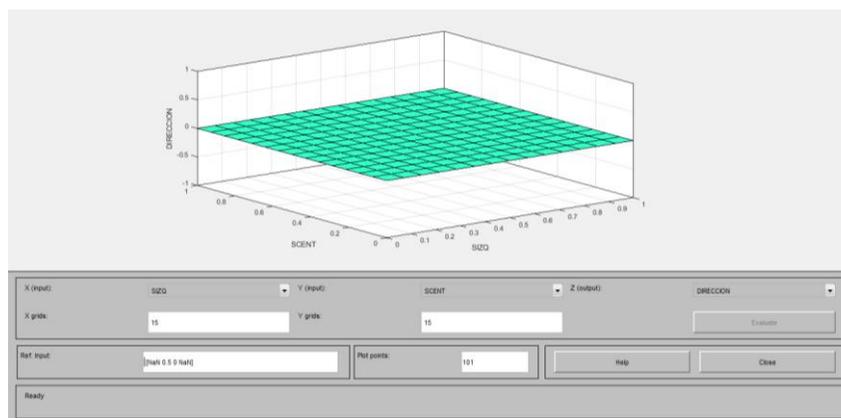


Figura 4.9. Gráfica del comportamiento de las entradas SCENT, SIZQ y la salida DIRECCION.

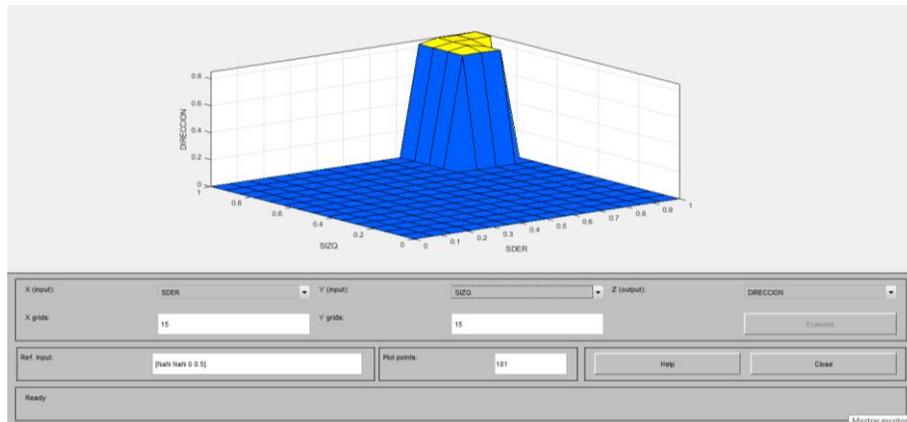


Figura 4.10. Gráfica del comportamiento de las entradas SIZQ, SDER y la salida DIRECCION.

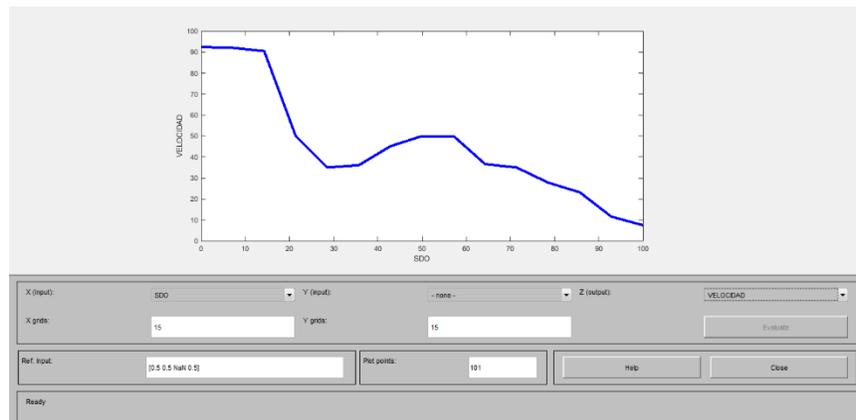


Figura 4.11.- Gráfica del comportamiento de la entrada SDO y la salida VELOCIDAD.

14.- Se observó el comportamiento de las salidas al dar valores a las entradas que corresponderían con datos obtenidos en posibles situaciones reales, el cual se muestra a continuación.

Primer caso en que SIZQ, SDER y SCENT detecten la línea negra y SDO no detecte obstáculos sus valores serían aproximadamente 0, 0, 0, 0 y 0 respectivamente. Por lo tanto, la acción sería mantener paralelas las llantas para seguir una línea recta (dirección centrada) y SDO no detecte ningún obstáculo entonces la velocidad debe ser

la máxima. Lo cual se aprecia en la siguiente figura 4.12:

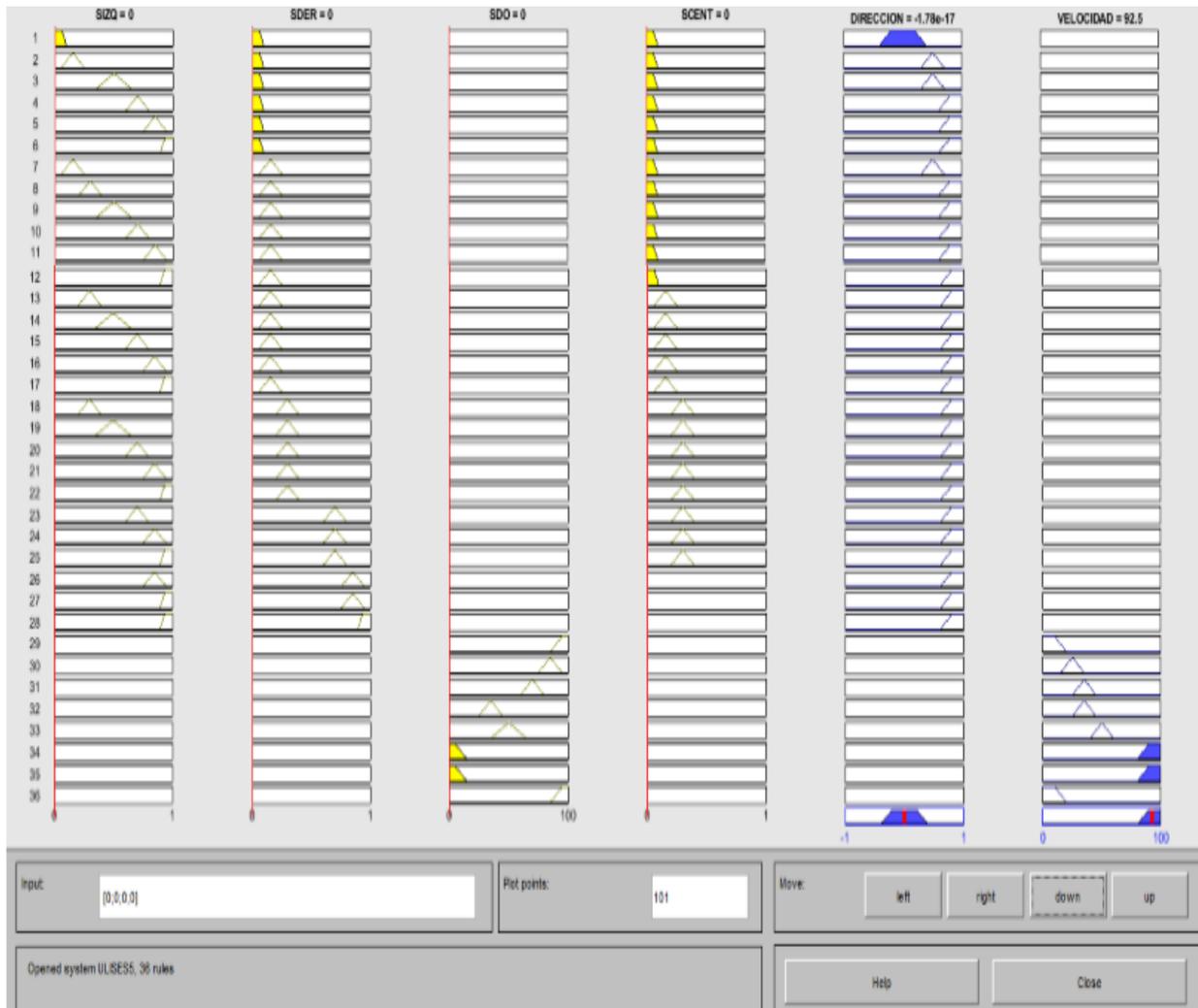


Figura 4.12.- El vehículo va sobre la línea negra.

Segundo caso el vehículo pierde la dirección tendiendo a ir al lado izquierdo y no existe obstáculos por lo que SIZQ, SDER, SDO y SCENT toman los siguientes valores: 0.8, 0, 0, 0. Esto debe generar un ajuste en la salida DIRECCION, con los datos de entrada toma un valor de 0.829, que permite cambiar el sentido hacia la derecha, figura 4.13.

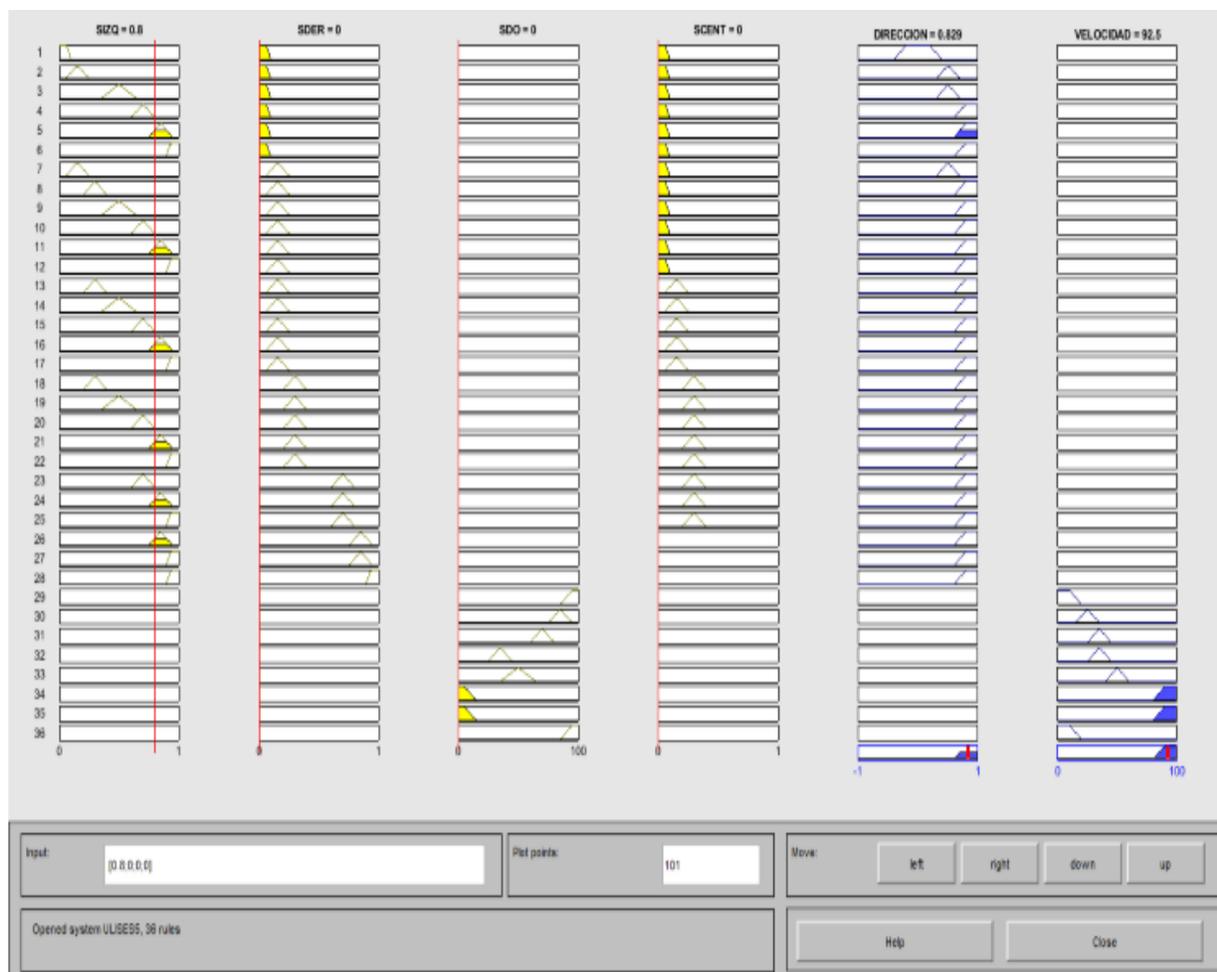


Figura 4.13.- El vehículo tiende a ir a la izquierda saliéndose de la trayectoria.

Tercer caso el vehículo pierde la dirección tendiendo a ir al lado derecho y no existe obstáculos por lo que SIZQ, SDER, SDO y SCENT toman los siguientes valores: 0, 0.8, 0, 0. Esto debe generar un ajuste en la salida DIRECCION, con los datos de entrada toma un valor de -0.828, que permite cambiar el sentido hacia la izquierda, figura 4.14.

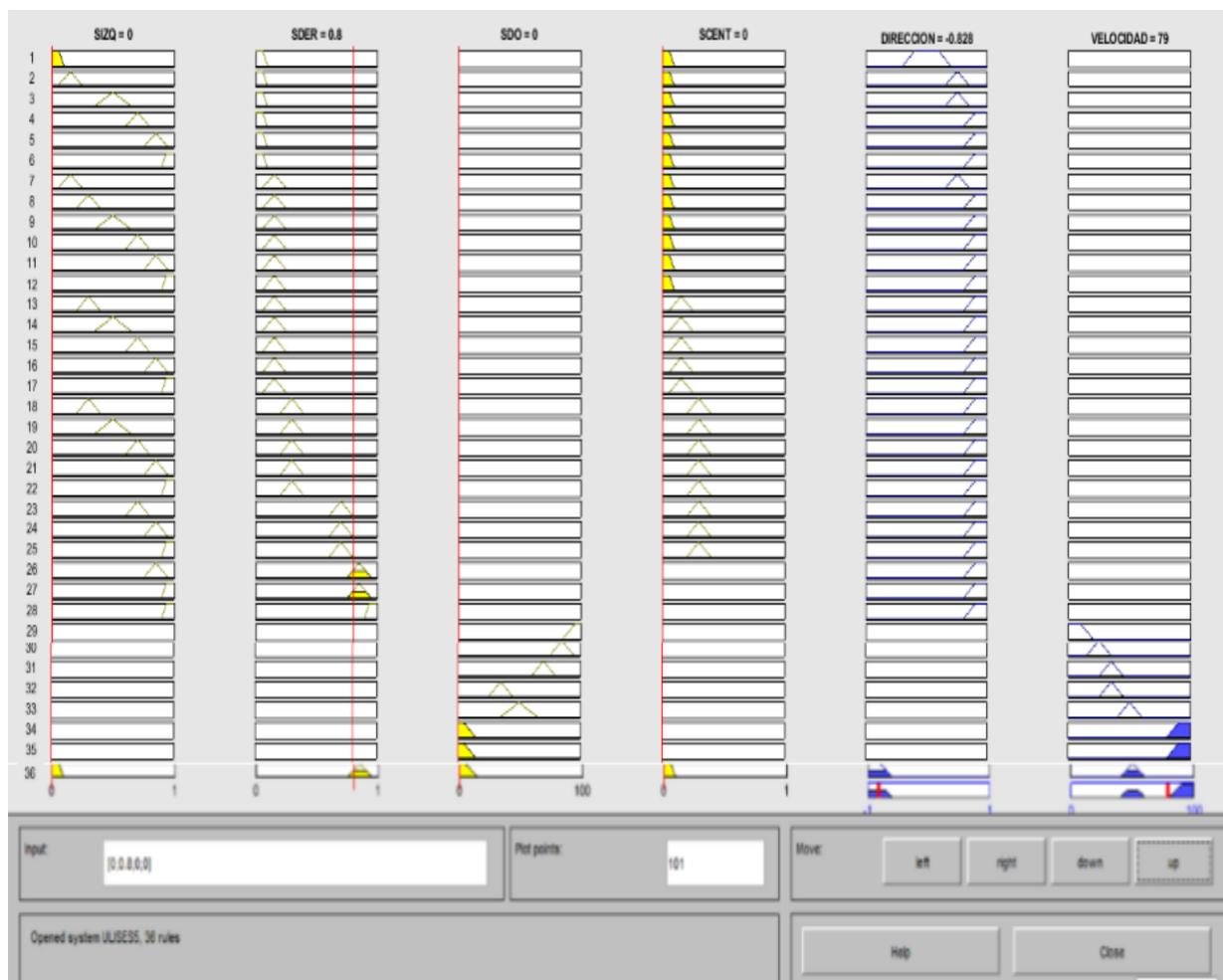


Figura 4.14.- El vehículo tiende a ir a la derecha saliéndose de la trayectoria.

Cuarto caso el vehículo va en línea recta y un obstáculo entra en el rango de detección. Entonces SDO inicia a tomar valores dependiendo de la distancia por lo que la velocidad debe tender a disminuir. Lo cual se aprecia en las siguientes figuras 4.15-4.19:

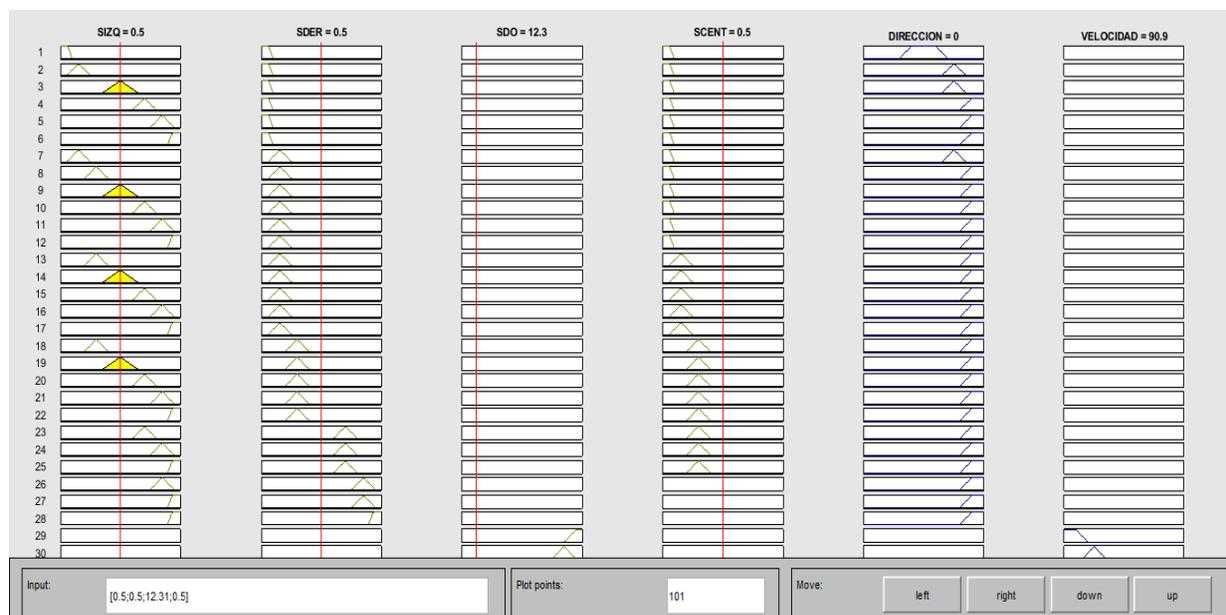


Figura 4.15- Un obstáculo ha sido detectado por SDO.

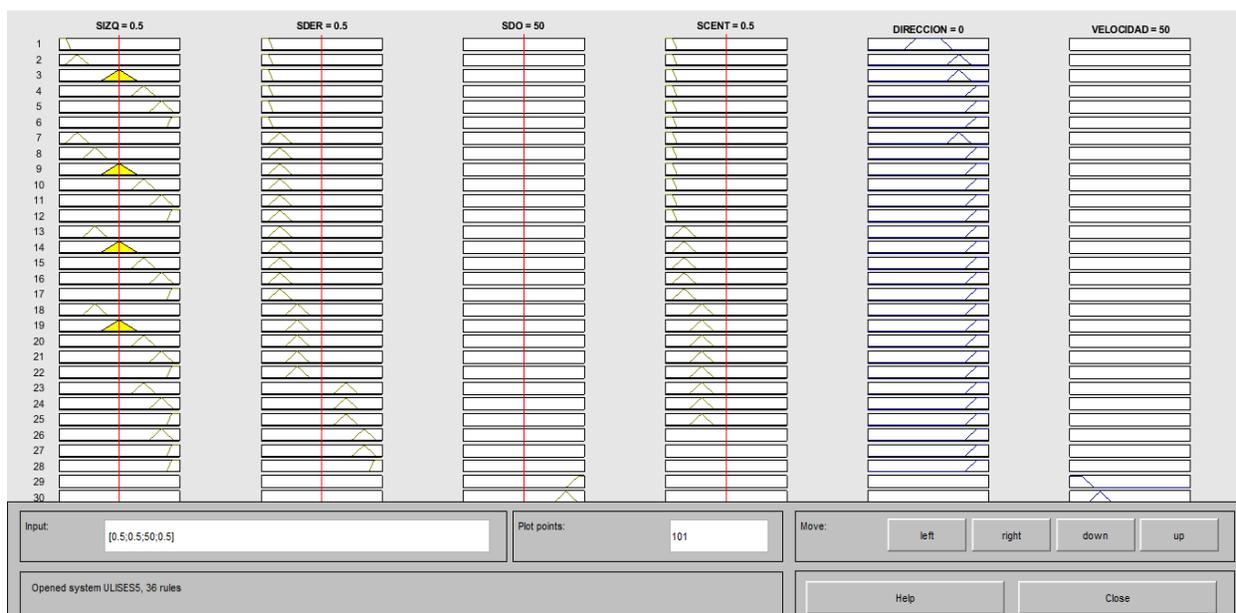


Figura 4.16.- El vehículo se está acercando al obstáculo.

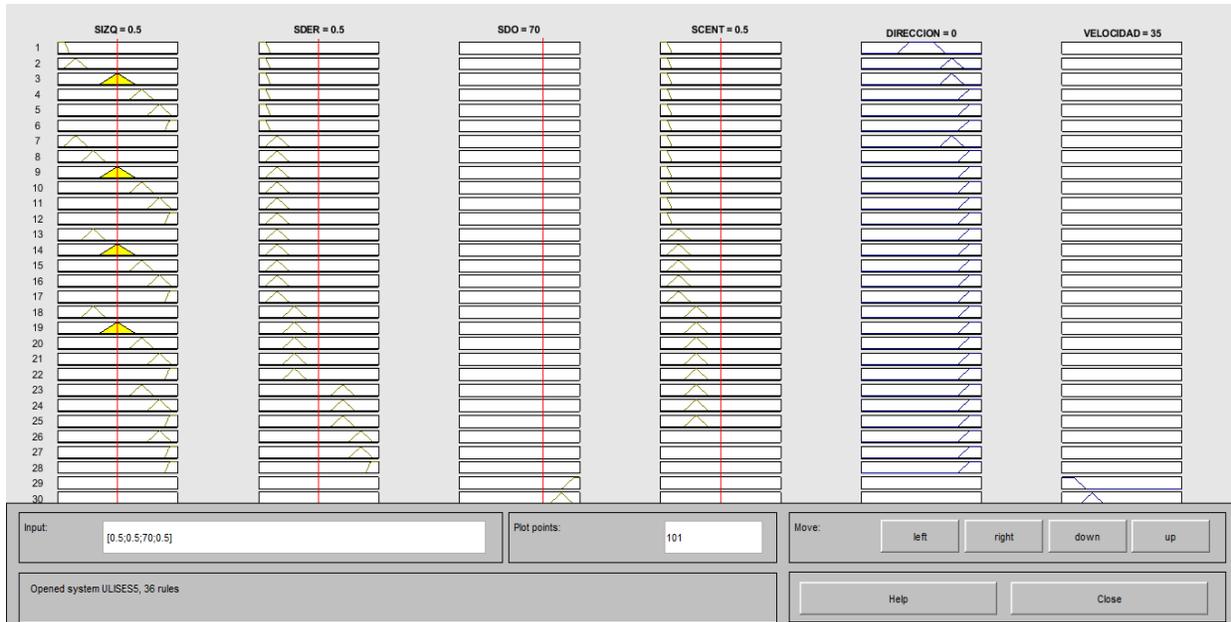


Figura 4.17.- El vehículo está cerca del obstáculo.

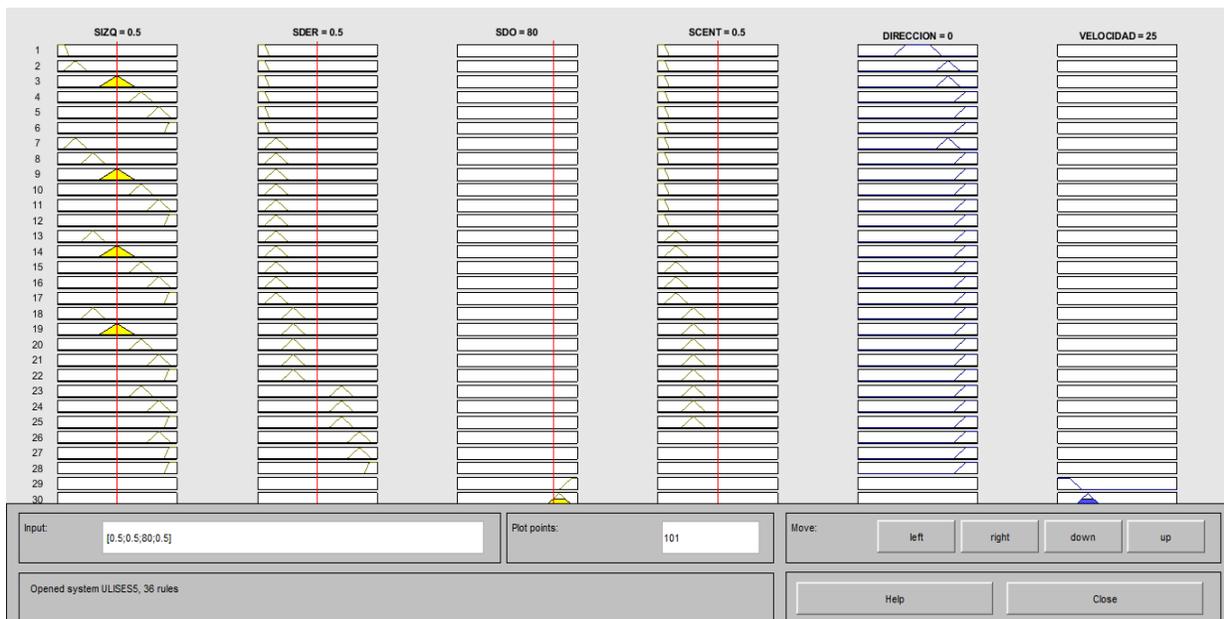


Figura 4.18.- El vehículo está muy cerca del obstáculo.

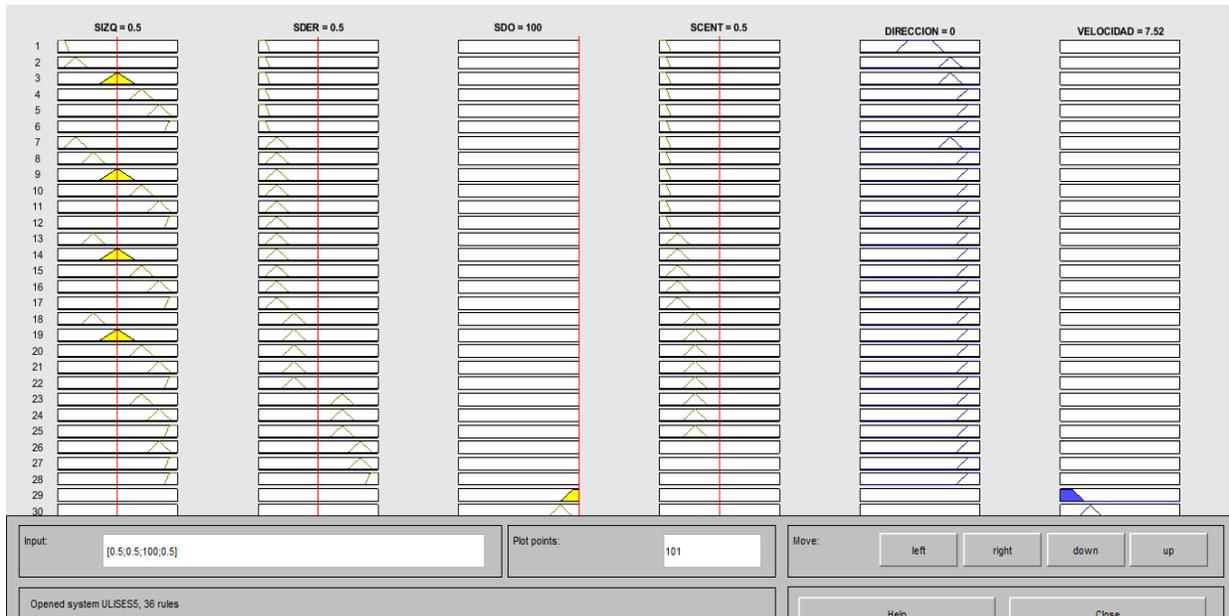


Figura 4.19.- El vehículo está muy próximo del obstáculo.

Todos los anteriores casos simulados son factibles de ocurrir en la realidad y se apreció que las salidas tomaban valores en función con las entradas que son los correctos. Por lo que se concluye que el modelo de inferencia difusa programado en **MATLAB®** utilizando **Fuzzy Logic Toolbox™**, puede ser implementado como sistema de control al vehículo propuesto.

5.- CONCLUSIONES Y RECOMENDACIONES

Del presente proyecto puedo concluir que las estrategias que ofrece la lógica difusa son muy útiles en la resolución de problemas de control, en buena medida porque no es necesario conocer a fondo el modelo matemático. Y puede ser diseñado a partir de las recomendaciones de un experto en forma lingüística. Sin embargo, no deja de ser una alternativa robusta y flexible de control, con una amplia variedad de aplicaciones.

En lo particular puedo decir que el controlador diseñado puede ser implementado sin mayores problemas a un vehículo, puesto que en las pruebas simuladas se obtuvieron resultados satisfactorios.

Otro punto que destacar es que al utilizar herramientas como **MATLAB®** con su software **Fuzzy Logic Toolbox™**, en conjunto con el uso de tarjetas de desarrollo compatibles con Arduino su diseño e implementación se facilitan.

Finalmente, queda por mencionar las conclusiones en lo referente personal:

El proyecto presentó todo un reto ya que mi formación académica es en el área de ingeniería industrial donde este tipo de temas no son vistos más que en vaguedad y el haberlo logrado fue una gran satisfacción al momento. Y fue gracias a que pude aplicar los conocimientos adquiridos a lo largo de la maestría en ingeniería mecatrónica, lo cual representa una superación personal y académica.

6.- BIBLIOGRAFÍA

- CETINA. (2012). *Aprendizaje por refuerzo*. V. U. College, E. T. (2012). PATH PLANNING OF LINE FOLLOWER ROBOT. *IEEE Proceedings of the 5th European DSP Education and Research Conference*, 42-54.
- Gilat, A. (2005). *MATLAB®: An Introduction With Applications*. Hoboken(NJ): Jhon Wyles and Sons, Inc.
- Gilat, A. (2005). *MATLAB®: An Introduction With Applications*. Hoboken(NJ): Jhon Wyles and Sons, Inc.
- Hanss, M. (2004). *Applied Fuzzy Arithmetic: An Introduction With Engineering Applications*. Springer.
- Helmick, D. M. (2004). Path following using visual odometry for a mars rover in high-slip environments. . *En Aerospace conference, 2004. proceedings 2004 ieee Vol. 2*, pp. 772–789).
- J., P. (2009). *Manual de Arduino*. Universidad de Alicante: Grupo de Innovación Educativa en Automática.
- Mamdani, E. H. (1974). *Application of fuzzy algorithms for control of simple dynamic plant*. Academic Press.
- Mamdani, E. H. (1974). *Application of fuzzy algorithms for control of simple dynamic plant*. Academic Press.
- Melin, O. C. (2007). *Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics : Theory and Applications*. Springer.
- Morales, R. R. (2011). *PROCESAMIENTO Y ANALISIS DIGITAL DE IMAGENES*. Ra Ma.
- Pedro, P. (2010). *Inteligencia Artificial con Aplicaciones a la Ingenieria*. Ciudad De Mexico: Alfa Omega Grupo Editor S.A. de C.V., Mexico.
- Ross, T. J. (1995). *Fuzzy Logic with Engineering Applications*. Wiley.
- Tom, M. R. (1955). *Machine Learning*. Morgan Kaufmann.
- Zadeh. (1962). From Circuit Theory to System Theory. *Proc of the IRE*. 50(5), 856-865.
- Zadeh. (1965). Fuzzy Sets. *Information and Control*. 8(3),, 338-353.

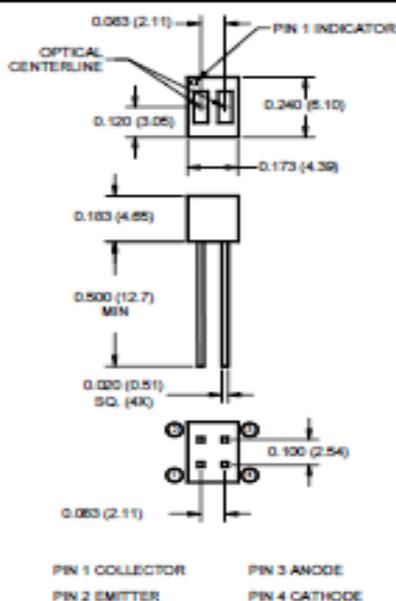
7.- ANEXOS

Hoja de datos QR1114



QRD1113/1114 REFLECTIVE OBJECT SENSOR

PACKAGE DIMENSIONS



NOTES:

1. Dimensions for all drawings are in inches (millimeters).
2. Tolerance of $\pm .010$ (.25) on all non-nominal dimensions unless otherwise specified.
3. Pins 2 and 4 typically $.050^\circ$ shorter than pins 1 and 3.
4. Dimensions controlled at housing surface.

FEATURES

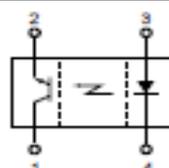
- Phototransistor Output
- No contact surface sensing
- Unfocused for sensing diffused surfaces
- Compact Package
- Daylight filter on sensor



NOTES (Applies to Max Ratings and Characteristics Tables.)

1. Derate power dissipation linearly 1.33 mW/°C above 25°C.
2. RMA flux is recommended.
3. Methanol or isopropyl alcohols are recommended as cleaning agents.
4. Soldering iron $\frac{1}{16}$ " (1.6mm) from housing.
5. As long as leads are not under any spring tension.
6. D is the distance from the sensor face to the reflective surface.
7. Cross talk (I_{CT}) is the collector current measured with the indicator current on the input diode and with no reflective surface.
8. Measured using an Eastman Kodak neutral white test card with 90% diffused reflecting as a reflective surface.

SCHEMATIC



ABSOLUTE MAXIMUM RATINGS ($T_A = 25^\circ\text{C}$ unless otherwise specified)

Parameter	Symbol	Rating	Units
Operating Temperature	T_{OPS}	-40 to +85	$^\circ\text{C}$
Storage Temperature	T_{STG}	-40 to +85	$^\circ\text{C}$
Lead Temperature (Solder Iron) ^(1,2)	T_{SOL-I}	240 for 5 sec	$^\circ\text{C}$
Lead Temperature (Solder Flow) ^(1,2)	T_{SOL-F}	260 for 10 sec	$^\circ\text{C}$
EMITTER			
Continuous Forward Current	I_F	50	mA
Reverse Voltage	V_R	5	V
Power Dissipation ⁽¹⁾	P_D	100	mW
SENSOR			
Collector-Emitter Voltage	V_{CEO}	30	V
Emitter-Collector Voltage	V_{ECO}		V
Power Dissipation ⁽¹⁾	P_D	100	mW



QRD1113/1114 REFLECTIVE OBJECT SENSOR

ELECTRICAL / OPTICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$)						
PARAMETER	TEST CONDITIONS	SYMBOL	MIN	TYP	MAX	UNITS
EMITTER						
Forward Voltage	$I_F = 20\text{ mA}$	V_F	—	—	1.7	V
Reverse Current	$V_R = 5\text{ V}$	I_R	—	—	100	μA
Peak Emission Wavelength	$I_F = 20\text{ mA}$	λ_{PE}	—	940	—	nm
SENSOR						
Collector-Emitter Breakdown	$I_C = 1\text{ mA}$	BV_{CEO}	30	—	—	V
Emitter-Collector Breakdown	$I_E = 0.1\text{ mA}$	BV_{ECO}	5	—	—	V
Dark Current	$V_{CE} = 10\text{ V}, I_C = 0\text{ mA}$	I_D	—	—	100	nA
COUPLED						
QRD1113 Collector Current	$I_F = 20\text{ mA}, V_{CE} = 5\text{ V}$ $D = .050''$ (R/R)	$I_{CP(R)}$	0.300	—	—	mA
QRD1114 Collector Current	$I_F = 20\text{ mA}, V_{CE} = 5\text{ V}$ $D = .050''$ (R/R)	$I_{CP(R)}$	1	—	—	mA
Collector Emitter Saturation Voltage	$I_F = 40\text{ mA}, I_C = 100\text{ }\mu\text{A}$ $D = .050''$ (R/R)	$V_{CE(sat)}$	—	—	0.4	V
Cross Talk	$I_C = 20\text{ mA}, V_{CE} = 5\text{ V}, E_C = 0$ (2)	I_{CX}	—	200	10	μA
Rise Time	$V_{CE} = 5\text{ V}, R_L = 100\text{ }\Omega$	t	—	10	—	μs
Fall Time	$I_{CP(R)} = 5\text{ mA}$	t	—	50	—	μs

Datos del sensor HSCR04

Los sensores ultrasónicos usan sonar para determinar la distancia de un objeto como los murciélagos o delfines lo hacen, ofrecen un excelente rango de detección sin contacto, su operación no es afectada por la luz del sol o materiales oscuros, aunque materiales acústicamente blandos son difíciles de detectar.

Características:

Voltaje de alimentación: +5VDC Corriente en espera: <2mA Corriente de trabajo: 15mA

Ángulo eficaz: <15°

Rangos de distancia: 2cm a 400cm o 1" a 13ft Resolución: 0.3cm

Ángulo de medida: 30°

Ancho de pulso de disparo (Trigger Input Pulse Width): 10us se aplica pulso de (10us en Trig)

Eco (Echo): salida del sensor. Frecuencia de ultrasonido: 40KHz

PINES DE CONEXIÓN

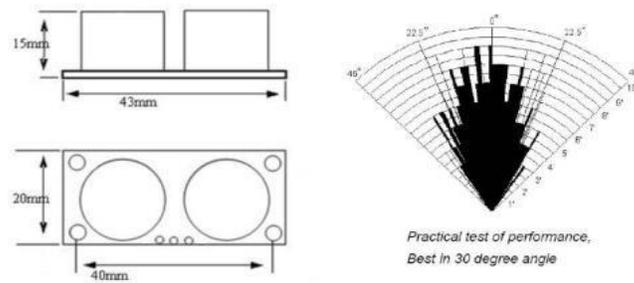


VCC = +5VDC

Trig = Trigger input of Sensor, entrada del pulso de disparo para iniciar la medición.

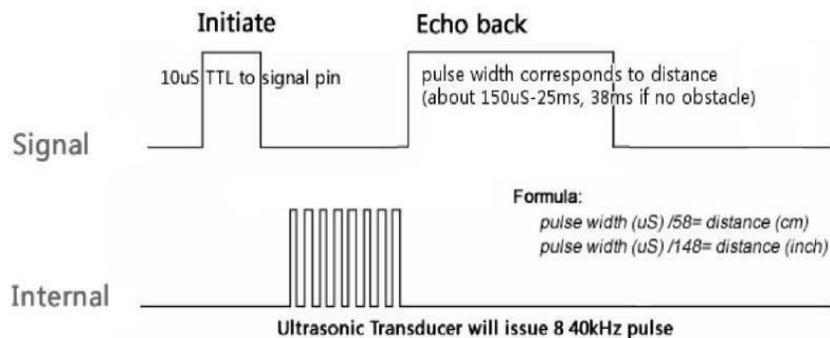
Echo = Echo output of Sensor, pin de eco, que retorna un pulso proporcional a la distancia que rebota el sonido.

GND = GND



Operación

Para iniciar la medición el pin **Trig**, el de la señal de disparo (trigger) debe recibir un pulso de cinco voltios por al menos diez micro segundos, esto indicará al sensor que transmita una ráfaga de ocho ciclos ultrasónicos a 40KHz y espere por la ráfaga reflejada cuando el sensor detecta la señal de ultrasonido en el receptor enviará una señal en alto (5v) por el pin **Echo**, esta señal tendrá un retardo (ancho) proporcional a la distancia, así que para obtener la distancia se debe medir el Ton (tiempo en alto en el pin Echo)



Tiempo que tarda la señal en volver es igual al ancho del pulso en el pin Echo en microsegundos,

Datos del puente H

Product Summary

The **BTS 7960** is a fully integrated high current half bridge for motor drive applications. It is part of the **NovalithiC™** family containing one p-channel highside MOSFET and one n-channel lowside MOSFET with an integrated driver IC in one package. Due to the p-channel highside switch the need for a charge pump is eliminated thus minimizing EMI. Interfacing to a microcontroller is made easy by the integrated driver IC which features logic level inputs, diagnosis with current sense, slew rate adjustment, dead time generation and protection against overtemperature, overvoltage, undervoltage, overcurrent and short circuit.

The **BTS 7960** provides a cost optimized solution for protected high current PWM motor drives with very low board space consumption.

Basic Features

- Path resistance of typ. 16 mΩ @ 25 °C
- Low quiescent current of typ. 7 μA @ 25 °C
- PWM capability of up to 25 kHz combined with active freewheeling
- Switched mode current limitation for reduced power dissipation in overcurrent
- Current limitation level of 43 A typ.
- Status flag diagnosis with current sense capability
- Overtemperature shut down with latch behaviour
- Overvoltage lock out
- Undervoltage shut down
- Driver circuit with logic level inputs
- Adjustable slew rates for optimized EMI

BTS 7960B

P-TO-263-7



BTS 7960P

P-TO-220-7



Type	Ordering Code	Package
BTS 7960B	Q87060-S6160	P-TO-263-7
BTS 7960P	on request	P-TO-220-7

1 Overview

The **BTS 7960** is part of the **NovalithiC™** family containing three separate chips in one package: One p-channel highside MOSFET and one n-channel lowside MOSFET together with a driver IC, forming a fully integrated high current half-bridge. All three chips are mounted on one common leadframe, using the chip on chip and chip by chip technology. The power switches utilize vertical MOS technologies to ensure optimum on state resistance. Due to the p-channel highside switch the need for a charge pump is eliminated thus minimizing EMI. Interfacing to a microcontroller is made easy by the integrated driver IC which features logic level inputs, diagnosis with current sense, slew rate adjustment, dead time generation and protection against overtemperature, overvoltage, undervoltage, overcurrent and short circuit. The BTS 7960 can be combined with other BTS 7960 to form H-bridge and 3-phase drive configurations.

1.1 Block Diagram

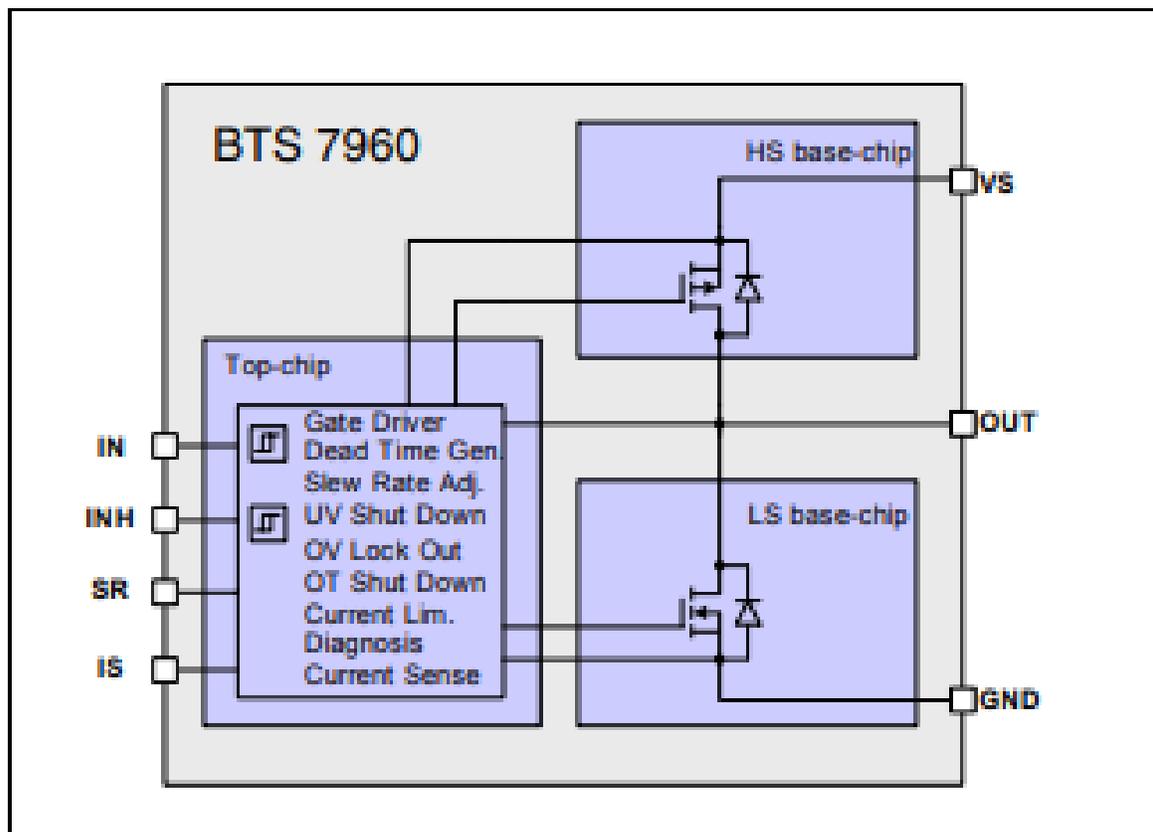


Figure 1 Block Diagram

2 Pin Configuration

2.1 Pin Assignment

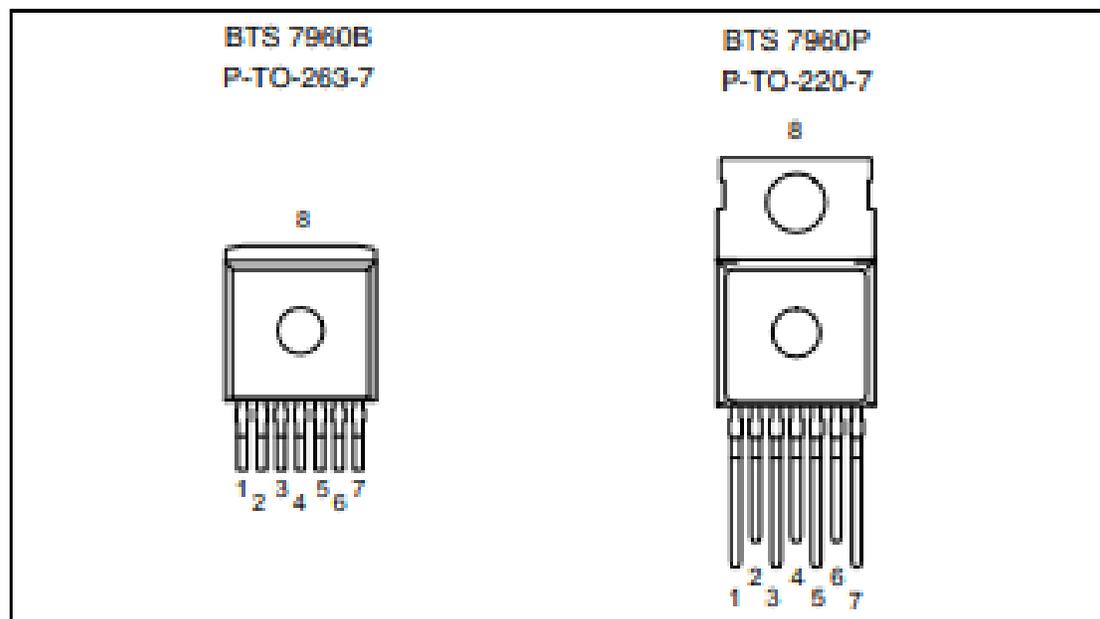


Figure 3 Pin Assignment BTS 7960B and BTS 7960P (top view)

2.2 Pin Definitions and Functions

Pin	Symbol	I/O	Function
1	GND	-	Ground
2	IN	I	Input Defines whether high- or lowside switch is activated
3	INH	I	Inhibit When set to low device goes in sleep mode
4,8	OUT	O	Power output of the bridge
5	SR	I	Slew Rate The slew rate of the power switches can be adjusted by connecting a resistor between SR and GND
6	IS	O	Current Sense and Diagnosis
7	VS	-	Supply

Bold type: pin needs power wiring

Microinterruptor o sensor final de carrera.

TECHNICAL DATASHEET

IMO

Microswitches MS Series

General purpose microswitches for heavier duty applications

- High precision snap action mechanism
- 16A 250V AC resistive rating (I_{res})
- Wide range of actuator styles
- Screw terminals with self lifting plate
- 10 million mechanical operation cycles

- Protection degree IP20, IP40 or IP65
- Versions with positive opening
- Mechanically interchangeable with previous products (see cross reference section)
- UL and cUL Approvals





Options & Ordering Codes

	MS	ST	C1	1	D	A01	G
Terminal Type							
Screw terminals with self-lifting plate	ST						
Contact Block							
1NO+NC, snap action	C1						
Max. Protection Degree							
IP40 (with protection)	1						
IP65 (with protection)	2						
Actuation Type							
Direct action	D						
Back direct action	B						
Inverted action	R						
Contacts Type							
Silver contacts (standard)	G						
Silver contacts gold plated 1 µm	G						
Actuator (see following pages)							
With pin	A01						
With pin	A02						
With small push-button	A03						
	-						

ATTENTION: The feasibility of a code number does not ensure effective availability of a product. There are other options possible. If you can not see the option you require please contact IMO.



www.imopc.com

Microswitches MS Series

Introduction



The MS series of microswitches has been developed with added features to replace the existing MV range. The main features of the new range have been kept the same as the existing MV range to allow for interchangeability.

However, extra features have increased the application field where these switches can be used. The innovative feature of this series is the tripping device which has evolved with the use of modern technology, allowing added features that offer a higher number of solutions when compared with similar devices currently present in the market.

The contacts of the new MS range have a higher reliability factor which has been achieved with the use of double contacts which are also used where positive opening of the contact is required.

The housing has been designed so a gasket can be added as an option in order to seal the device against fine dust or liquids up to IP65.

The terminals are more practical and allow for connection of a wider range of cable diameters. There are also options available with Flut-On terminals, with the choice of three different terminal exit angles.

Contact Block Reliability

In the following table we refer to the typical microswitch contact structure (type A) normally used in the industry, compared with the innovative solution that IMO Precision Controls uses in new MS series microswitches: movable contact with single interruption and double contacts (type B).

As you can see from the table below, this last structure (type B) offers half of the contact resistance (R) than the simple mobile contact (type A) and a lower probability of failure (f).

In fact, defined x the probability of a commutation failure of a single interruption, it results that in the type A the failure probability is =x, in the type B the probability is =x². This means that if in a certain situation the failure probability x is equal, for instance, to 1 x 10⁻⁴ (1 failed interruption every 10.000), we will have:

- in type A one failed commutation every 10.000

- in type B one failed commutation every 100.000.000

Type	Figure	Description	Contact resistance R	Probability of failure f
A Common microswitch		Contacts with single interruption	$R = R_0$	$f = x$
B IMO microswitch MV series		Contacts with single interruption and double contacts	$R = R_0/2$	$f = x^2$

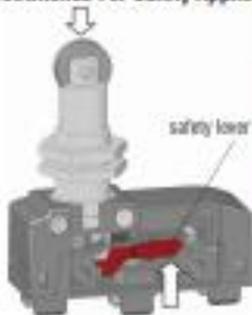


Extended Temperature Range

-40°C

On request the new MS series are also available with an extended temperature range. Where the IMO standard MS microswitches have a temperature range of -25°C to +55°C (in), these special versions can be used in places where the ambient temperature changes from -40°C to +85°C leading to possible installation inside cold stores, sterilizers or other equipment using very low ambient temperatures. Special materials have been used to realize these versions and these allow the specifications and features to remain unchanged under these conditions, thereby widening the installation possibilities. To have this option, add suffix H6 to the part number.

Microswitches For Safety Applications



All microswitches that have beside the part number have a positive opening mechanism therefore suitable for safety applications.

These microswitches are provided with a rigid connection between the actuating plunger and the NC contacts, which means these are opened by force through a strong/hurdy internal safety lever.

The positive opening is in conformity with the IEC 60947-5-1 standard and as such these microswitches are suitable for installation in protection application.

TECHNICAL DATASHEET

IMO

Microswitches MS Series

Protection Degree IP20

By installing microswitches type MSSTC1 box with terminal cover AC-C01 it is possible to obtain a microswitch that is IP20.

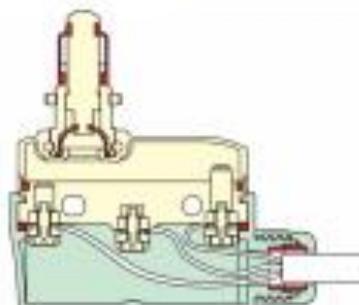
Protection Degree IP40

By installing microswitch types MSSTC1 box with terminal cover AC-C02 it is possible to obtain a microswitch that is IP40.

Protection Degree IP65

By installing microswitch types MSSTC1 box (not stocked) with terminal covers AC-MSCV22 or AC-MSCN23 it is possible to obtain a microswitch that is dustproof and waterproof and hence achieve IP65.

- Switch
- Microswitch: MSN12F2
- Terminal cover: VI-MN2V2

**Clamping Screw Terminal For Different Size Cable**

The clamping mechanism of the MS microswitches has been designed to allow for connection of different diameter cables. The clamping plate is designed in such a way to force the cable towards the screw hence achieving the most robust termination possible for all cable sizes within its specification.

Terminal Covers With Cable Gland Entry

Terminal covers can be supplied that incorporate a trap cable gland to achieve a protection level up to IP65.

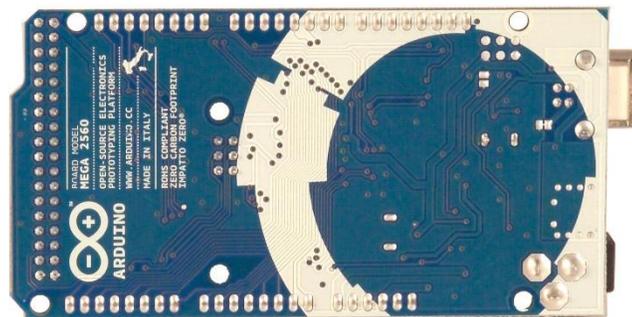
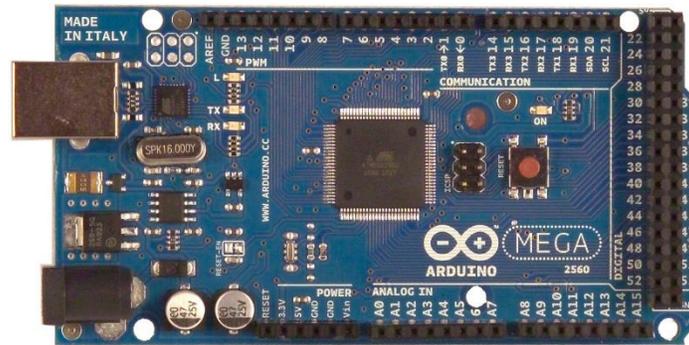
These terminal covers are snap-in assembled and when used increase the size of the microswitch. The use of these covers can also be extended to installations where a number of microswitches are clamped together.

**Rotating Actuators**

The microswitches have been designed to allow the user to rotate the actuator head (roller plunger types only) by 90° steps and this is possible by removing the holding screws, rotating the head and then retightening the screws back.


www.imopc.com

Arduino Mega 2560 Datasheet



Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC- to-DC

adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Schematic & Reference Design

EAGLE files: arduino-mega2560-reference-design.zip

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz