

Fase 1. Robot móvil para recorrer un camino con obstáculos

Carlos Fernando Quintero Olaya (2010006),
carlos.fernando.quintero@correounivalle.edu.co,
Joan Esteban Velasco Larrea (1827539)
joan.velasco@correounivalle.edu.co
Jhonier Andrés Vargas (1745647),
jhonier.vargas@correounivalle.edu.co

Resumen. En el siguiente informe se definen las especificaciones y requerimientos para el diseño de un Robot seguidor de línea que evade obstáculos, en donde dicho camino está definido a partir de una línea de color negro, de tal manera que el robot sea capaz de esquivar obstáculos, saliéndose del camino momentáneamente. Para el diseño de este robot, se ha planteado la metodología RUP, la cual ayuda a conceptualizar y plantear proyectos de ingeniería, a través de un análisis de requerimientos funcionales y no funcionales, y el diseño de diagramas secuenciales, conceptuales y diagramas de flujo.

Palabras clave. Robot Seguidor de Línea, RUP, Atmega328P, HC-SR04, TCRT500, L298N.

I. OBJETIVOS

- Concebir un robot móvil capaz de recorrer un camino y evadir obstáculos en medio de este.
- Usar la metodología RUP para llevar a cabo el desarrollo del software para el robot móvil.

II. INTRODUCCIÓN

Desde los desarrollos en la robótica a lo largo del mundo hasta el enfoque particular de la robótica móvil, es decir, avances en sistemas capaces de operar de manera autónoma con el fin de cumplir un objetivo o tarea se ha buscado siempre la mejor manera de alcanzar resultados óptimos, eficientes, reutilizables y demás con el fin de llevar a cabo tareas que requieren de aspectos que el ser humano se le pueden llegar a dificultar ofrecer, como lo es la precisión, exposición a ambientes que atenten contra su salud, acciones repetitivas y demás tareas que pueden ser satisfechas por medio de robots móviles.

Es por eso que, con el fin de llevar a cabo el desarrollo de un robot móvil, se presenta en este documento toda la información concerniente a la concepción de un robot móvil para seguir un camino hecho por medio de una línea negra, capaz de evadir obstáculos que estarán conforme el avanza o recorre el camino. Esto se busca por medio del uso de un microcontrolador programable en C/C++ y el uso de una metodología de desarrollo de software pues otro objetivo además de concebir, desarrollar e implementar un robot móvil es el de familiarizarse con metodologías de desarrollo de software. Para este proyecto

se empleará la metodología RUP (Rational Unified Process) la cual busca definir todo lo necesario y que se debe tener en cuenta para que la creación del código y en sí de todo el proyecto sea llevado a cabo de una manera más fluida.

III. METODOLOGÍA

La metodología dispuesta para el desarrollo de este proyecto, en especial esta fase 1 que tiene su importancia en la concepción del desarrollo para el proyecto y cómo se buscará implementar el robot móvil es la metodología RUP o Rational Unified Process por sus siglas en inglés o Proceso de Desarrollo Unificado en español, tiene por objetivo brindar un camino o guía a un grupo desarrollador para que el proceso de concepción y creación de un software sea llevado a cabo bajo unos lineamientos y requerimientos específicos que hacen que se pueda dar origen a un software apropiado y que cumpla con los requerimientos específicos, para lograr esto bajo esta metodología es necesario lo siguiente.

A. Análisis de requerimientos

Con esto se busca listar todo lo necesario para que el proyecto cumpla los requerimientos pedidos (capacidades o funcionalidades) para llevar a cabo y que se puedan manejar por medio del software. Estos requerimientos se dividen en requerimientos funcionales y no funcionales.

B. Requerimientos funcionales.

Los requerimientos funcionales son todas las tareas que deben llevar a cabo todas las partes que componen al actor, en este caso el proyecto el cual es el robot móvil. Es necesario ser muy claros y específicos, pues de esta base es que se realizan los casos de uso real otra parte necesaria de la metodología RUP donde se planea las guías que servirán para la producción del código en las siguientes fases. Los requerimientos funcionales para este proyecto se encuentran en el anexo 1.

C. Requerimientos no funcionales.

En este apartado se listan y describen aquellos elementos que forman parte del proyecto, pero no tienen una implicación funcional que interfiera en el desarrollo de la tarea que el robot móvil tenga que cumplir o satisfacer, como también el ambiente de concepción, desarrollo e implementación del software y

hardware. Los requerimientos para este proyecto se encuentran en el anexo 2.

D. Diagrama conceptual.

El diagrama conceptual es una explicación gráfica del funcionamiento del proyecto que se desarrollará, esa explicación es a través de conceptos sobre cosas de la vida real que no son componentes software para que el funcionamiento pueda ser entendido por cualquier persona que no esté relacionada o sepa sobre desarrollo de software o esté involucrada en el ámbito de la programación de microcontroladores y robótica móvil. El diagrama de mapa conceptual del proyecto es mostrado en la figura 1.

E. Casos de uso real.

Los casos de uso real describen un conjunto de acciones que representan situaciones satisfactorias y no satisfactorias de la disposición del software para llevar a cabo una tarea en específico, por cada función necesaria a realizar habrá un caso de uso real. Esto sirve además para tomarse como guía posteriormente en el momento en que se vaya a desarrollar el software. Los casos de uso real se encuentran en el anexo 3.

F. Diagramas de secuencia.

Los diagramas de secuencia son más informativos entre las personas involucradas en el proyecto y personas relacionadas al ámbito del desarrollo de software, un diagrama de secuencia muestra de manera cronológica los mensajes que se dan entre actores, componentes y módulos que componen al proyecto y requieren software para el funcionamiento. Para llevar a cabo la realización de los diagramas de secuencia es necesario tener claros los requerimientos funcionales, por ende, los casos de uso real pues de estos se extraen los componentes o módulos que son requeridos por el software a desarrollar. La distribución de un diagrama de secuencia muestra los componentes a lo largo del eje horizontal o x, y en el eje vertical o y el tiempo o transcurso de los mensajes entre eventos. Los diagramas de secuencia se encuentran en el anexo 4.

G. Diagramas de flujo.

Describe gráficamente las especificaciones e interfaces de la clase de software, siendo mostrado en la figura 5.

IV. MARCO TEÓRICO

Para la concepción de un robot capaz de seguir una línea de color negro, y que el robot sea capaz de evadir obstáculos (abandonando temporalmente la línea del camino claro está) es necesario definir una serie de agente, u objetos, y las interacciones entre estos: Por ejemplo, el camino delimitado por la línea de color negro es uno de los objetos dentro del proyecto, como lo es también el robot mismo, y la interacción entre estos objetos es que el robot recorre el camino. La figura 1 muestra un diagrama conceptual de los objetos e interacciones, de tal

manera que se ilustre la idea de lo que hace el robot para seguir el camino y evadir los obstáculos.

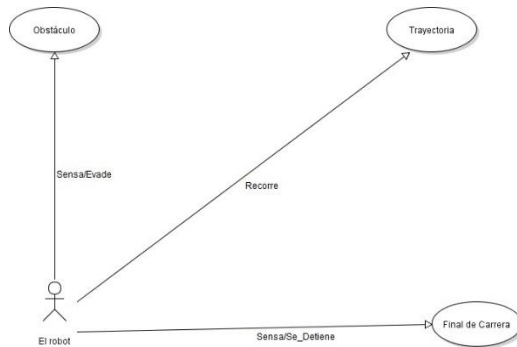


Figura 1. Diagrama Conceptual del Robot Móvil para Recorrer un Camino con Obstáculos.

A continuación, se presenta la estrategia que utilizará el robot para recorrer los caminos y evada los obstáculos.

A. Estrategia para recorrer el Camino

La estrategia desarrollada para que el robot recorra el camino, es necesario usar sensores, que permitan constantemente estar evaluando, si esta sobre la línea negra y sobre la superficie blanca. De esta manera, una estrategia es el uso de luz infrarroja, ya que cuando la luz incide sobre una superficie negra, la luz es absorbida en su totalidad, en cambio, cuando la luz infrarroja es reflejada por una superficie blanca, la luz se refleja casi por completo. Siendo así, al tener en cuenta este fenómeno físico, el sensor ha utilizar, es el sensor infrarrojo, el cual está compuesto por un LED que actúa como transmisor de la luz infrarroja, y el receptor por un fotodiodo, el cual comprueba si la luz ha sido reflejada. Este tipo de sensor arroja un voltaje de salida que es proporcional a la luz reflejada, en donde obtiene un voltaje alto cuando la superficie es clara, y para una superficie oscura es muy baja [1]. Ahora bien, teniendo en cuenta el funcionamiento del sensor se plantea la siguiente estrategia para mantener el robot sobre la línea, usando 2 sensores IR:

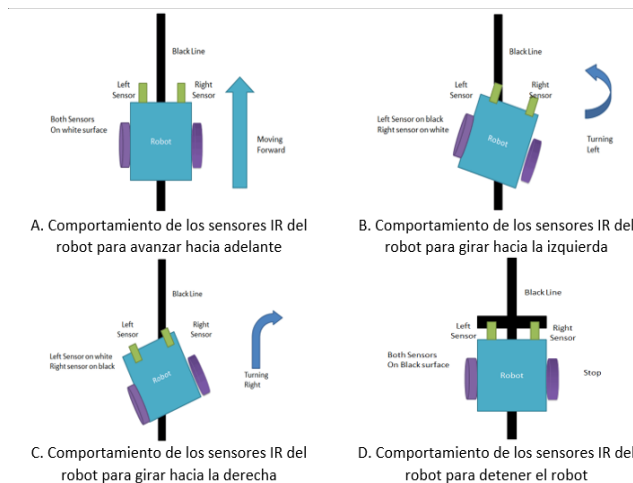


Figura 2. Estrategias para mantener el robot sobre la línea

Como se observa en la Figura 2, se observan 4 estrategias, las cuales son:

- En la Figura 2.A, se observa que cuando el sensor IR de la izquierda y derecha, se encuentran sobre la superficie blanca el robot avanza hacia adelante.
- En la Figura 2.B, se observa que cuando el sensor IR de la izquierda esta sobre la línea oscura, y el sensor de la derecha esta sobre la superficie blanca, el robot deberá girar hacia la izquierda.
- En la Figura 2.C, se observa que cuando el sensor IR de la izquierda esta sobre la superficie blanca, y el sensor de la derecha esta sobre la línea oscura, el robot deberá girar hacia la derecha.
- En la Figura 2.D, se observa que cuando ambos sensores IR, izquierdo y derecho, detectan la línea oscura, indica que el camino ha finalizado, por tanto el robot deberá detenerse.

B. Estrategia para esquivar el obstáculo

La estrategia desarrollada para que el robot esquive los obstáculos que hay en el camino, es necesario usar un sensor, que permitan constantemente estar evaluando, si hay un obstáculo en el camino, indicándole al robot la distancia que tiene sobre el objeto que se le interpone en el camino [1]. De esta manera, una estrategia es el uso de ondas sonoras, las cuales sean reflejadas sobre el objeto, tal y como se observa en la siguiente figura.



Figura 3. Reflexión de onda sonora

Para realizar esto, es necesario de un sensor de ultrasonido, el cual nos ayuda a medir distancias, por medio de un altavoz, el cual es el encargado de enviar el pulso sonoro, y un micrófono, que capta la señal sonora de regreso. De esta manera, el sensor mide el tiempo entre el envío y recepción de la señal sonora [2], usando la siguiente fórmula física:

$$\text{Distancia} = \text{Tiempo} * \text{Velocidad del sonido} \quad (1)$$

Ahora bien, como el sensor nos arroja el tiempo tanto de ida como vuelta, el valor se debe dividir entre 2. Por tanto, la fórmula a usar es:

$$\text{Distancia (Cm)} = \frac{\text{Tiempo}(\mu s) * 0,0343 \frac{\text{cm}}{\mu s}}{2} \quad (2)$$

Teniendo en cuenta, una distancia límite entre el robot y el obstáculo, se plantea que el robot rodee el obstáculo del siguiente modo:

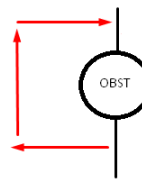


Figura 4. Estrategia para rodear el obstáculo

Como se observa en la Figura 4, para realizar la maniobra de evadir el obstáculo, es seguir una estrategia de seguimiento del contorno, por lo tanto el robot debe de disponer de 2 sensores de ultrasonido(un sensor frontal y un sensor lateral a la derecha del robot), los cuales dependiendo de la situación avanzará o girará a la derecha o izquierda. Cabe destacar, que en el primer momento que el robot detecte un obstáculo, el robot debe girar 90° grados hacia la izquierda. Para luego, verificar las siguientes situaciones:

1. Cuando el sensor Frontal no detecta nada y el sensor lateral detecta el contorno del objeto, el robot debe avanzar hacia adelante.
2. En el momento que el sensor frontal y lateral ya no detecten el contorno del objeto, el robot debe ir avanzando y girando hacia la derecha.
3. Por otro lado, un caso especial es que si el sensor frontal detecta un objeto o una pared y el sensor lateral no detecte nada, el robot debe girar a la derecha.
4. Finalmente, el último caso es que si el sensor lateral y frontal detectan el contorno del obstáculo (Si es irregular), el robot deberá girar hacia la izquierda.

De esta manera, de acuerdo con las estrategias explicadas se ha obtenido el siguiente diagrama de flujo:

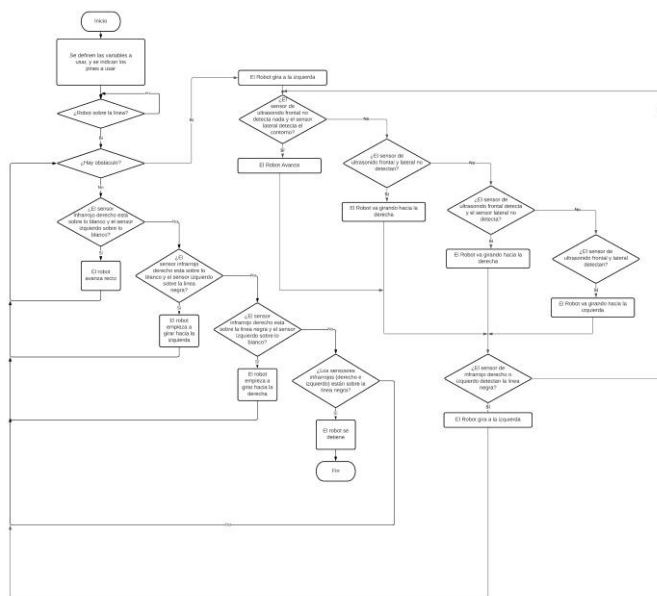


Figura 5. Diagrama de Flujo del funcionamiento del robot

En la Figura 5, se observa que primero se definen las variables y pines que se van a usar a la hora de programar el robot. Luego, antes de avanzar, se debe verificar que el robot se encuentra sobre la línea. Al haber verificado esto, también es necesario comprobar si hay un obstáculo cerca, de tal manera que si lo hay realice las acciones explicadas sobre la Figura 4. En caso de no haber un obstáculo, el robot ya puede avanzar comprobando constantemente lo que detectan los sensores infrarrojos izquierdo y derecho, para así realizar las acciones explicadas de la Figura 2. Cabe destacar, que en el momento de estar realizando la acción de seguimiento de la línea, el robot rápidamente también comprueba si hay un obstáculo, hasta que los sensores infrarrojos detecten si están sobre la línea negra, deteniéndose así en su totalidad.

V. DESCRIPCIÓN DEL HARDWARE

Las características de los elementos a usar son:

Atmega 328p.

El Atmega328p es un chip microcontrolador creado por Atmel y que pertenece a la serie megaAVR. A continuación, se presentan algunas de sus características, y la figura 6 muestra una imagen del microprocesador.[3]

- Memoria Flash: 32 Kbytes
- SRAM: 2 Kbytes
- Cantidad de pines: 28
- Frecuencia Máxima de operación: 20MHZ
- CPU: 852542-bit-AVIR
- Pines Máximos de E/S: 23
- Interrupciones Internas: 24
- Canales ADC: 8
- Resolución de ADC: 10
- Memoria Eeprom: 1Kbytes
- Canales PWM: 6
- Voltaje de Operación: 1.8-5.5v
- Timers: 3



Figura 6. Microprocesador Atmega 328p.

Modulo TCRT5000.

Este módulo cuenta con un diodo de emisión infrarroja TCRT5000 y un potenciómetro ajustable. Útil para equipos de detección de líneas (que es para lo que se utiliza), muestra de datos de impulsos, etc. A continuación, se presentan algunas características del sensor y una imagen de éste. [4]

- Uso de sensores infrarrojos de reflexión TCRT5000
- Prueba de reflexión distancia: 1 mm a 25 mm.
- Salida del comparador libre, forma de onda de la señal, capacidad conducción fuerte mayor a 15 mA.
- Potenciómetro de precisión para ajustar la sensibilidad.
- Voltaje de funcionamiento de 3.3v a 5v.
- Salida: Salida de conmutación digital (0 y 1)
- Agujero de perno fijo para una instalación simple y conveniente.
- Dimensiones: 3.1cm x 1.4cm
- Comparador de tensión amplia LM393
- Detección de línea: blanco y negro.



Figura 7. Sensor óptico Infrarrojo TCRT5000.

Módulo HC-SR04.

El sensor HC-SR04 permite medir distancia mediante ultrasonido, para utilizar en Arduino o PIC, muy utilizado en robots exploradores. En este caso el sensor es utilizado para detectar los obstáculos. A continuación, se presentan algunas características del sensor y una imagen de éste. [5]

- Voltaje de operación: 5Vdc
- Corriente de reposo: 2mA
- Corriente de trabajo: 15mA
- Rango de medición: 2cm a 450cm
- Precisión: +-3cm
- Ángulo de apertura: 15°
- Frecuencia de ultrasonido: 40KHz
- Duración mínima del pulso de disparo TRIG (Nivel TTL): 10us
- Dimensiones: 45mm x 20mm x 15 mm
- Tiempo mínimo de espera entre una medida y el inicio de otra: 20ms (recomendable 50ms)



Figura 8. Sensor Ultrasónico HC-SR04.

Módulo L298N.

Este módulo permite controlar el giro y la velocidad de 2 motores DC de hasta 2A o un Motor paso a paso bipolar. A continuación, se muestra algunas de las características de este módulo y una imagen de éste. [6]

- Chip: L298N
- Canales: 2 (Soporta 2 motores DC o 1 motor PAP)
- Voltaje lógico: 5v
- Voltaje de potencia: 5v – 35v DC
- Consumo de corriente (lógico): 0 a 36mA
- Capacidad de corriente: 2A (Picos de hasta 3A)
- Potencia máxima: 25W
- Dimensiones: 43x43x27mm
- Peso: 30g

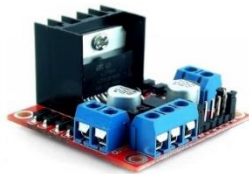


Figura 9. Módulo L298N

Ahora bien, después de haber resaltado las características y especificaciones de cada módulo, se mostrará el diagrama de conexión de los módulos sobre el microcontrolador, tal y como se ve en la siguiente imagen:

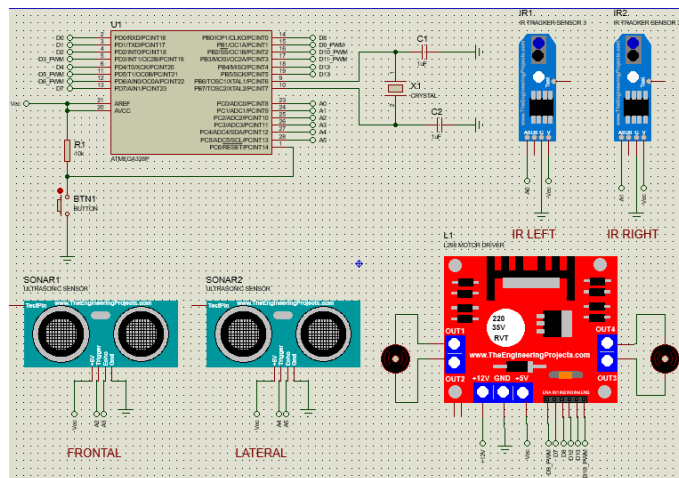


Figura 10. Diagrama de Conexiones

Como se observa en la Figura 10, los módulos se conectan al microcontrolador ATMEGA328P, de la siguiente manera:

- Los sensores infrarrojos Izquierdo y Derecho, es necesario que la entrada A0 del módulo TCRT5000, se conecte al pin analógico A0 para el izquierdo, y para el derecho se debe conectar al pin A1 del microcontrolador.

- El sensor de Ultrasonido Frontal, las entradas Trigger y Echo se conectan al pin analógico A2 y A3 del ATMEGA. De igual manera, para el Ultrasonido Lateral, conectando los entradas Trigger y Echo a los conectan al pin analógico A4 y A5 del ATMEGA
- Finalmente el Motor Drive L298N, las entradas ENA y ENB del módulo se conectan a los pines que puedan generar señales PWM, las cuales son el pin D9 y D10 del microcontrolador, siendo esto útil para controlar las revoluciones de los motores. Por otro lado, las entradas IN1, IN2, IN3, IN4, se deben conectar a los pines digitales del ATMEGA como lo son el D1, D8, D12 Y D13, los cuales sirven para controlar los diferentes modos de operación (Avance, Giro a la derecha, Giro a la izquierda y Detener).

VI. CONCLUSIONES

- El Robot móvil seguidor de línea y evasor de obstáculos, es un sistema autónomo capaz de seguir el camino indicado, siendo a su vez un sistema inteligente ya que es capaz de esquivar obstáculos, sin chocarse. Este Robot es un sistema dotado de sensores como el sensor IR y un sensor de ultrasonido, los cuales usan los principios básicos de la física y propiedades como lo es la reflexión del sonido y la luz. De tal manera, que el sensor IR, es el que ayuda a seguir el camino, y el sensor de ultrasonido es el que evita que se choque con los obstáculos.

- La metodología de desarrollo RUP es una metodología estructurada en la que los desarrolladores deben identificar y planear una estrategia para la resolución de un problema incluso antes de proceso de programación.

- La metodología de desarrollo RUP permite crear algunos diagramas en las que un usuario ajeno al proyecto, y a la ingeniería en general, pueda entender de forma general la solución planteada.

VII. REFERENCIAS

- [1] S. S. Suryawan, S. M. Musamwar, S.R. Kolhe, S. A. Thengane, S. S. Hanumante, P. H. Sahare, "Line Follower & Obstacle Avoider Robot", IRJET, Volume 06, 2019. Disponible en: <https://www.irjet.net/archives/V6/i12/IRJET-V6i1250.pdf>
- [2] UNITELECTRONICS, "Sensor Ultrasonido HC-SR04". Disponible en: <https://uelectronics.com/producto/sensor-ultrasonico-hc-sr04/>
- [3] Wikipedia. Atmega 328. Disponible en: <https://es.wikipedia.org/wiki/Atmega328/>
- [4] Sandorobotics. Módulo Óptico reflexivo Infrarrojo TCRT5000. Disponible en: <https://sandorobotics.com/producto/hr0214-8/>
- [5] Naylamp Mechatronics. Sensor Ultrasónico. Disponible en: <https://naylampmechatronics.com/sensores-proximidad/10-sensor-ultrasonido-hc-sr04.html>
- [6] Naylamp Mechatronics. Driver Puente H L298N 2A. Disponible en: <https://naylampmechatronics.com/drivers/11-driver-puente-h-l298n.html>

ANEXOS

 Universidad del Valle	Universidad del Valle Robot móvil para recorrer un camino con obstáculos	Rev.: 000
Título: ESPECIFICACIÓN DE REQUERIMIENTOS FUNCIONALES		Documento : ERF-001
		Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción del documento	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

Ref #	Funciones	Categoría
1.0	El robot móvil debe ser capaz de detectar los diferentes obstáculos en el camino, usando dos sensores de ultrasonido con referencia HC-SR04	E
2.0	El robot móvil debe ser capaz de detectar la línea negra, usando 2 sensores ópticos infrarrojos TCRT5000	E
3.0	El robot móvil debe ser capaz de regular la velocidad de las ruedas, para controlar el giro por medio del módulo L298N	E
4.0	El robot debe ser capaz de mantenerse dentro de la trayectoria y detenerse una vez llega a un final de carrera señalado en la pista.	E
5.0	El robot debe ser capaz de evadir los obstáculos rodeándolos.	E

Anexo 1. Requerimientos Funcionales del Robot móvil para recorrer un camino con obstáculos

 Universidad del Valle	<div> Universidad del Valle </div> <div> Robot móvil para recorrer un camino con obstáculos </div>		Rev.: 000
Título: ESPECIFICACIÓN DE REQUERIMIENTOS NO FUNCIONALES		Documento : ERF-001	Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción del documento	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

Ref #	Descripción	Categoría
1.0	Para la realización de este proyecto se usará el Atmel Studio 7.0 – IDE ARDUINO	E
2.0	Se usará un microcontrolador Atmega328	E
3.0	Para la realización del esquema físico y conexiones se usará Proteus 8	O

Anexo 2. Requerimientos No Funcionales del Robot móvil para recorrer un camino con obstáculos

 Universidad del Valle	Universidad del Valle Robot móvil para recorrer un camino con obstáculos		Rev.: 000
Título: CASO DE USO Requerimiento funcional 1. El robot móvil debe ser capaz de detectar los diferentes obstáculos en el camino, usando dos sensores de ultrasonido (uno enfrente y otro a un lado) con referencia HC-SR04		Documento : CUR-001	Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción de caso uso	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

INFORMACIÓN GENERAL	
Actores:	Robot (HW) y el firmware (SW)
Propósito:	El robot debe ser capaz de detectar los obstáculos para evitar choques, a través de dos sensores ultrasónicos (uno en frente y otro a un lado) con referencia HC-SR04
Resumen:	Cuando el robot móvil detecte un obstáculo, por medio de los sensores de ultrasonido, se enviará las señales al microcontrolador para que la procese y almacene en una variable de software.
Tipo:	Real

Curso Normal de los Eventos	
Acción del Robot	Respuesta del Firmware
	1. Configurar los puertos de entrada y salida para configurar los sensores de ultrasonido
	2. Configurar los timers para el procesamiento de la señal
3. Los dos sensores ultrasónicos envían señales eléctricas al puerto de entrada y salida	
	4. Se mide el tiempo que permanece activo el ancho de pulso por medio de un timer.
	5. Se calcula la distancia entre el obstáculo y el robot por medio del tiempo medido.
	6. Se compara la distancia calculada con una de referencia para determinar la presencia o no de un obstáculo.
	7. Se almacena la información de la presencia del obstáculo a través de una variable de software.

Anexo 3.1. Caso de uso para el Requerimiento funcional 1

 Universidad del Valle	Universidad del Valle Robot móvil para recorrer un camino con obstáculos		Rev.: 000
Título: CASO DE USO Requerimiento funcional 2. El robot móvil debe ser capaz de detectar la línea negra, usando 2 sensores ópticos infrarrojos TCRT5000		Documento : CUR-001	Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción de caso uso	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

INFORMACIÓN GENERAL	
Actores:	Robot, firmware.
Propósito:	El robot móvil debe ser capaz de detectar la línea negra, usando 2 sensores ópticos infrarrojos TCRT5000
Resumen:	Lo que se busca es que el robot móvil sea capaz de reconocer la trayectoria a seguir, la cual es una línea negra por medio de 2 sensores ópticos infrarrojos.
Tipo:	Real

Curso Normal de los Eventos	
Acción del robot	Respuesta del firmware
	1. Configurar los puertos de entrada y salida para configurar los dos sensores ópticos infrarrojos.
2. Los dos receptores infrarrojos envían una señal eléctrica al puerto de entrada y salida.	
	3. Se almacena el estado binario de las dos señales recibidas desde los receptores infrarrojos.
	4. Se notifica el evento (línea negra detectada o no) a través de una variable en el software.

Anexo 3.2. Caso de uso para el Requerimiento funcional 2

 Universidad del Valle	Universidad del Valle Robot móvil para recorrer un camino con obstáculos		Rev.: 000
Título: CASO DE USO Requerimiento funcional 3 El robot móvil debe ser capaz de regular la velocidad de las ruedas, para controlar el giro por medio del módulo L298N		Documento : CUR-001	Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción de caso uso	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

INFORMACIÓN GENERAL	
Actores:	Robot, firmware.
Propósito:	El robot debe regular la velocidad de cada una de las ruedas para controlar la velocidad del robot y su giro.
Resumen:	Se espera que a partir de los cambios que puedan existir en la trayectoria marcada y los obstáculos el robot cambie la velocidad en cada rueda para girar.
Tipo:	Real

Curso Normal de los Eventos
Respuesta del firmware
1. Se lee el valor de la variable de software (Dirección: Derecha, Izquierda, Adelante, Parar) dado por las funciones que procesan el estado del robot (caso de uso real 4 y 5).

Si el valor de la variable dirección es Parar, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Configurar los puertos de entrada y salida para enviar la señal de control a los motores.
2. Se envía un cero en cada una de las entradas del módulo L298N para que los motores se detengan.

Si el valor de la variable dirección es Avanzar, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Configurar los puertos de entrada y salida para enviar la señal de control a los motores.
2. Se envía dos unos y dos ceros (en el orden correspondiente) en las entradas del módulo L298N para que los motores giren ambos en sentidos contrarios, para que así el robot avance.

Si el valor de la variable dirección es Derecha, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Configurar los puertos de entrada y salida para enviar la señal de control a los motores.
2. Se envía dos unos y dos ceros (en el orden correspondiente) en las entradas del módulo L298N para que los motores giren ambos en el mismo sentido, para que así el robot gire hacia la derecha.

Si el valor de la variable dirección es izquierda, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Configurar los puertos de entrada y salida para enviar la señal de control a los motores.
2. Se envía dos unos y dos ceros (en el orden correspondiente) en las entradas del módulo L298N para que los motores giren ambos en el mismo sentido, para que así el robot gire a la izquierda.

Anexo 3.3. Caso de uso para el Requerimiento funcional 3

 Universidad del Valle	<div> <div>Universidad del Valle</div> <div>Robot móvil para recorrer un camino con obstáculos</div> </div>	Rev.: 000
Título: CASO DE USO Requerimiento funcional 4. El robot debe ser capaz de mantenerse dentro de la trayectoria y detenerse una vez llega a un final de carrera señalado en la pista.		Documento : CUR-001 Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción de caso uso	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

INFORMACIÓN GENERAL	
Actores:	firmware.
Propósito:	El robot debe ser capaz de seguir la línea negra detectada y avanzar
Resumen:	Lo que se busca es que el robot móvil sea capaz de reconocer la trayectoria a seguir, la cual es una línea negra por medio de 2 sensores ópticos infrarrojos.
Tipo:	Real

Curso Normal de los Eventos
Respuesta del firmware
1. Se lee el valor de las variables de software que almacenan el estado de los sensores izquierdo y derecho: 00, 01, 10, 11.

Si el valor de la variable es 00, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Parar"

Si el valor de la variable es 01, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Izquierda"

Si el valor de la variable es 10, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Derecha"

Si el valor de la variable es 11, entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Avanzar"

 Universidad del Valle	Universidad del Valle Robot móvil para recorrer un camino con obstáculos	Rev.: 000
Título: CASO DE USO Requerimiento funcional 5 El robot móvil debe ser capaz de evadir los obstáculos rodeandolos		Documento : CUR-001 Página : 1 de 1

REVISIÓN HISTÓRICA			
Rev.	Descripción del Cambio	Autor	Fecha
001	Construcción de caso uso	Carlos Fernando Quintero, Jhonier Andrés Vargas, Joan Velasco	2021-11-21

INFORMACIÓN GENERAL	
Actores:	firmware.
Propósito:	El robot debe girar y avanzar para rodear un obstáculo que detecte
Resumen:	Se espera que a partir de los cambios que puedan existir en la trayectoria marcada y los obstáculos el robot cambie la velocidad en cada rueda para girar.
Tipo:	Real

Curso Normal de los Eventos
Respuesta del firmware
1. Se lee el valor de las variables de software que almacén la presencia del obstáculo (Enfrente, A_un_lado, Enfrente_y_A_un_Lado, Ninguno).
2. Se define e inicializa una variable Estado_robot en "0" (dentro de la línea).

Nota: Los conceptos "Enfrente", "A un lado", "Enfrente y a un lado" y "Ninguno" se refieren:

- Enfrente: Significa que hay presencia de un objeto detectado por el sensor frontal del robot.
- A un lado: Significa que hay presencia de un objeto detectado por solo el sensor lateral del robot.
- Enfrente y a un lado: Significa que hay presencia de un objeto detectado por el sensor frontal y lateral del robot.
- Ninguno: Significa que el sensor frontal y lateral no detectan ningún objeto.

Para cuando Estado_robot= 0

Si la variable de presencia de obstáculo es "Enfrente", entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Izquierda"

Si la variable de presencia de obstáculo es "A_un_lado", entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable Estado_robot a "1"

Para cuando Estado_robot=1

Si la variable de presencia de obstáculo es "Enfrente", entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "derecha"

Si la variable de presencia de obstáculo es "A_un_lado", entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Avanzar"

Si la variable de presencia de obstáculo es "Enfrente_y_A_un_lado", entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Izquierda"

la variable de presencia de obstáculo es "Ninguno" y la variable estado_robot es 1, entonces:

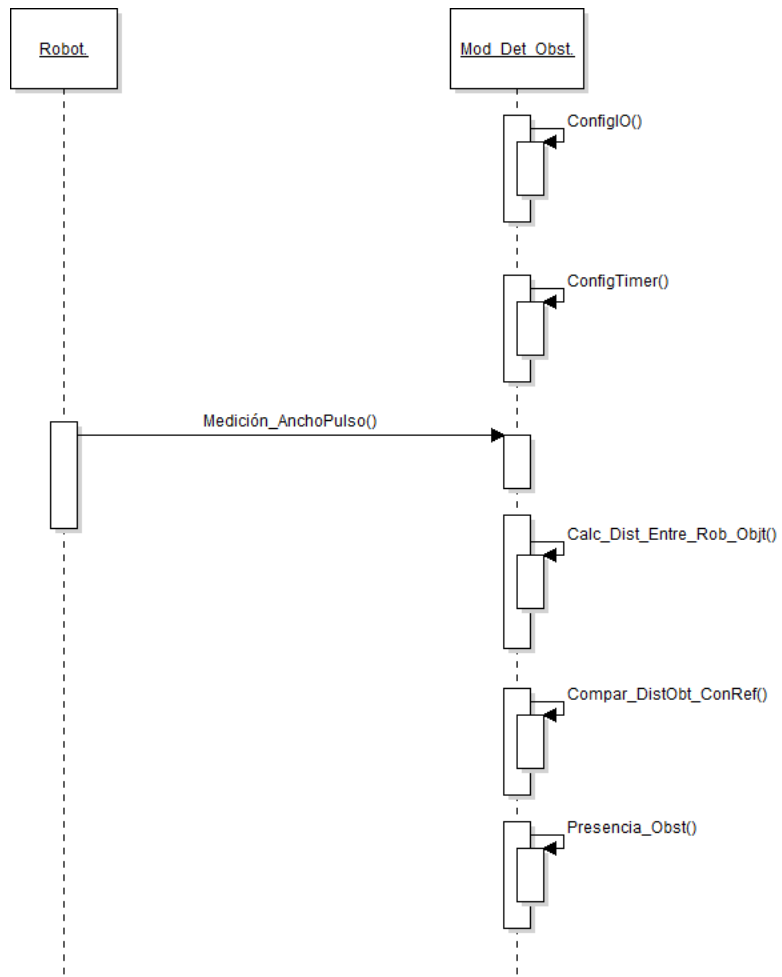
Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "derecha"

Curso Normal de los Eventos
Respuesta del firmware
1. Se lee el valor de la variable de software que almacena el estado de los sensores infrarrojos.

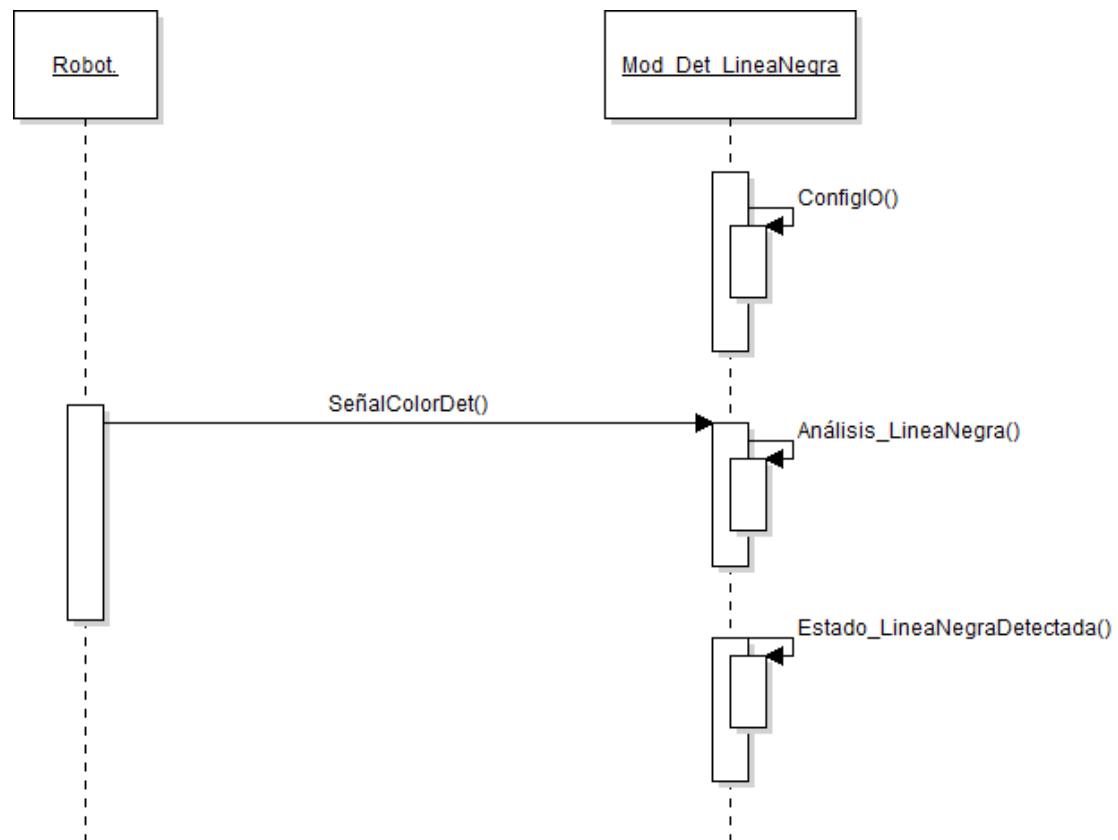
Si la variable de estado de los sensores es "00", "01" o "10", entonces:

Curso Normal de los Eventos
Respuesta del firmware
1. Se cambia el valor de la variable dirección a "Izquierda"
2. Se cambia el valor de la variable estado_robot a "0"

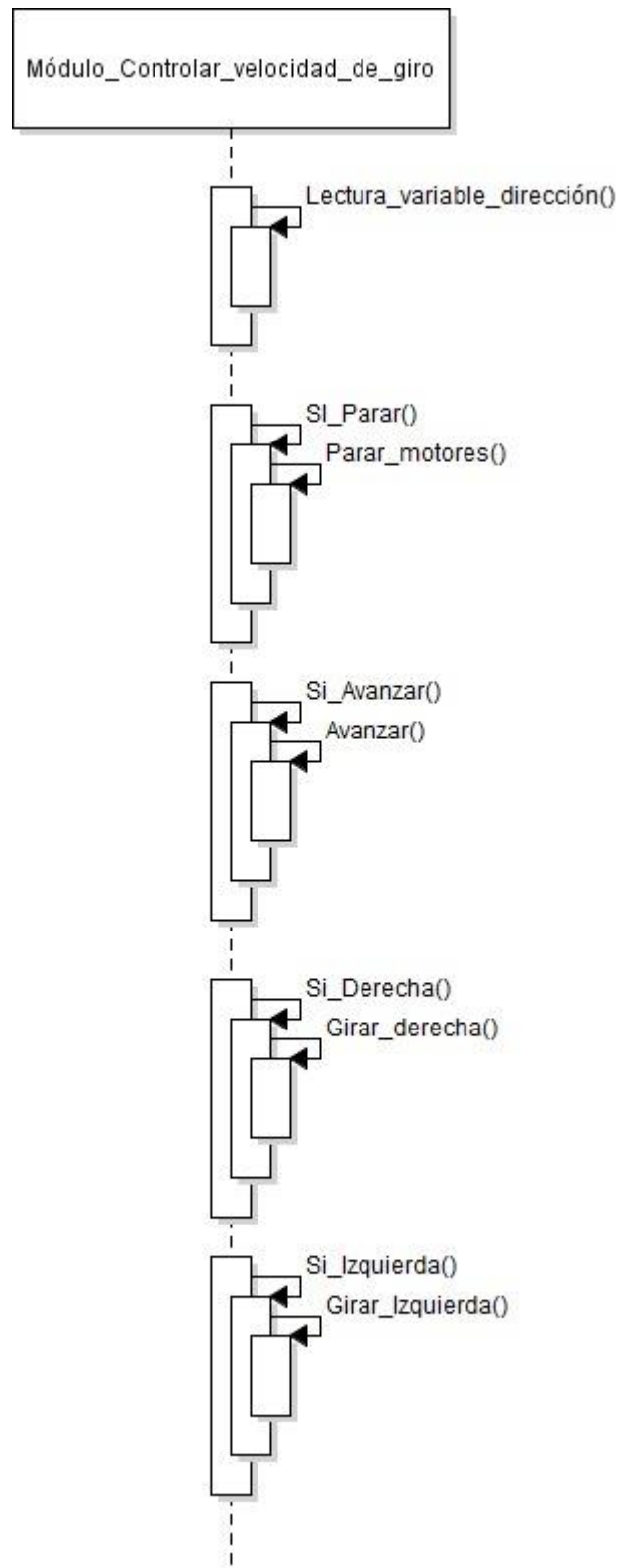
Anexo 3.5. Caso de uso para el Requerimiento funcional 5



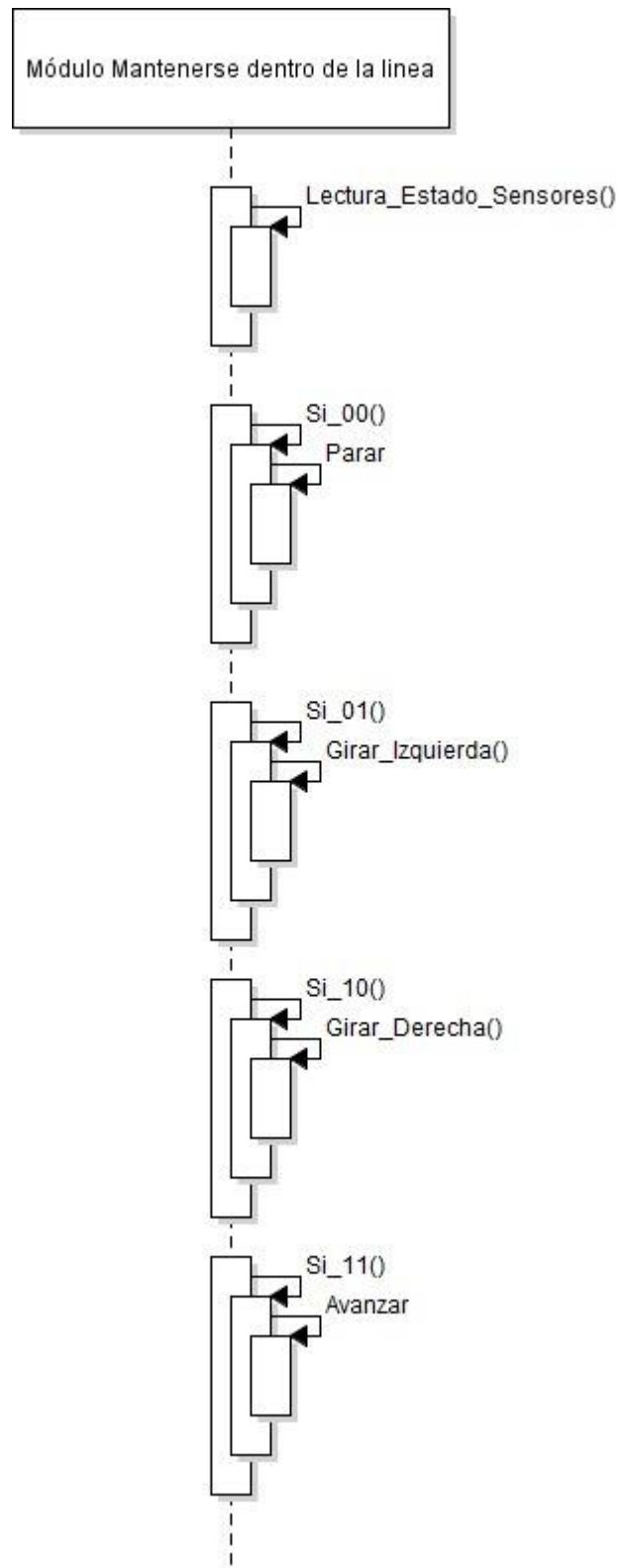
Anexo 4.1. Diagrama secuencial para el Requerimiento funcional 1



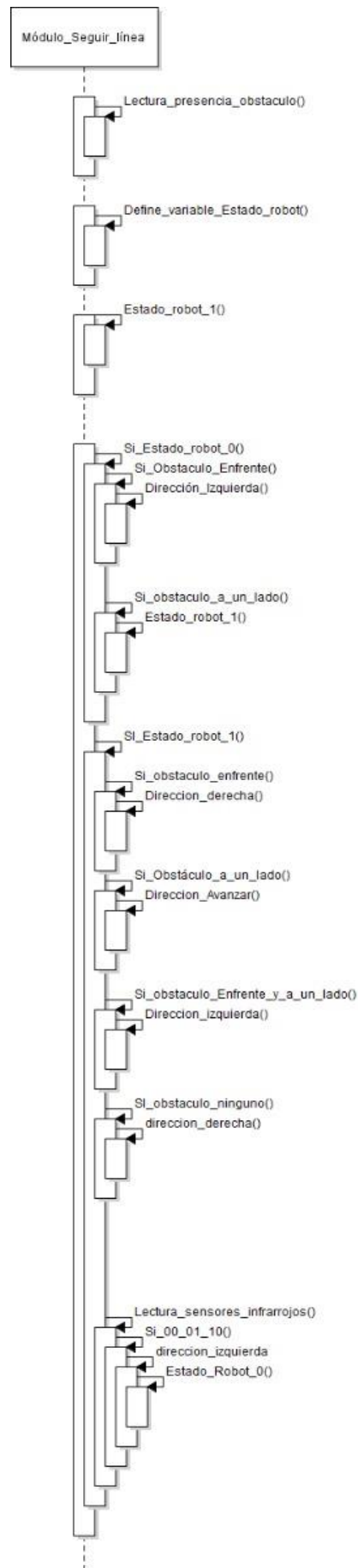
Anexo 4.2. Diagrama secuencial para el Requerimiento funcional 2



Anexo 4.3. Diagrama secuencial para el Requerimiento funcional 3



Anexo 4.4. Diagrama secuencial para el Requerimiento funcional 4



Anexo 4.5. Diagrama secuencial para el Requerimiento funcional 5