



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA

MAESTRÍA EN CIENCIAS DE LA ELECTRÓNICA
OPCIÓN EN AUTOMATIZACIÓN

TESIS PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS DE LA ELECTRÓNICA
OPCIÓN EN AUTOMATIZACIÓN

**ALGORITMO REACTIVO APLICADO EN
LA EVITACIÓN DE OBSTÁCULOS**

Presenta:

Ing. Hermes Javier Ramírez García**

Directora:

Dra. Amparo Dora Palomino Merino

Co Directora:

Dra. Ma. Aurora Diozcora Vargas Treviño

Puebla, Pue., 5 Enero 2018

*Trabajo financiado por VIEP-BUAP

**Becario CONACYT

Agradecimientos

A mi asesora la Dra. Amparo Dora Palomino Merino por el apoyo, orientación y la confianza que me brindo durante este trabajo y al inicio del mismo para poder llevarlo a cabo.

A mi co asesora la Dra. Ma. Aurora Diozcora Vargas Treviño por su apoyo, guía y confianza que me brindo durante el desarrollo de este trabajo.

A los miembros del jurado revisor Dr. Jose Fernando Reyes Cortes, Dr. Sergio Vergara Limón y al Dr. David Eduardo Pinto Avendaño por sus valiosas correcciones, sugerencias y paciencia para mejorar mi desempeño en el proyecto.

Al Consejo Nacional de Ciencia y Tecnología por el apoyo económico brindado y por darme la oportunidad de superarme personal y profesionalmente.

A la Facultad de Ciencias de la Electrónica de la BUAP por brindarme su apoyo para terminar en tiempo y forma este proyecto.

Dedicatoria

Quiero dedicar este trabajo a mi abuela Lilia Lara Amador por ayudarme a comprender por qué necesitaba de los estudios y a comprender mejor la vida, que en paz descanse.

A mi madre y padre que siempre estuvieron encima de mí para no abandonar los estudios e impulsarme a seguir adelante en la vida.

A mi tía Teresa que me brindó su apoyo y cariño durante mis estudios.

A mi hermana que siempre me apoyo en las buenas y las malas como familia y amiga.

A mis sobrinas por apoyarme y brindarme su felicidad y alegría.

A Gabino, Raúl, Rigoberto y demás compañeros de generación y amigos de la maestría que me con sus puntos de vista, comentarios, consejos y compañía me ayudaron a mejorar cada etapa del proyecto y el conocimiento sobre el mismo.

Resumen

En este documento se describe el trabajo realizado en la tesis Algoritmo Reactivo Aplicado en la Evitación de Obstáculos desde el estudio y comprensión de los conceptos teóricos hasta la implementación de los mismos en una plataforma robot móvil para evadir obstáculos dentro de un ambiente controlado. El robot móvil es uno tipo ackerman, cuyo robot ya se tiene en el laboratorio, sin embargo, para fines de control, es necesario hacer modificaciones y adecuaciones a los elementos sensores y actuadores.

Inicialmente se abordan los conceptos teóricos de base que se utilizan para entender el sistema y poder hallar los modelos cinemático y dinámico, conceptualizando las restricciones que, por su propia naturaleza, el sistema físico posee. Posteriormente se propone abordar el control reactivo mediante la estrategia de campos potenciales. Con esta propuesta se realizan simulaciones dentro del ambiente de programación del programa MATLAB obteniendo los resultados que muestran que la estrategia propuesta funciona. Bajo la idea de probar de forma experimental esta, se adecúa la plataforma experimental a las necesidades actuales. Sin embargo, dado que poner a punto la plataforma experimental involucra demasiadas cuestiones de mecánicas, de software y de instrumentación, por falta de tiempo, no se presentan los resultados experimentales, pero sí, la descripción completa de dicha plataforma.

Dentro de los conceptos teóricos se analiza y estudia los modelos matemáticos del robot móvil tipo ackerman correspondientes a su cinemática y dinámica para entender cómo se comportaría el robot móvil cuando se le aplique el control reactivo propuesto, el cual también es analizado su modelo matemático para comprender como es que funciona el mismo y así poder aplicarlo de forma eficiente.

Índice

Índice	v
Índice de Figuras	vii
Índice de tablas	x
Introducción	1
Planteamiento del problema	1
Justificación	1
Propuesta de solución de tesis	2
Objetivos	2
Objetivo General:	2
Objetivos específicos:	2
Capítulo 1	3
Antecedentes	3
1.1 Historia	3
1.2 Estado del Arte	4
Capítulo 2	10
Robots móviles y Modelado	10
2.1 Características de Robot Móviles	10
2.1.1 Locomoción	10
2.1.2 Percepción	11
2.1.3 Razonamiento	12
2.1.4 Comunicación	12
2.1.5 Sistema Holonómico	13
2.2 Modelado	14
2.2 Modelo cinemático	15
2.2 Modelo dinámico	18
2.3 Control punto a punto	21
Capítulo 3	21
Métodos de control para la evitación de obstáculos	22

3.1 Método de campos potenciales artificiales	23
3.1.1 Estructura general	24
3.1.2 Campo potencial de atracción	25
3.1.3 Campo potencial de repulsión	25
Capítulo 4	26
Plataforma experimental del Robot móvil	26
4.1 Estructura mecánica	27
4.2 Actuadores	30
4.3 Sensores	31
4.4 Comunicación	33
Capítulo 5	35
Implementación del control	35
5.1 Módulos Xbee-Pro	35
5.2 Entorno de programación entre Labview-Arduino-Xbee	38
5.3 Procesamiento digital de las señales emitidas por los sensores	44
5.4 Procesamiento de señales digitales de control de motores	48
5.4 Descripción general de la implementación de control	49
Capítulo 6	50
Simulaciones	50
Pruebas experimentales	56
Conclusiones	63
Apéndices	64
Apéndice A Manual HUMMER H2	64
Apéndice B Componentes para instrumentar el robot móvil	66
Apéndice C Publicaciones	90
Bibliografía	99

Índice de Figuras

Figura I. Diagrama a bloques del sistema.	2
Figura 1.1 Modelo Eléctrico.	4
Figura 1.2 Ángulo de detección de LIDAR.	5
Figura 1.3 Trayectoria con obstáculos.	6
Figura 1.4 Orientación cartesiana considerada.	7
Figura 1.5 Análisis de rango y respuesta.	8
Figura 1.6. Comparación entre las trayectorias planteadas.	9
Figura 2.1 Clasificación de robots móviles por locomoción.	10
Figura 2.2 Clasificación de robots con ruedas.	11
Figura 2.3 Interfaz de comunicación.	12
Figura 2.4 Sistemas Holonómicos y no Holonómicos.	13
Figura 2.5 Robot móvil tipo Ackerman.	15
Figura 2.6 Simplificación del modelo del robot móvil.	15
Figura 2.7 Coordenadas generales del robot móvil.	16
Figura 2.8 Coordenadas del robot móvil.	17
Figura 2.9 Diagrama de espacio de trabajo para un robot móvil tipo Ackerman.	19
Figura 3.1 Campo potencial artificial para evitación de obstáculos.	23
Figura 4.1 Robot móvil HUMMER H2.	26
Figura 4.2 Carcasa del robot móvil HUMMER H2.	27
Figura 4.3 Muelle delantero del robot móvil HUMMER H2.	27
Figura 4.4 Muelle trasero del robot móvil HUMMER H2.	28
Figura 4.5 Porta batería del robot móvil HUMMER H2.	28
Figura 4.6 Batería de alimentación del robot móvil HUMMER H2.	29
Figura 4.7 Llantas tipo estándar del robot móvil HUMMER H2.	29
Figura 4.8 Servomotor de dirección del robot móvil HUMMER H2.	30
Figura 4.9 Motor de tracción del robot móvil HUMMER H2.	30
Figura 4.10 Encoder utilizado como sensor de la posición del robot móvil HUMMER H2.	31
Figura 4.11 Potenciómetro utilizado como sensor de la dirección del robot móvil HUMMER H2.	31

Figura 4.12 Sensor infrarrojo SHARP de distancia para medir la misma entre el robot móvil y el obstáculo.	32
Figura 4.13 Sensor IMU-6050 para obtener la posición del robot móvil.	32
Figura 4.14 Tarjeta de interfaz USB (XBIB-U-DEV).	33
Figura 4.15 Tarjeta Arduino UNO.	34
Figura 4.16 Tarjeta SHIELD.	34
Figura 5.1 Conexiones mínimas requeridas para el módulo XBEE-PRO.	35
Figura 5.2 Sistema de flujo de datos mediante microcontroladores.	36
Figura 5.3 Sistema de comunicación inalámbrica.	37
Figura 5.4 Entorno de programación Labview 2011.	38
Figura 5.5 Programa VISA-NIMAX versión 5.5.	39
Figura 5.6 JKI VI Package Manager (VIPM) 2014.	39
Figura 5.7 Descarga del toolkit de la interfaz entre Labview y Arduino.	40
Figura 5.8 Arduino I.D.E. 1.0.4.	40
Figura 5.9 Software X-CTU.	41
Figura 5.10 Archivo LIFA_Base.	42
Figura 5.11. Configuración de parámetros a través del controlador VISA NI MAX.	42
Figura 5.12. Configuración de parámetros a través del administrador de dispositivos.	43
Figura 5.13 Parámetros de inicialización de Arduino en Labview.	43
Figura 5.14 Diagrama de flujo del encoder.	44
Figura 5.15 Programa para la lectura del Encoder en LABVIEW.	44
Figura 5.16 Diagrama de flujo del potenciómetro.	45
Figura 5.17. Programa para la lectura del Potenciómetro en LABVIEW.	45
Figura 5.18 Diagrama de flujo del sensor infrarrojo SHARP.	46
Figura 5.19 Programa para lectura del sensor infrarrojo.	46
Figura 5.20 Gráfica de respuesta del sensor infrarrojo SHARP.	46
Figura 5.21 Diagrama de flujo de la lectura del sensor IMU-6050.	47
Figura 5.22 Lectura de los datos crudos del sensor IMU-6050.	47
Figura 5.23 Interfaz de usuario de la adquisición de datos del sensor IMU-6050.	47
Figura 5.24 Diagrama de flujo del concepto general del control propuesto.	49
Figura 6.1 Diagrama de flujo del código de la simulación	51
Figura 6.2 Gráfica de evasión de un obstáculo.	50

Figura 6.3 Gráfica de evasión de un obstáculo en coordenadas (-x,-y).	52
Figura 6.4 Gráfica de evasión de un obstáculo en coordenadas (-x, y).	53
Figura 6.5 Gráfica de evasión de un obstáculo en coordenadas (x, -y).	53
Figura 6.6 Gráfica de trayectoria original.	54
Figura 6.7 Gráfica de evasión con dos obstáculos.	54
Figura 6.8 Gráfica de evasión de obstáculo con fórmula de obstáculo diferente.	55
Figura 6.9 Switch para evitar el paso de energía.	56
Figura 6.10 Plataforma creada con las dos placas de acrílico.	57
Figura 6.11 Marco de soporte para los sensores infrarrojos.	57
Figura 6.12 Programa de Labview para determinar el ángulo del robot móvil con respecto al plano horizontal del ambiente de trabajo.	58
Figura 6.13 Obstáculo con forma hexagonal de caras planas.	59
Figura 6.14. Obstáculo con forma circular	59
Figura 6.15 Obstáculo con forma hexagonal de caras curvas.	59
Figura 6.16. Unidad de medida ejemplificada en las losas para el control propuesto.	60
Figura 6.17 Interfaz de usuario en Labview para el control punto a punto implementado.	61
Figura 6.18. Trayectoria Realizada por el Robot Móvil	62

Índice de tablas

Tabla 2.1 Variables de la figura 2.3	16
Tabla 2.2 Variables a trabajar en el modelo dinámico.	18
Tabla 2.3 Variables de control punto a punto.	21
Tabla 2.4 Variables de apoyo para el cálculo del control	21
Tabla 6.1 Ecuaciones de control del robot móvil	62

Introducción

Planteamiento del problema

Para mejorar la producción en la industria en los últimos años, se ha aplicado el uso de máquinas para sustituir la mano de obra humana para aumentar la velocidad de la producción y así aumentar su número de ventas. Este crecimiento en la aplicación de máquinas o robots en la industria ha llevado a crear diferentes estrategias de control para que estos dispositivos realicen determinada tarea de forma satisfactoria.

Para llevar a cabo el control deseado en un robot se diseñan rutinas de trabajo en forma secuencial, es decir, que realiza un movimiento después de que el anterior se haya terminado de ejecutar. Estas secuencias se programan por medio de la captura de los puntos a donde se mueve el robot o por medio de un programa. Este tipo de control no contempla un cambio en el área de trabajo del robot. Esta problemática introdujo la necesidad de la aplicación de sensores para que el robot se pueda adaptar, pero cuando estos sensores se activan el robot deja de trabajar; lo que lleva ahora a una solución en tiempo real donde el robot no deje de trabajar mientras se adapta a la situación de cambio.

Por medio del control reactivo se demuestra cómo un obstáculo puede considerarse como un campo potencial de repulsión, de una determinada magnitud; así, a mayor magnitud, mayor repulsión, de tal manera que al acercarse el robot al obstáculo “experimentará” una fuerza de repulsión que evitará que el robot choque con el obstáculo. En cuanto a la plataforma experimental, es necesario adecuar son sensores que permitan detectar la presencia de los objetos y obtiene una mejor respuesta de control en los actuadores a través de la lectura en tiempo real de los sensores que, en cuanto detecten la situación cambiante, el robot a través de los actuadores empezará a tomar las medidas necesarias para adaptarse a tal situación.

Justificación

La robótica móvil se considera actualmente un área de la tecnología avanzada manejadora de problemas de alta complejidad. Sus productos se constituyen en aplicaciones de las áreas de control, programación, inteligencia artificial, percepción e instrumentación, y sirven de base para el avance en diversos campos de la industria, aportando soluciones tecnológicas innovadoras orientadas al desarrollo de mejores robots y a la ampliación del abanico de aplicaciones disponibles

Sin embargo, muchas veces se presentan limitaciones debido a factores relacionados con las imprecisiones de los algoritmos de control (por su propia naturaleza), así como de las imprecisiones de los elementos relacionados con la instrumentación.

El estudio de la dinámica de los robots, permite conocer el comportamiento del sistema. Con esta información, es posible diseñar y/o seleccionar estrategias de control en base a su desempeño particular.

Propuesta de solución de tesis

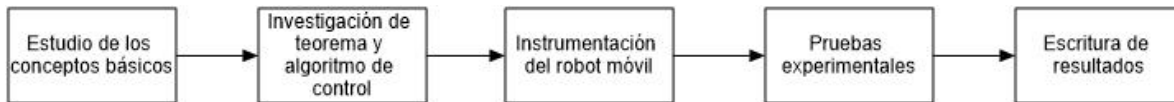


Figura I. Diagrama a bloques del sistema.

Objetivos

Para llevar a cabo el proyecto de tesis, se fijaron los siguientes objetivos.

Objetivo General:

Controlar la navegación de un robot móvil terrestre, mediante una estrategia de control reactivo, en la evitación de obstáculos.

Objetivos específicos:

1. Estudiar los conceptos de robots móviles
2. Estudiar la plataforma del robot móvil HUMMER H2
3. Estudiar el modelo cinemático y dinámico del robot HUMMER
4. Puesta a punto de la instrumentación de la plataforma
5. Implementación de una estrategia de control
6. Pruebas experimentales
7. Publicación de Resultados
8. Escritura de tesis.

Capítulo 1

Antecedentes

1.1 Historia

Cuando se menciona la palabra robótica uno pensaría que se habla de tecnología reciente, pero la robótica es más antigua de lo que se cree. Las primeras apariciones de la robótica surgen en el siglo I a. C. realizando diversos diseños de máquinas y autómatas, destacando entre sus principales inventores a Herón de Alejandría creador de “Autómata”, el primer libro de robótica de la historia.

El término robot aparece por primera vez en la obra teatral R.U.R. (Rossum’s Universal Robots) del novelista y autor dramático checo Karel Capek en cuyo idioma la palabra “robota” significa fuerza de trabajo o servidumbre. Por aquellos años la producción en grandes series se había introducido en numerosas fábricas.

El desarrollo de robots móviles responde a la necesidad de extender el campo de aplicación de la robótica, restringiendo inicialmente al alcance de una estructura mecánica anclada en uno de sus extremos. Se trata también de incrementar la autonomía limitando todo lo posible la intervención humana.

Desde el punto de vista de la autonomía, los robots móviles tienen como precedentes los dispositivos electromecánicos, tales como los denominados micro-mouse, creados desde los años treinta para desarrollar funciones inteligentes tales como descubrir caminos en laberintos.

Aunque la robótica tiene sus inicios hace más de quinientos años, no es hasta el año 1966 en que se desarrolla formalmente por la universidad de Stanford la disciplina conocida como “Robótica Móvil”, siendo “Shakey” el primero en su tipo.

En los años setenta se vuelve a trabajar en el desarrollo de robots móviles dotados de una mayor autonomía. La mayor parte de las experiencias se desarrollan empleando plataformas que soportan sistemas de visión. Sin embargo, el desarrollo tecnológico todavía no era el suficiente para lograr la navegación autónoma de forma eficiente. En los años ochenta el incremento espectacular de la capacidad computacional y el desarrollo de nuevos sensores, mecanismos y sistemas de control, permite aumentar la autonomía. En esta década cabe mencionar los desarrollos de robots móviles, tanto para interiores como para navegación en exterior, realizados en la Carnegie Mellon University (Pittsburgh EE.UU.) [1], [2].

1.2 Estado del Arte

1.2.1 Construyendo un Modelo Eléctrico de un Vehículo e Implementando un Algoritmo de Evitación de Obstáculos

En este trabajo se presenta un modelo eléctrico de un vehículo con su propio sistema de adquisición y cómputo de información (Figura 1.1), en el cual se prueba el algoritmo de control de colisión en interiores y exteriores con localización GPS. El algoritmo usado en este proyecto es el método de campo potencial. Se escogió usar un sensor LIDAR que puede adquirir la distancia del ambiente en 2D. El principal objetivo de esta tesis es realizar evasión de obstáculos con un robot móvil en tiempo real. [3]

A diferencia del trabajo que se encuentra en proceso, los sensores a ocupar son más básicos, el robot móvil tiene una configuración diferente y solo se hace prueba en interiores.



Figura 1.1. Modelo Eléctrico

1.2.2 Algoritmo de Detección y Evitación de Obstáculos Basado en un 2-D LIDAR.

La habilidad de evitación de obstáculos es de gran importancia en los robots móviles terrestres, y la garantía básica de que el robot móvil realice diversas tareas. Las tecnologías de evitación de obstáculos son divididas en dos tipos, una es basada en el mapeo global y otra en los sensores correspondientes.

Este documento se enfoca en la evitación de obstáculos locales basado en el método de sensores. El estudio de la detección y evitación de obstáculos son dos partes inseparables en la investigación en la habilidad de evitar obstáculos. Este documento propone una detección de obstáculos eficiente y un algoritmo de evitación basado en un LIDAR 2D. Un método es propuesto para obtener la información de los obstáculos por filtrado y agrupamiento de la información del apuntador laser. También este método genera el ángulo frontal y velocidad del robot basado en el principio de la función del coste mínimo. El algoritmo de detección y evasión de obstáculos tiene ventajas de un simple modelo matemático y buen desempeño en tiempo real. La efectividad del algoritmo es propuesta y verificada en la plataforma de simulación de MATLAB. [4]

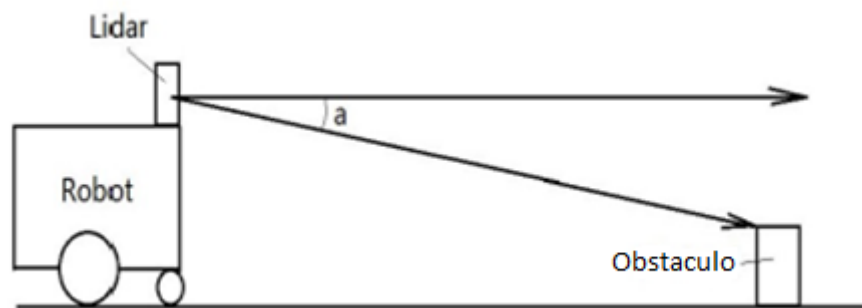


Figura 1.2. Ángulo de detección de LIDAR

1.2.3 Diseño e Implementación de un Algoritmo Difuso de Evasión de Obstáculos en base de una Programación de Ganancia.

Este artículo presenta un nuevo algoritmo de evitación de obstáculos. Usando la combinación de lógica difusa y la teoría de programación de ganancia, una nueva metodología que reduce el costo computacional comparado a la metodología difusa convencional, especialmente cuando las variables a controlar son demasiadas. Para propósitos de comparación, se implementó un algoritmo de campo potencial. Ambos algoritmos son examinados en una serie de experimentos para determinar si el nuevo algoritmo es al menos tan bueno como el algoritmo de campo potencial. Los parámetros definidos para estos experimentos son: el número de veces que el agente colisiona (colisiones), el tiempo empleado para terminar la trayectoria propuesta (tiempo empleado) y la energía restante del agente al finalizar el experimento (consumo de energía). Los resultados muestran que el algoritmo propuesto alcanza un nivel bajo de colisiones. También, el algoritmo propuesto muestra una mejora considerable en el tiempo empleado para completar las tareas propuestas. Finalmente, los resultados demuestran una considerable reducción en la energía de consumo usando el algoritmo propuesto en comparación al algoritmo de campo potencial. [5]

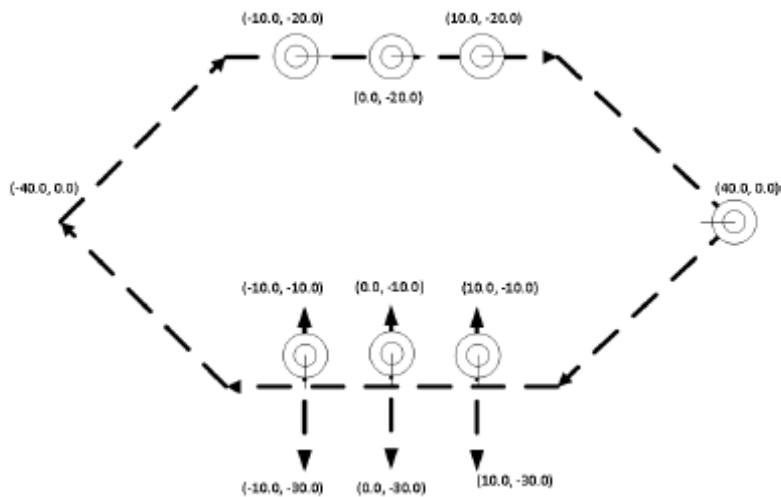


Figura 1.3. Trayectoria con obstáculos.

1.2.4 Algoritmo de Evasión y Sensado de Múltiples Objetivos para Obstáculos Estáticos y Dinámicos.

En este trabajo se presenta un Nuevo algoritmo computacional eficiente S&A para la implementación en tiempo real de aplicaciones para vehículos aéreos no tripulados. Basado en la optimización de aproximación simplificada, el algoritmo procesado tiene como objetivo proveer una resolución de maniobra confiable para múltiples escenarios los cuales incluyen obstáculos/objetivos terrestres y aéreos. En presencia de un riesgo de conflicto, la maniobra de evasión es definida como una variación de pasto en la variación de altitud del comando de pilotos automáticos. Este comando de paso es optimizado con el fin de mantener la mínima distancia de separación entre la propia nave y todos los objetivos durante la maniobra total. El algoritmo procesa la distancia de separación entre el vehículo aéreo no tripulado y los objetivos calculando futuras trayectorias en cada paso de tiempo de ambos la nave y el objetivo, mientras toma en cuenta el desempeño de la nave dentro de los límites. Los algoritmos fueron evaluados en simulación, donde los objetivos terrestres son derivados de las elevaciones del suelo, mientras que los objetivos aéreos la nave aérea comunica su información a través de un transpondedor ADS-B modo S. [6]

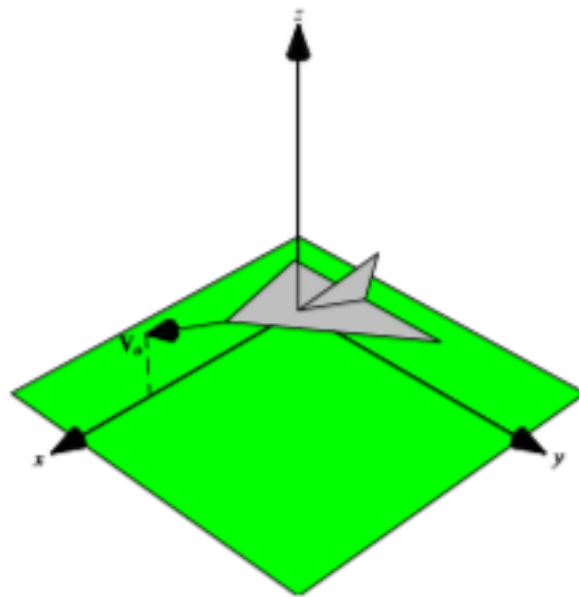


Figura 1.4. Orientación Cartesiana considerada

1.2.5 Documento de Revisión de Detección, Sensado y Evasión para Sistemas Aéreos no Tripulados.

Este documento de revisión de detección, sensado y evasión (DCE) incluye tecnologías y enfoques que pueden ser usados en sistemas aéreos no tripulados (SAT) y eso permitió a la Administración Federal de Aviación a entender los tipos de DCE disponibles para SAT operando en el Sistema nacional Aeroespacial. Más de 1000 artículos fueron examinados y más de 150 referencias y/o citas relacionadas a tecnologías potencialmente útiles fueron examinadas para esta revisión. [7]

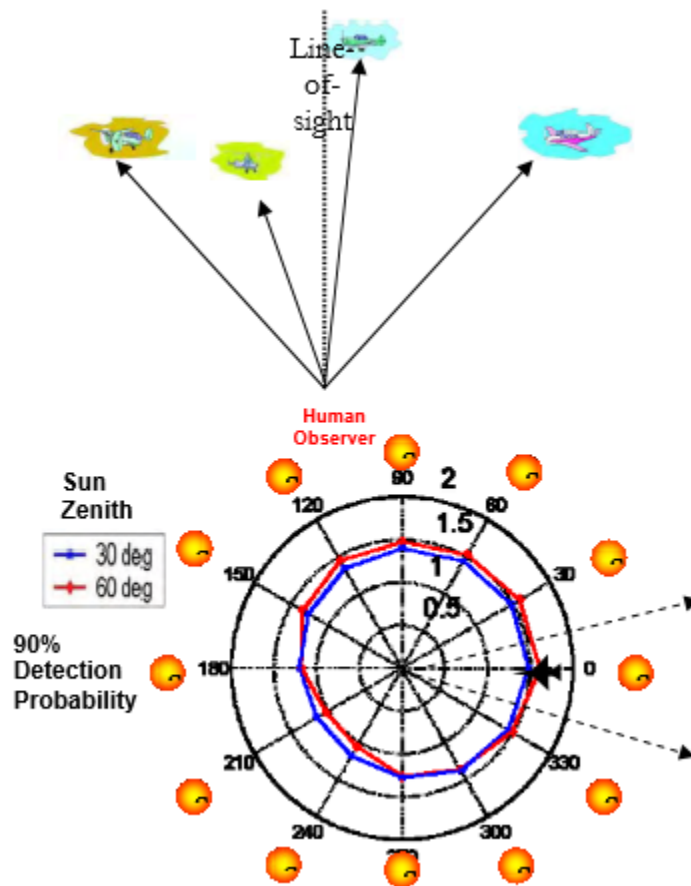


Figura 1.5. Análisis de Rango y Respuesta.

1.2.6 Propuesta de un algoritmo para la planeación de trayectorias de robots móviles empleando campos potenciales y enjambres de partículas activas brownianas.

En este documento se propone un algoritmo para la planificación de trayectorias basado en el movimiento de partículas activas Brownianas. Uno de los métodos más populares en la planificación de trayectorias es el basado en campos potenciales artificiales. Sin embargo, este método tiene la desventaja de presentar mínimos locales que pueden hacer que el robot no llegue al punto destino. El algoritmo propuesto emplea un modelo de partículas activas Brownianas con comportamientos circulares lo cual permite escapar de mínimos locales. También se realiza un análisis aproximado para la versión determinista del modelo seleccionado, comprobando mediante simulaciones que los resultados obtenidos son consistentes con el comportamiento del sistema. El algoritmo propuesto se probó en cinco casos observando tanto resultados cualitativos como cuantitativos del desempeño del algoritmo. [8]

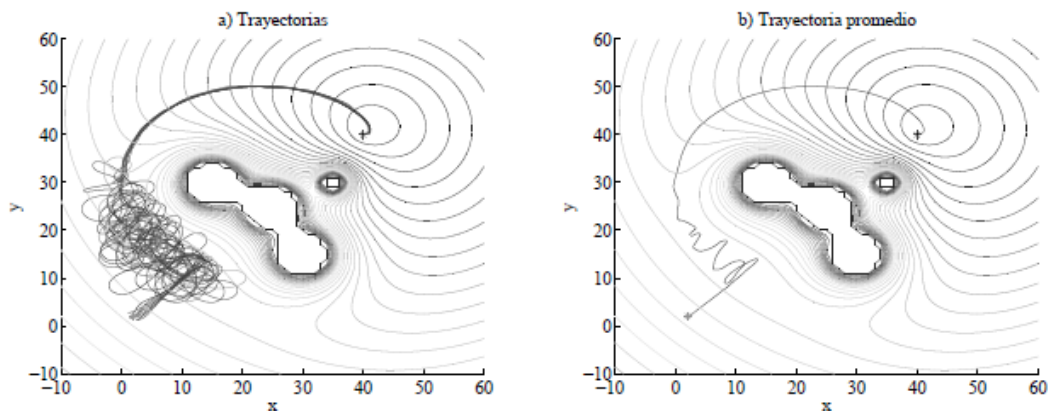


Figura 1.6. Comparación entre las trayectorias planteadas.

Capítulo 2

Robots móviles y Modelado

2.1 Características de Robot Móviles

Se define a un robot móvil como un dispositivo conformado de componentes físicos y computacionales, divididos en cuatro subsistemas específicamente denotados por: Locomoción, Percepción, Razonamiento y Comunicación.

2.1.1 Locomoción

La locomoción es la capacidad de los robots móviles para desplazarse de un lugar a otro. Los robots móviles se pueden clasificar según los elementos que emplean para realizar la locomoción con sus respectivos actuadores.

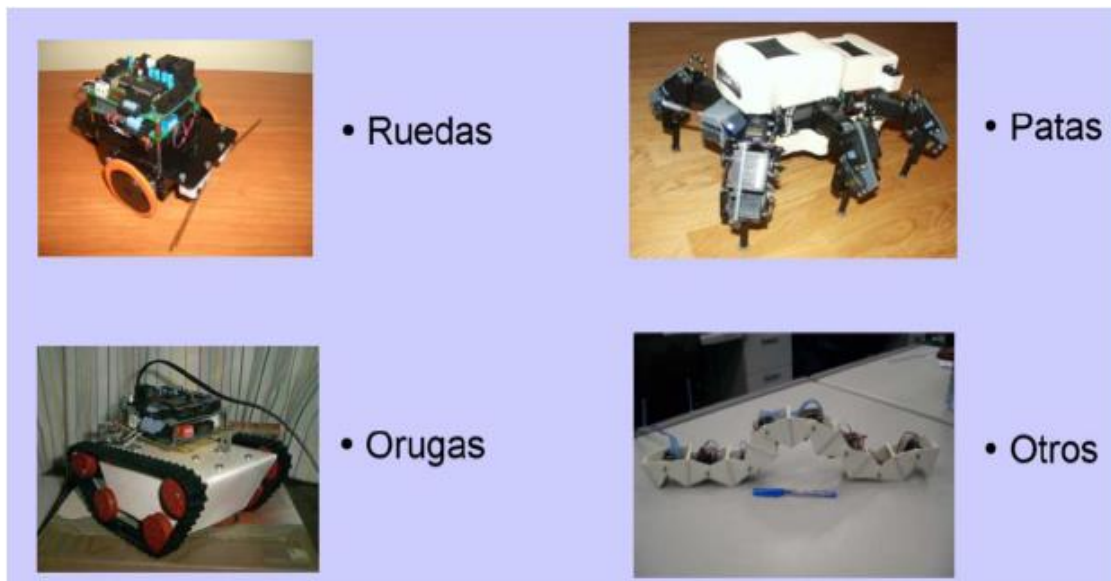


Figura 2.1 Clasificación de robots móviles por locomoción.

Robots Móviles con Ruedas

Los robots móviles terrestres poseen diversas aplicaciones en la industria, tales como el bodegaje, inspección, control de producción a distancia, entre otras.

Los vehículos de ruedas son los más populares por varias razones prácticas, son más sencillos y más fáciles de construir, la carga que pueden transportar es mayor a la que pueden soportar otros tipos de robots móviles. A esto se puede añadir el que se pueden utilizar vehículos de ruedas de radio control para usarlos como bases de robots. La principal desventaja de las ruedas es su empleo en terreno irregular. Normalmente un vehículo de ruedas podrá sobrepasar un obstáculo que tenga una altura no superior al radio de sus ruedas.

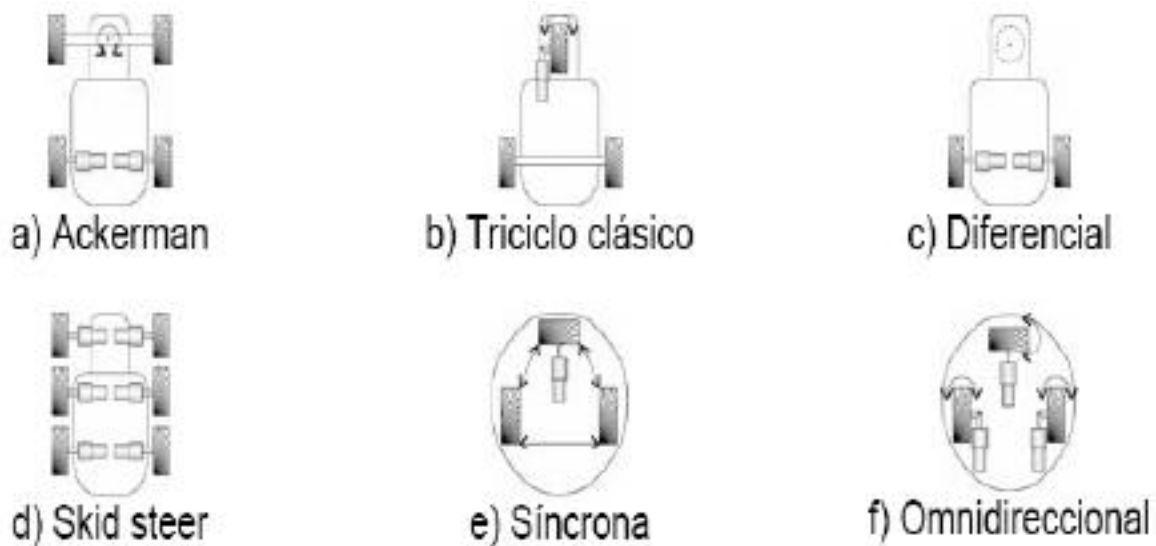


Figura 2.2 Clasificación de robots con ruedas.

2.1.2 Percepción

La percepción es un proceso que le permite al robot o sistema, a través de los sentidos (sensores), recibir, elaborar e interpretar la información proveniente de su área de trabajo.

2.1.5 Sistema Holonómico

Un sistema holonómico es el cual donde el número de grados de libertad son iguales al número de coordenadas necesarias para especificar la configuración del sistema, es decir que el robot móvil no requiere de una segunda acción de los actuadores para moverse en una dirección.

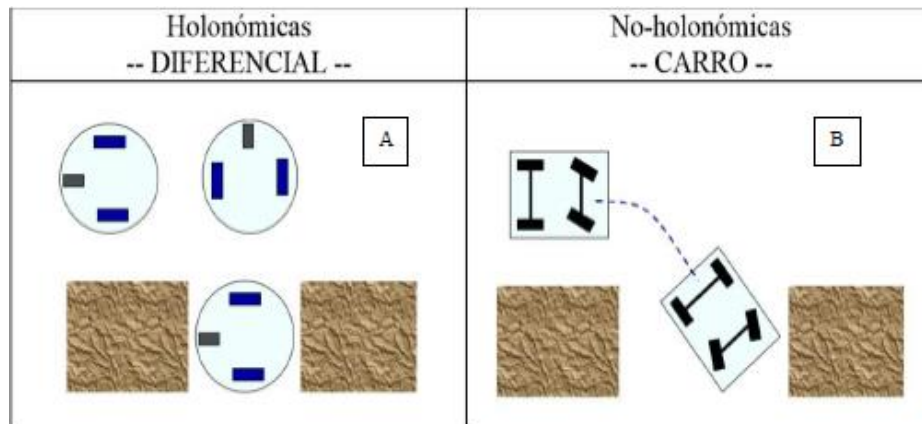


Figura 2.4 Sistemas Holonómicos y no Holonómicos.

En el caso del robot móvil tipo Ackerman, el cual es el modelo de robot móvil con el que se trabaja en este proyecto, es considerado un sistema no holonómico de acuerdo a lo establecido en la figura 2.4 ya que para poder cambiar de dirección requiere la activación de un segundo actuador

2.2 Modelado

Modelar un sistema en sus ecuaciones cinemáticas y dinámicas permite comprender al investigador las propiedades y comportamiento del sistema a partir de diferentes parámetros definidos.

El modelo cinemático describe el movimiento del sistema sin considerar las fuerzas externas que lo causan. Para poder obtener el modelo cinemático de un robot móvil con ruedas generalmente se toman en cuenta las siguientes consideraciones:

- El robot se mueve sobre una superficie plana.
- No existen elementos flexibles en la estructura del robot (incluyendo las ruedas).
- Las ruedas poseen un eje de direccionamiento, que siempre es perpendicular al suelo.
- Se desprecia todo tipo de fricción en los elementos móviles del robot móvil contra el suelo.

En contra posición con el modelo dinámico, el cual describe el movimiento del robot a partir de las fuerzas externas que mueven al robot, comúnmente es el torque que proporcionan los motores a las ruedas del robot móvil.

2.3 Modelo cinemático

En particular, este trabajo se enfocará al robot móvil tipo Ackerman por su semejanza a la configuración de un vehículo estándar, ya que proporciona una gran estabilidad y es aplicable a la vida cotidiana.

La configuración Ackerman se conforma de 4 ruedas distribuidas en 2 ejes que poseen la misma longitud b a una distancia l entre ellos, con las llantas delanteras indicando la dirección en la que se moverá el robot móvil; en donde ésta dirección se define en función de los ángulos θ_0 , θ_1 y Φ , todos ellos relacionados con el sistema de referencia P_1 (Fig. 2.5).

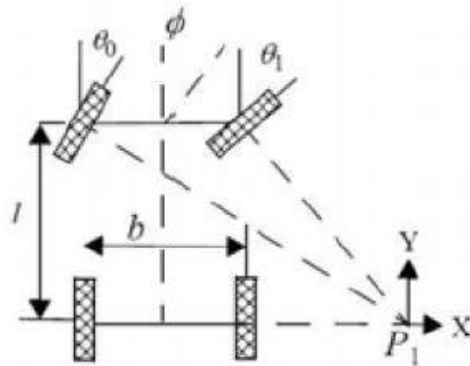


Figura 2.5 Robot móvil tipo Ackerman.

Para obtener el modelo cinemático de este tipo de robot móvil se emplean las restricciones no holonómicas del sistema, estas restricciones asumen que no existe deslizamiento en las ruedas del robot móvil y las mismas se encuentran relacionadas con las velocidades del vehículo. Como resultado la ecuación general de las restricciones no holonómicas para un robot móvil tipo Ackerman queda [9]

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0$$

donde \dot{x} y \dot{y} son las velocidades lineales de las llantas dentro del sistema de coordenadas (x, y) y θ es el ángulo del vehículo con respecto al eje x . Con las restricciones holonómicas establecidas se procede a modelar el robot móvil en el sistema de coordenadas (x, y) simplificando el modelo al de una bicicleta como se ilustra en la Fig. 2.6. [] [] []

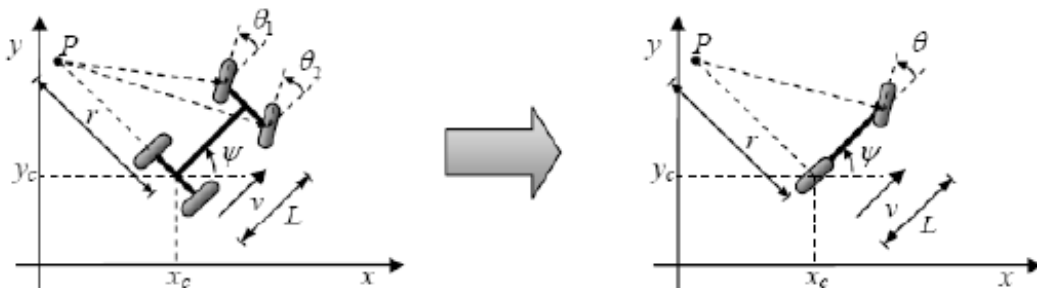


Figura 2.6 Simplificación del modelo del robot móvil

Bajo estas condiciones se replantean las coordenadas generales que permitirán describir el sistema de la Fig. 2.7, donde las variables descritas en el diagrama se explican en la tabla 2.1.

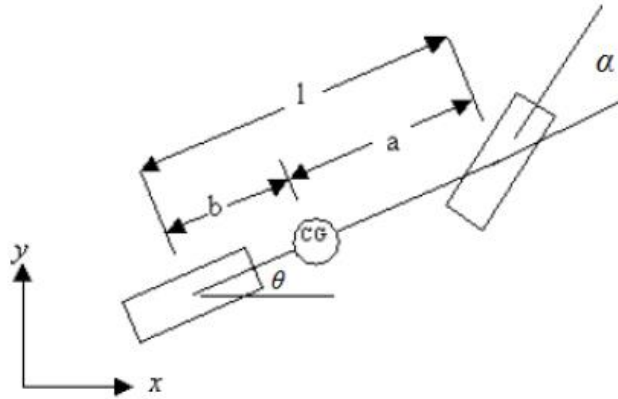


Figura 2.7 Coordenadas generales del robot móvil

l	Distancia entre los ejes de las llantas
b	Distancia entre el eje de la llanta trasera y CG
a	Distancia entre el eje de la llanta delantera y CG
θ	Angulo del robot móvil con respecto al plano (x,y)
α	Ángulo de dirección del robot móvil
CG	Centro de Gravedad del robot móvil

Tabla 2.1 Variables de la figura 2.3

Se plantea (x, y) como la posición del centro de gravedad del vehículo, θ la orientación del vehículo con respecto al eje x y α el ángulo de orientación de la rueda delantera con respecto al eje del cuerpo del robot. Sea (x_1, y_1) la posición de la rueda trasera y (x_2, y_2) la posición de la rueda delantera, definidas por (2) a (5)

$$x_1 = x - b\cos(\theta) \quad x_2 = x + a\cos(\theta) \quad (2)$$

$$y_1 = y - b\sin(\theta) \quad y_2 = y + a\sin(\theta) \quad (3)$$

$$\dot{x}_1 = \dot{x} + \dot{\theta}b\sin(\theta) \quad \dot{x}_2 = \dot{x} - \dot{\theta}a\sin(\theta) \quad (4)$$

$$\dot{y}_1 = \dot{y} - \dot{\theta}b\cos(\theta) \quad \dot{y}_2 = \dot{y} + \dot{\theta}a\cos(\theta) \quad (5)$$

Las restricciones holonómicas para las ruedas trasera y delantera están descritas de la siguiente forma:

$$\dot{x}_1\sin(\theta) - \dot{y}_1\cos(\theta) = 0 \quad (6)$$

$$\dot{x}_2\sin(\theta + \alpha) - \dot{y}_2\cos(\theta + \alpha) = 0 \quad (7)$$

Sustituyendo las ecuaciones (4) y (5) en (6) y (7) obtenemos:

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) + b\theta = 0 \quad (8)$$

$$\dot{x}\sin(\theta + \alpha) - \dot{y}\cos(\theta + \alpha) + a\theta\cos(\theta) = 0 \quad (9)$$

Usando las coordenadas en el centro de masa del vehículo, definidas en la Fig. 2.3, se define el eje u a lo largo del eje central del robot móvil y el eje w perpendicular al eje u , ejemplificado en la Fig. 2.8, se obtienen (10) y (11).

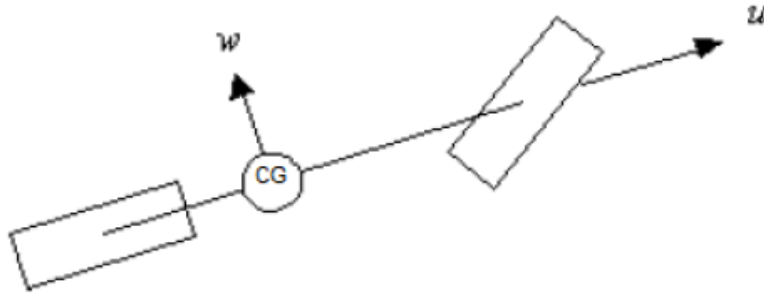


Figura 2.8 Coordenadas del robot móvil.

$$\dot{x} = v_u \cos(\theta) - v_w \sin(\theta) \quad (10)$$

$$\dot{y} = v_u \sin(\theta) + v_w \cos(\theta) \quad (11)$$

donde v_u y v_w son las velocidades del centro de gravedad (CG) a lo largo de los ejes u y w respectivamente. Sustituyendo (10) y (11) en (8) y (9) se obtienen (12) y (13).

$$v_w = b\theta \quad (12)$$

$$\theta = \frac{v_u}{l} \tan(\alpha) \quad (13)$$

2.4 Modelo dinámico

Con el modelo cinemático definido, se procede a determinar el modelo dinámico usando la metodología de Newton-Euler, donde se tienen las fuerzas longitudinales y laterales ejercidas sobre las llantas. Estas fuerzas denotadas por F_{ui} y F_{wi} ($i=1$ para las llantas delanteras e $i=2$ para las llantas traseras) respectivamente. La fuerza de propulsión, F_p , es aplicada por medio de las llantas traseras en la dirección del eje u de las ruedas, m la masa del robot móvil, J el momento de inercia y las distancias a , b , l definidas en la sección anterior; la dinámica del robot móvil está dada por las ecuaciones (14) a (19) [10]:

$$\dot{x} = v_u \cos(\theta) - v_w \sin(\theta) \quad (14)$$

$$\dot{y} = v_u \sin(\theta) + v_w \cos(\theta) \quad (15)$$

$$\dot{\theta} = \frac{v_u}{l} \tan(\alpha) \quad (16)$$

$$\dot{v}_u = v_w \Omega + \frac{F_{u1} \cos(\alpha)}{m} - \frac{F_{w1} \sin(\alpha)}{m} + \frac{F_{u1} + F_p}{m} \quad (17)$$

$$\dot{v}_w = -v_u \Omega + \frac{F_{u1} \sin(\alpha)}{m} + \frac{F_{w1} \cos(\alpha)}{m} + \frac{F_{w2}}{m} \quad (18)$$

$$\dot{\Omega} = \frac{F_{u1} a \sin(\alpha)}{j} + \frac{F_{w1} a \cos(\alpha)}{j} - \frac{F_{w2} b}{j} \quad (19)$$

Donde las variables son definidas en la tabla 2.2 e ilustradas en la figura 2.9.

v_u	Vector velocidad sobre el eje u
v_w	Vector velocidad sobre el eje w
θ	Es el ángulo del robot con respecto a x
α	Es el ángulo de orientación del robot móvil
l	Es la distancia entre los ejes del robot móvil
m	Es la masa del robot móvil
Ω	Es el momento de fuerzas del robot móvil.
j	Es el momento de inercia del robot móvil.
F_{u1}	Es la fuerza ejercida sobre la llanta delantera en el eje u
F_{w1}	Es la fuerza ejercida sobre la llanta delantera sobre el eje w
F_p	Es la fuerza de propulsión otorgada por el motor de tracción del robot móvil
F_{w2}	Es la fuerza ejercida sobre la llanta trasera sobre el eje w

Tabla 2.2 Variables a trabajar en el modelo dinámico.

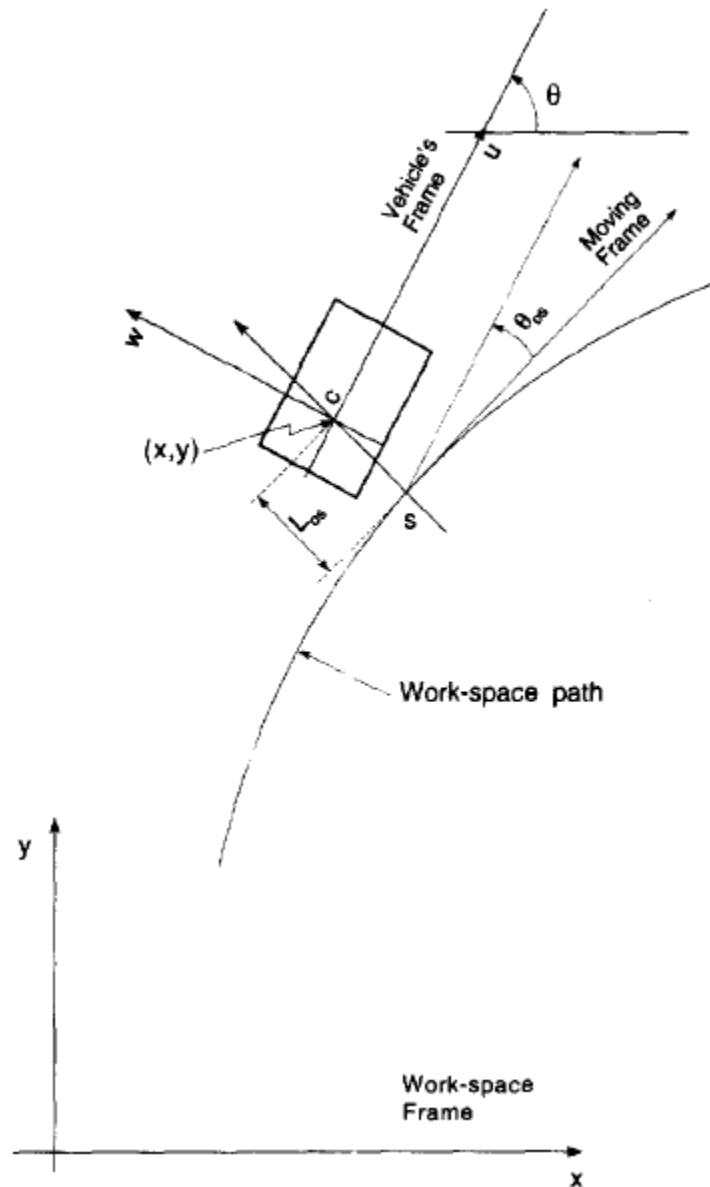


Figura 2.9 Diagrama de espacio de trabajo para un robot móvil tipo Ackerman.

Donde θ_{os} y L_{os} son variables de error que se describirán más adelante.

Este modelo dinámico se complementa con la dinámica de la rueda de dirección la cual se denota por $\dot{\alpha} = F_s$, donde F_s es la variable de control para modificar el ángulo de orientación del robot móvil.

Considerando que no existe deslizamiento en las ruedas, el modelo dinámico del robot móvil tipo Ackerman, queda descrito por las ecuaciones (20) a (27):

$$\dot{v}_u = A_p F_p + A_s F_s \quad (20)$$

$$v_w = \frac{v_u b}{l} \tan(\alpha) \quad (21)$$

$$\dot{\theta} = \frac{v_u}{l} \tan(\alpha) \quad (22)$$

$$\dot{\alpha} = F_s \quad (23)$$

$$\dot{x} = v_u \cos(\theta) - v_w \sin(\theta) \quad (24)$$

$$\dot{y} = v_u \sin(\theta) + v_w \cos(\theta) \quad (25)$$

$$A_p = \left(m + \left(\frac{\tan(\alpha)}{l} \right)^2 (mb^2 + j) \right)^{-1} \quad (26)$$

$$A_s = \frac{(mb^2 + j) * v_u \tan(\alpha)}{(l^2) \cos^2(\alpha)} A_p \quad (27)$$

Donde las variables A_p y A_s son elementos que ayudan a resumir la ecuación 20 con el propósito de tener una mejor visualización de las ecuaciones.

Con la dinámica planteada para el robot móvil tipo Ackerman, queda por plantear el control que nos permitirá simular el movimiento que trazaría el robot móvil en un plano cartesiano (x, y) , para ello se plantea un control punto a punto.

(x_d, y_d)

2.5 Control punto a punto

Para llevar a cabo el control punto a punto del robot móvil se plantea como variables de entrada las coordenadas de un punto fijo al cual se desea llegar (posición deseada) y como salida las fuerzas de control F_p y F_s , para ello se plantean (28) a (31) para llevar a cabo la tarea deseada [10].

El control de dirección está dado por (29) y el control de propulsión por (30).

$$F_s(t) = -k_s[\theta_{os}(t) l_{os}(t) \alpha_{os}(t)]' \quad (28)$$

$$F_p(t) = F_{p1}(t) + F_{p2}(t) \quad (29)$$

$$F_{p1}(t) = A_p^{-1}(\dot{v}_{ud} - A_s F_s) \quad (30)$$

$$F_{p2}(t) = A_p^{-1}(k_{p1} v_{os}) \quad (31)$$

donde

\dot{v}_{ud}	es la aceleración deseada del robot móvil
k_s	es un vector de ganancias constantes positivas
$\theta_{os} = \theta - \theta_d$	es el error del ángulo theta del robot móvil
$\alpha_{os} = \alpha - \alpha_d$	es el error del ángulo alfa
$l_{os} = l - l_d$	es el error de posición en el plano (x, y) del robot móvil con respecto a la trayectoria deseada
k_{p1}	es una ganancia constante positiva

Tabla 2.3 Variables de control punto a punto.

para obtener los valores de θ_d , l_{os} y α_d se ocupan (32) a (37).

$$\alpha_d = k_\alpha(\tan^{-1}(\frac{y_d - y}{x_d - x}) - \theta) \quad (32)$$

$$\theta_d = \tan^{-1}(\frac{y_d - y}{x_d - x}) \quad (33)$$

$$l_{os} = -\{(x - x_d) * \cos(\theta_d - \theta_{vd})\} + \{(y - y_d) * \sin(\theta_d - \theta_{vd})\} \quad (34)$$

$$\theta_{vd} = \tan^{-1}(\frac{\tan(\alpha_d)}{2}) \quad (35)$$

$$V_{ud} = k_u * (x_d - x) \quad (36)$$

$$V_{wd} = k_w * (y_d - y) \quad (37)$$

donde

k_α, k_u, k_w	son constantes de ganancia positiva
y	es la posición actual del robot en el eje y
x	es la posición actual del robot en el eje x
y_d	es la posición deseada del robot en el eje y
x_d	es la posición deseada del robot en el eje x

Tabla 2.4 Variables de apoyo para el cálculo del control

Capítulo 3

Métodos de control para la evitación de obstáculos

Dentro de la categoría de robots móviles con ruedas, la principal tarea a desarrollar es evadir obstáculos durante su tránsito hacia la meta, para ello se han desarrollado series de algoritmos con la finalidad de minimizar errores y proporcionar una mayor autonomía cada vez más certera en el tránsito del robot móvil. Ante tal finalidad las referencias que se citan a continuación son las más representativas sobre robots móviles. En particular nos enfocaremos al problema de la evasión de obstáculos empleando campos potenciales con robots móviles cuya tracción principal son ruedas. La teoría de control moderno circunscribe al robot móvil de ruedas convencionales como un sistema no-holonómico, cuando desarrolla la tarea de evasión de obstáculos, debido a la restricción que impone el elemento motriz convencional en el desplazamiento, en dirección paralela al eje de rotación, considerando que el movimiento se realiza sin deslizamiento.

El problema a solucionar consiste en evadir obstáculos en espacios de trabajo desconocidos o parcialmente conocidos, respetando las restricciones cinemáticas inherentes al modelo hasta que el móvil arribe a la meta, la diversidad de técnicas para lograr esto comienza con el desarrollo de algoritmos primitivos que detienen el tránsito del robot al momento de detectar un obstáculo; continuando con algoritmos sofisticados con capacidad de predecir la trayectoria del móvil al detectar un objeto y finaliza con algoritmos complejos con base a múltiples mediciones de sensores que detectan la dimensión y ubicación del obstáculo y logran planificar rutas hasta la meta. Del conjunto de estos algoritmos en la tarea de evasión de obstáculos, se identifican las siguientes tendencias:

CONTROL REFLEXIVO.

Esta estrategia de control tiene como característica la ausencia de algún método cognoscitivo, es decir, no existe alguna metodología que planifique el tránsito. Con base a simples funciones detectan el obstáculo mediante algún tipo de sensor y ejecutan acciones básicas de evasión o seguimiento predefinido. La magnitud de respuesta depende de la magnitud del estímulo y se caracteriza por tiempos de respuesta rápidos.

CONTROL REACTIVO.

Está basado en el modelo tradicional de inteligencia artificial de la cognición humana; siendo típicos algoritmos sofisticados y complejos que generalmente inician con la percepción del medio ambiente por algún tipo de sensor que provee la información al procesador digital, el cual generalmente construye un modelo del medio ambiente, tomando decisiones para la planificación del siguiente tránsito.

Posee un nivel jerárquico de control. Son necesarios grandes recursos de cómputo si se requiere que trabajen en tiempo real, estimulando un sistema electromecánico con tiempo de respuesta relativamente lento.

Para este trabajo se enfoca en el control reactivo y se propone ocupar el método de campos potenciales artificiales.

3.1 Método de campos potenciales artificiales

Este método trata al robot móvil como una partícula que se encuentra bajo la influencia de un campo potencial artificial denominado U (Figura 3.1). La función del potencial es típicamente definida como la suma de un potencial de atracción que atrae al robot a la meta designada y un potencial de repulsión que aleja al robot de los obstáculos. Cabe recalcar que el campo potencial mencionado en este método es hipotético, es decir, no existe de forma real y en cambio se usa una ecuación matemática que lo representa [11].

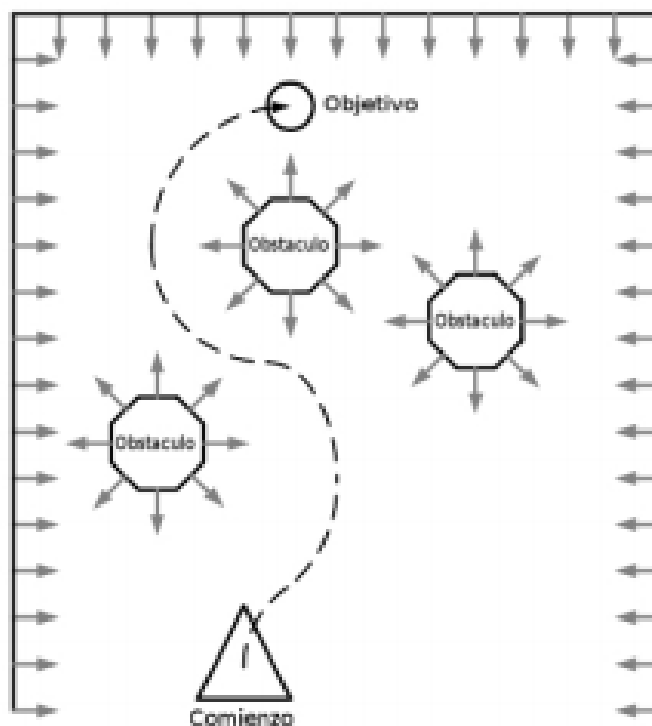


Figura 3.1. Campo potencial artificial para evitación de obstáculos.

El método de campo potencial es mencionado a veces como un método local. Esto es porque la mayoría de las funciones potenciales son definidas de tal forma que sus valores son independientes de la distribución y formas de los obstáculos dentro de un ambiente definido.

Para implementar el método propuesto, se plantea lo siguiente.

3.1.1 Estructura general

El campo de fuerzas artificiales $\vec{F}(q)$ en C , donde C denota el universo del ambiente en el cual se mueve el robot, es producido por una función potencial diferenciable $U: C_{free} \rightarrow \mathbb{R}$, con:

$$\vec{F}(q) = -\nabla\vec{U}(q) \quad (38)$$

donde $\nabla\vec{U}(q)$ denota el vector gradiente de U evaluado en q , donde q representa la posición; de esta manera, se puede reescribir como $q=(x, y)$ o $q=(x,y,z)$ según sea el caso:

$$\nabla\vec{U}(q) = \begin{pmatrix} \partial U / \partial x \\ \partial U / \partial y \end{pmatrix} \quad (39) \quad \nabla\vec{U}(q) = \begin{pmatrix} \partial U / \partial x \\ \partial U / \partial y \\ \partial U / \partial z \end{pmatrix} \quad (40)$$

para hacer que el robot sea atraído hacia la meta designada, mientras se aleja de los obstáculos, U se denota como la suma de dos funciones potenciales elementales

$$U(q) = U_{att}(q) + U_{rep}(q) \quad (41)$$

donde U_{att} es el potencial de atracción asociado a la meta designada como q_{meta} y U_{rep} es el potencial relacionado con la región del obstáculo, independientes una de la otra

con estas convenciones, \vec{F} es la suma de dos vectores:

$$\vec{F}_{att} = -\nabla\vec{U}_{att} \quad (42) \quad \text{y} \quad \vec{F}_{rep} = -\nabla\vec{U}_{rep} \quad (43)$$

las cuales son llamadas fuerzas de atracción y repulsión respectivamente.

3.1.2 Campo potencial de atracción

El campo potencial de atracción U_{att} se define como:

$$U_{att}(q) = \frac{1}{2} \varepsilon \rho_{meta}^2(q) \quad (44)$$

donde ε es un factor escalar positivo y $\rho_{meta}(q)$ denota la distancia euclidiana de q a q_{meta} , dada por la expresión $\|q - q_{meta}\|$ la cual es diferenciable en C . La función U_{att} es positiva o nula; y llega a su mínimo en q_{meta} , esto es $U_{att}(q_{meta}) = 0$.

En todas las configuraciones de q , la fuerza artificial de atracción \vec{F}_{att} derivada de U_{att} es:

$$\begin{aligned} \vec{F}_{att}(q) &= -\nabla \vec{U}_{att}(q) \\ &= -\varepsilon \rho_{meta}(q) \nabla \vec{\rho}_{meta}(q) \\ &= -\varepsilon (q - q_{meta}) \end{aligned} \quad (45)$$

3.1.3 Campo potencial de repulsión

La idea principal dentro de la definición del potencial de repulsión es la de crear una barrera alrededor del obstáculo la cual no puede ser atravesada por el robot. Además, es deseable que el potencial de repulsión no afecte al movimiento del robot cuando se encuentre lo suficientemente alejado de los obstáculos. Una forma de conseguir estos límites es el definir la función potencial de la siguiente forma:

$$U_{rep}(q) = \begin{cases} \frac{1}{2} \eta \left(\frac{1}{\rho(q)} - \frac{1}{\rho_0} \right)^2 & \text{if } \rho(q) \leq \rho_0 \\ 0 & \text{if } \rho(q) > \rho_0 \end{cases} \quad (46)$$

donde η es un factor escalar positivo, $\rho(q)$ denota la distancia del robot a la región del obstáculo, es decir la distancia al campo artificial de repulsión, y ρ_0 es una constante positiva denominada como la distancia de influencia de la región de potencial de los obstáculos. La función U_{rep} es positiva y nula; tiende al infinito mientras el robot se acerca al obstáculo y es nulo cuando la distancia entre el robot y el obstáculo es más grande que ρ_0 .

Si el campo artificial de repulsión es una región convexa con un límite diferenciable en trozos, ρ es diferenciable en C . La fuerza de repulsión artificial derivada de U_{rep} es:

$$\begin{aligned} \vec{F}_{rep}(q) &= -\nabla \vec{U}_{rep}(q) \\ &= \begin{cases} \eta \left(\frac{1}{\rho(q)} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2(q)} \nabla \vec{\rho}(q) & \text{if } \rho(q) \leq \rho_0 \\ 0 & \text{if } \rho(q) > \rho_0 \end{cases} \end{aligned} \quad (47)$$

donde $\vec{\nabla} \rho(q)$ se denota como la distancia euclidiana del robot al obstáculo [11].

Capítulo 4

Plataforma experimental del Robot móvil

La plataforma experimental con la que se cuenta es la mostrada en la Figura 4.1, la cual fue utilizada en (12), bajo otras condiciones en los componentes de sensado de las variables físicas, ya que el modelado tuvo un enfoque distinto, así como en los actuadores. Motivo por el cual, en el presente trabajo, se tuvieron que hacer varias adecuaciones a ella.



Figura 4.1 Robot móvil HUMMER H2.

Es un robot móvil HUMMER H2 (apéndice A), clasificado como tipo Ackerman, el cual cuenta con las siguientes etapas:

4.1 Estructura mecánica

Carcasa

Es colocada en la parte superior del vehículo y tiene adaptadas las luces delanteras y traseras como indicadores de control. Su peso es de 460 gramos y se ilustra en la figura 4.2.



Figura 4.2 Carcasa del robot móvil HUMMER H2.

Muelle Delantero

Sujeta las 2 ruedas delanteras de dirección, tiene un acoplamiento para el eje transversal del vehículo y dentro de su estructura se encuentra el servomotor que dirige la dirección del vehículo. Su peso es de 600 gramos y se ilustra en la figura 4.3.



Figura 4.3 Muelle delantero del robot móvil HUMMER H2.

Muelle Trasero

Sujeta las 2 ruedas traseras de tracción, tiene un acoplamiento para el eje transversal del vehículo y dentro de su estructura se encuentra el motor de corriente directa que genera la tracción del vehículo. Su peso es de 594 gramos (sin considerar el peso del motor) y se ilustra en la figura 4.4.



Figura 4.4 Muelle trasero del robot móvil HUMMER H2.

Porta Batería

Compartimiento dedicado para alojar la batería. Su peso es de 236 gramos (sin considerar el peso de la batería) y se ilustra en la figura 4.5.



Figura 4.5 Porta batería del robot móvil HUMMER H2.

Batería

Batería recargable sellada, de ácido-plomo, de 12 Volts, 4 Amperes/hora y hasta 1000 ciclos de carga/descarga. No tiene efecto de memoria y se coloca en la parte trasera del vehículo. Su peso es de 1.603 kilogramos y se ilustra en la figura 4.6.



Figura 4.6 Batería de alimentación del robot móvil HUMMER H2.

Llantas

Posee 4 llantas tipo estándar, cada llanta tiene un diámetro de 0.1305 metros y un perímetro de 0.41 metros. Su peso individual es de 250 gramos y se ilustran en la figura 4.7.



Figura 4.7 Llantas tipo estándar del robot móvil HUMMER H2.

4.2 Actuadores

El robot móvil HUMMER H2, posee 2 motores actuadores, uno se encarga de dirigir la dirección y el otro se encarga de generar la tracción del vehículo.

Servomotor de Dirección

Está formado por una caja de engranes metálicos los cuales proporcionan un torque máximo de 15 kilogramos a la salida del servomotor. Su peso es de 250 gramos y se ilustra en la figura 4.8.



Figura 4.8 Servomotor de dirección del robot móvil HUMMER H2.

Motor de Tracción

Es un motor de corriente directa de 12 volts, genera hasta 6000 rpm., produciendo un elevado torque, debido al sistema de engranaje al que se encuentra adaptado. Su peso es de 450 gramos y se ilustra en la figura 4.9.



Figura 4.9 Motor de tracción del robot móvil HUMMER H2.

4.3 Sensores

Para obtener la medición de las variables físicas (posición, orientación, dirección, distancia entre el robot y el obstáculo) del robot móvil HUMMER H2, se cuenta con cuatro sensores los cuales proporcionan los datos necesarios para que, mediante un procesamiento de señales adecuado, se obtenga la información buscada en las condiciones deseadas.

Encoder en Cuadratura

Cuenta con una resolución de 80 pulsos/revolución, además de proporcionar la posición del vehículo, también nos otorga la información del sentido de giro del motor de tracción. Su peso es de 50 gramos y se ilustra en la figura 4.10.



Figura 4.10 Encoder utilizado como sensor de posición del robot móvil HUMMER H2.

Potenciómetro

Se encuentra empotrado dentro de la caja de engranes del servomotor de dirección, que forma parte de la infraestructura disponible inicial. Nos otorga una salida variable de voltaje que al procesar se transforma en la señal de información de dirección de las ruedas delanteras del vehículo. Su peso es de 25 gramos y se ilustra en la figura 4.11.



Figura 4.11 Potenciómetro utilizado como sensor de la dirección del robot móvil HUMMER H2.

Sensor infrarrojo

El sensor infrarrojo que se implementa para medir la distancia entre el robot móvil y el obstáculo son de la marca SHARP con una salida de voltaje analógica la cual se caracterizó para determinar la distancia mínima eficaz para su aplicación, se ilustra en la figura 4.12 y cuyos datos técnicos se encuentran en el apéndice A.

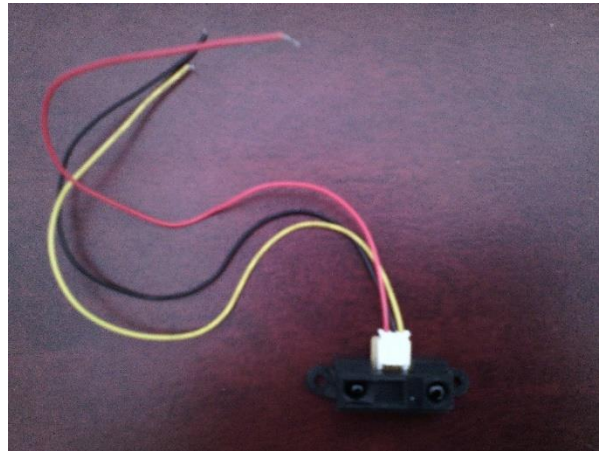


Figura 4.12 Sensor infrarrojo SHARP de distancia para medir la misma entre el robot móvil y el obstáculo.

Sensor IMU-6050

El sensor IMU (Inertial Measure Unit) contiene un acelerómetro de 3 ejes, un giroscopio de 3 ejes y un termómetro de los cuales con el uso del giroscopio y acelerómetro se puede obtener la posición del robot móvil mientras realiza su trayectoria, se ilustra en la figura 4.13 y cuyos datos técnicos se encuentran en el apéndice A.

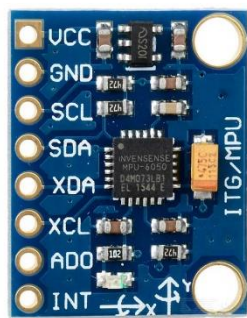


Figura 4.13 Sensor IMU-6050 para obtener la posición del robot móvil.

4.4 Comunicación

Como objetivo del anterior proyecto [12] se planteó realizar la comunicación inalámbrica con el robot móvil con una computadora para que así la computadora realice todo el proceso de control y toma de decisiones. Este método de instrumentación en robots móviles se ocupa para que el dispositivo que está leyendo las lecturas del robot móvil solo se concentre en esa tarea, convirtiéndolo en un sistema esclavo, y así solo esperar la toma de decisiones por parte de la computadora, la cual es el sistema maestro, evitando así un consumo innecesario de procesamiento por parte del dispositivo que está haciendo la lectura de los diversos sensores en el robot móvil; otra ventaja de usar un sistema inalámbrico de comunicación es que se evita el gasto de largos cables que este comunicando a los 2 sistemas y les permite operar a grandes distancias y en tiempo real.

Tarjeta de interfaz USB (Dispositivo Local)

La tarjeta de interfaz mediante cable RS-232 permite realizar la comunicación inalámbrica entre el robot móvil y la computadora se muestra en la figura 4.14 y cuyos datos técnicos se encuentran en el apéndice A.



Figura 4.14 Tarjeta de interfaz USB (XBIB-U-DEV).

La tarjeta de interfaz USB proporciona una conexión directa con dispositivos que usan protocolo USB. Además, facilita el acceso a los registros de los módulos de radiofrecuencia XBEE.

Para la adquisición de datos de los sensores se requiere de una tarjeta de adquisición de datos, el robot móvil cuenta con la tarjeta Arduino UNO, mostrada en la figura 4.15, y una tarjeta Arduino SHIELD, mostrado en la figura 4.16, la cual nos permite comunicar la tarjeta XBEE con la tarjeta Arduino uno para realizar la comunicación inalámbrica.



Figura 4.15 Tarjeta Arduino UNO.



Figura 4.16 Tarjeta SHIELD.

Capítulo 5

Implementación del control

Para implementar el control propuesto en la sección 3.1, en la plataforma del robot móvil se usa el ambiente de programación de Labview con comunicación a Arduino de forma inalámbrica por medio de los módulos de transmisión por radiofrecuencia Xbee-Pro.

Para este propósito la tarjeta Arduino uno se ocupa como una tarjeta de adquisición de datos para obtener las lecturas de los sensores y mandar las señales de activación a los motores encargados de mover al robot móvil, los modulos Xbee-Pro se encargaran de comunicar los datos enviados y recibidos por la tarjeta Arduino con la computadora que contiene el algoritmo de control y la toma de decisiones para realizar el trabajo designado.

5.1 Módulos Xbee-Pro

Los módulos Xbee-Pro son módulos de radiofrecuencia que operan en la banda de 2.4GHz con protocolo de comunicación IEEE_802.15.4. El alcance normal con antena dipolo en línea vista es de 1.6 Km al aire libre y en interiores con obstáculos de aproximadamente unos 100 metros.

La figura 5.1 muestra las conexiones mínimas que necesita el módulo Xbee-Pro para poder ser utilizado, la cual no permite el uso de Control de Flujo (RTS & CTS), por consiguiente esta opción debe estar desactivada.

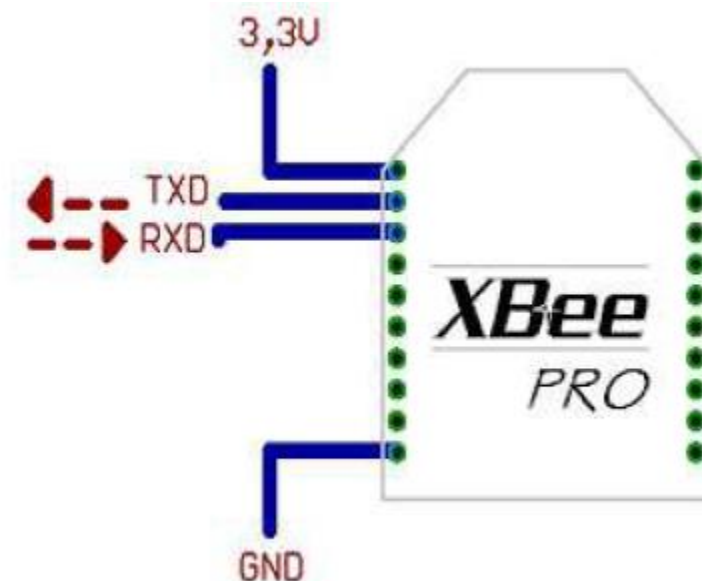


Figura 5.1. Conexiones mínimas requeridas para el módulo XBEE-PRO.

El módulo requiere una alimentación desde 2.8 a 3.4 V, la conexión a tierra y las líneas de transmisión de datos por medio del UART (TXD y RXD) para comunicarse con un microcontrolador, o directamente a un puerto serial utilizando algún conversor adecuado para los niveles de voltaje. Entre las necesidades que satisface el módulo se encuentran:

- Ultra-bajo consumo de potencia.
- Uso de bandas de radio libres y sin necesidad de licencias.
- Instalación barata y simple, redes flexibles.

Todos los dispositivos que disponen de una interfaz UART se pueden conectar directamente a los pines del módulo XBEE-PRO, como se muestra en la figura 5.2.

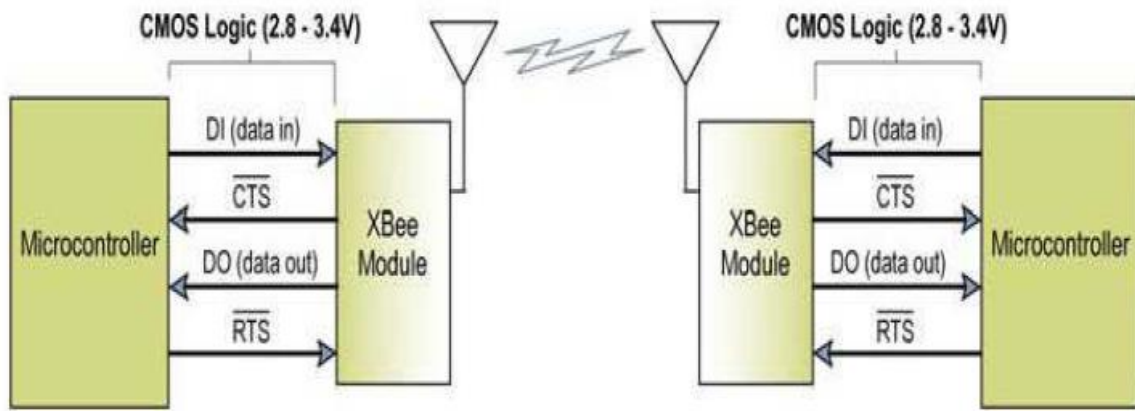


Figura 5.2 Sistema de flujo de datos mediante microcontroladores.

Vista general del sistema de comunicación inalámbrica

La interfaz utilizada para la recepción de datos de todos los sensores (potenciómetro, encoder, sensor infrarrojo e IMU-5060) y el envío de datos a los actuadores (motores) se ilustra en la figura 5.3, donde la tarjeta ARDUINO UNO funge como tarjeta de adquisición de datos, los módulos XBEE-PRO son el canal por el cual se envían y reciben los datos de forma bidireccional y finalmente la computadora es el elemento que ejecuta el algoritmo de control en lazo cerrado.

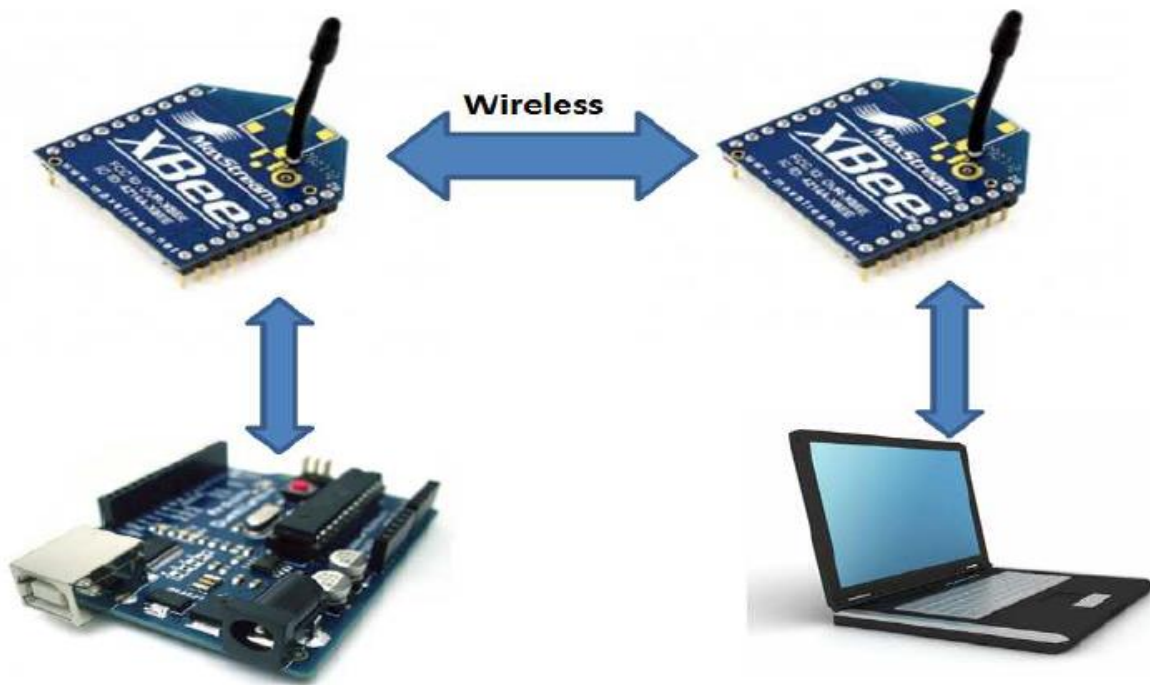


Figura 5.3 Sistema de comunicación inalámbrica.

5.2 Entorno de programación entre Labview-Arduino-Xbee

Entorno de programación Labview-Arduino

La interfaz de LABVIEW para Arduino (LIFA) Toolkit es una herramienta gratuita que se puede descargar desde el servidor de NI (National Instruments) y que permite a los usuarios de Arduino adquirir datos del microcontrolador Arduino y procesarlos en el entorno de programación gráfica de LABVIEW.

Este trabajo pretende ser una respuesta a la necesidad que existe en el ámbito académico universitario de poder disponer de un entorno de diseño y prototipado de Aplicaciones de Medida, Control y Automatización de Procesos y Espacios Físicos. La plataforma de Open Hardware Arduino ha demostrado en su corta, pero intensa vida, ser una opción muy interesante para incluir en un laboratorio de prototipado. Son muchas sus ventajas, entre las que destacamos su costo, su libre difusión y exención de costos de patentes por desarrollo, así como la gran comunidad de usuarios que se ha creado y que está generando una cantidad numerosa de aplicaciones.

El microcontrolador Arduino es una plataforma de bajo costo de electrónica de prototipos. Con la interfaz de LABVIEW para Arduino LIFA se puede aprovechar la potencia del entorno de programación gráfica de LABVIEW para interactuar con Arduino en una nueva dimensión.

Instalación del entorno de programación Lavbiew-Arduino-Xbee

Para configurar la interfaz de Labview para Arduino se realizan los siguientes pasos:

1. INSTALACIÓN DEL SOFTWARE LABVIEW 2011.

Se descarga el software LABVIEW 2011 mostrado en la figura 5.4, para implementar la interfaz requerida, utilizando las ventajas y recursos que este programa ofrece.

Se elige la versión 2011 debido a la facilidad de encontrar los toolkits gratuitos en la red que complementan las librerías requeridas para este trabajo.



Figura 5.4 Entorno de programación Labview 2011.

2. *INSTALACIÓN DEL CONTROLADOR VISA-NIMAX 5.5.*

Se instala el controlador VISA-NIMAX versión 5.5 (figura 5.5), el cual nos permite configurar los parámetros del puerto virtual utilizado en la interfaz de Labview como por ejemplo: baud rate, data bits, parity, stop bits y flow control.



Figura 5.5 Programa VISA-NIMAX versión 5.5.

3. *INSTALACIÓN DEL PAQUETE JKI VI PACKAGE MANAGER (VIPM) 2014.*

Posteriormente se instala el paquete JKI VI Package Manager (VIPM) 2014 (figura 5.6), por medio del cual se realizara la descarga e instalación de los toolkits necesarios para la implementación de la interfaz para el control del robot móvil HUMMER H2 dentro del ambiente de programación de Labview.



Figura 5.6 JKI VI Package Manager (VIPM) 2014.

4. INSTALACIÓN DEL TOOLKIT INTERFAZ DE LABVIEW PARA ARDUINO.

Ahora se descarga el toolkit interfaz LABVIEW para Arduino a través del paquete JKI VI Package Manager (VIPM) 2014 ilustrado en la figura 5.7.

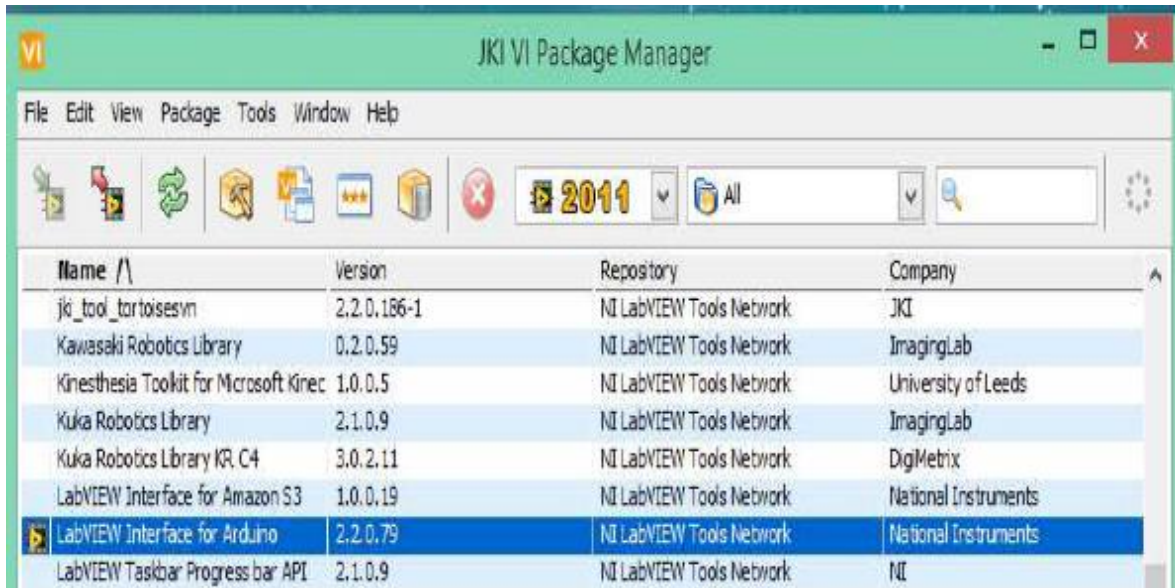


Figura 5.7 Descarga del toolkit de la interfaz entre Labview y Arduino.

5. INSTALACIÓN DEL SOFTWARE ARDUINO IDE 1.0.4.

Se realiza la instalación del entorno de Arduino IDE versión 1.0.4. (Figura 5.8), para realizar la programación de la placa Arduino a través de un puerto serial y para complementar la interfaz Arduino LABVIEW.



Figura 5.8 Arduino I.D.E. 1.0.4.

5. INSTALACIÓN DEL SOFTWARE X-CTU.

Para realizar la comunicación inalámbrica a través de los dispositivos Xbee, es necesario configurar el firmware de los mismos por medio de este software proporcionado por DIGI y visto en la figura 5.9.

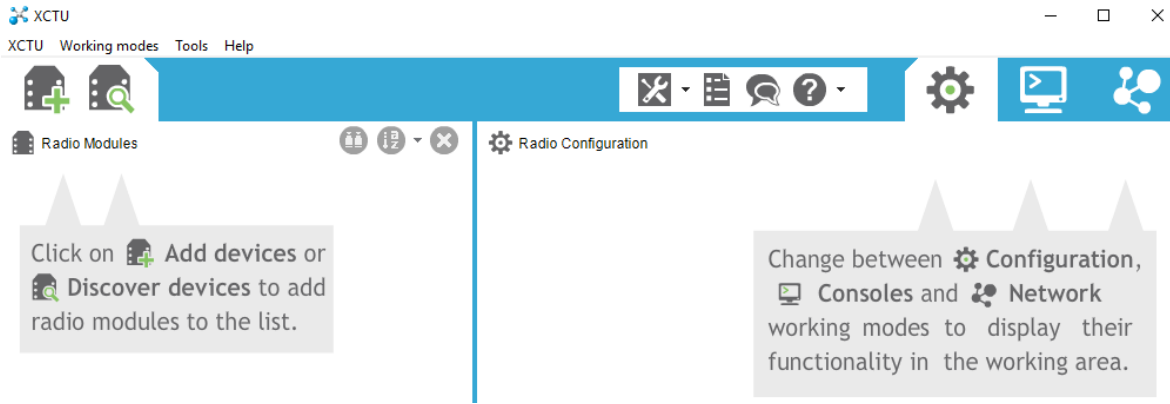


Figura 5.9 Software X-CTU.

Sincronización de la interfaz Labview-Arduino-Xbee

Una tarea fundamental para lograr el acoplamiento en la transmisión de datos de las tres plataformas LABVIEW, Arduino y XBEE, es establecer una misma velocidad de transmisión de datos y configurar los parámetros a los que se desea trabajar.

Para lograr este objetivo se requirió conocer a fondo como se realiza la comunicación entre cada una de las plataformas.

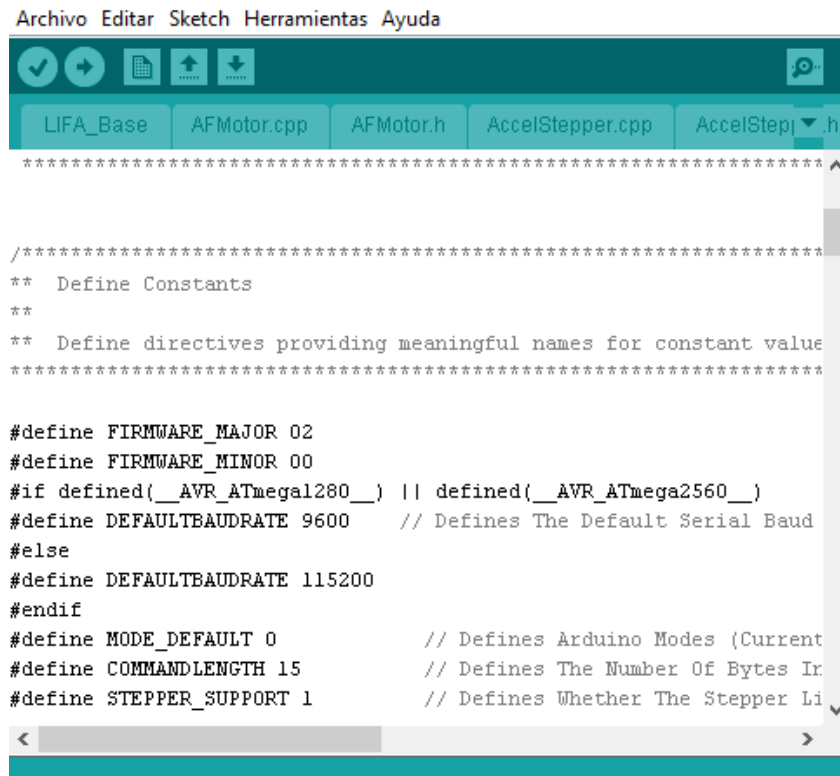
De esta manera entra a escenario el archivo LIFA_BASE, ubicado en: *C:\Program Files\National Instruments\LabVIEW2011\vi.lib\LabVIEWInterfaceforArduino\Firmware LIFA_Base* (Figura 5.10).

Este archivo contiene el firmware entre LABVIEW y Arduino, y está abierto para su libre modificación, dependiendo de la tarea que se desea realizar.

Este es el primer lugar donde se modifica la velocidad conocido como "baud rate" (bits/segundo).

También se realiza una mejora en este archivo para sincronizar correctamente la comunicación, ya que inicialmente era necesario utilizar un reset para su funcionamiento y en ocasiones la comunicación se perdía arrojando un error 5002, 5005 o 5003.

Estos errores se deben a la falta de sincronización, lo que provoca el desborde de los datos en el buffer, lo que a su vez provoca la pérdida de la transmisión.



```
*****  
  
/*****  
** Define Constants  
**  
** Define directives providing meaningful names for constant values  
*****  
  
#define FIRMWARE_MAJOR 02  
#define FIRMWARE_MINOR 00  
#if defined(__AVR_ATmega1280__) || defined(__AVR_ATmega2560__)  
#define DEFAULTBAUDRATE 9600 // Defines The Default Serial Baud  
#else  
#define DEFAULTBAUDRATE 115200  
#endif  
#define MODE_DEFAULT 0 // Defines Arduino Modes (Current  
#define COMMANDLENGTH 15 // Defines The Number Of Bytes In  
#define STEPPER_SUPPORT 1 // Defines Whether The Stepper Li
```

Figura 5.10 Archivo LIFA_Base.

El segundo lugar donde se deben establecer los parámetros deseados es en el puerto virtual a través del controlador VISA como se muestra en la figura 5.11 y también por medio del administrador de dispositivos de Windows 8.1 mostrado en la figura 5.12.

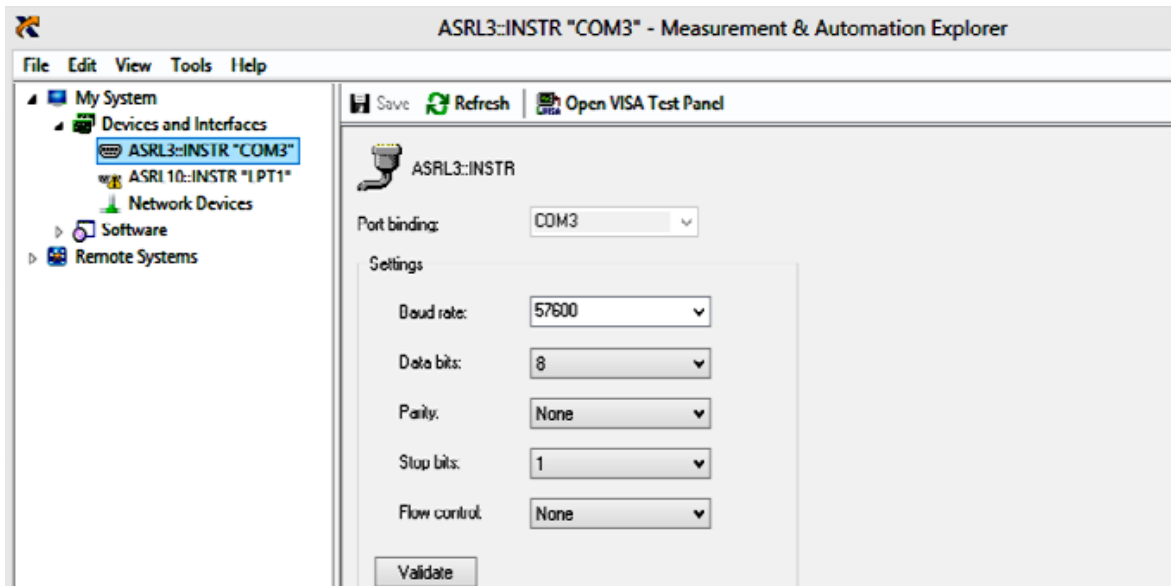


Figura 5.11. Configuración de parámetros a través del controlador VISA NI MAX.

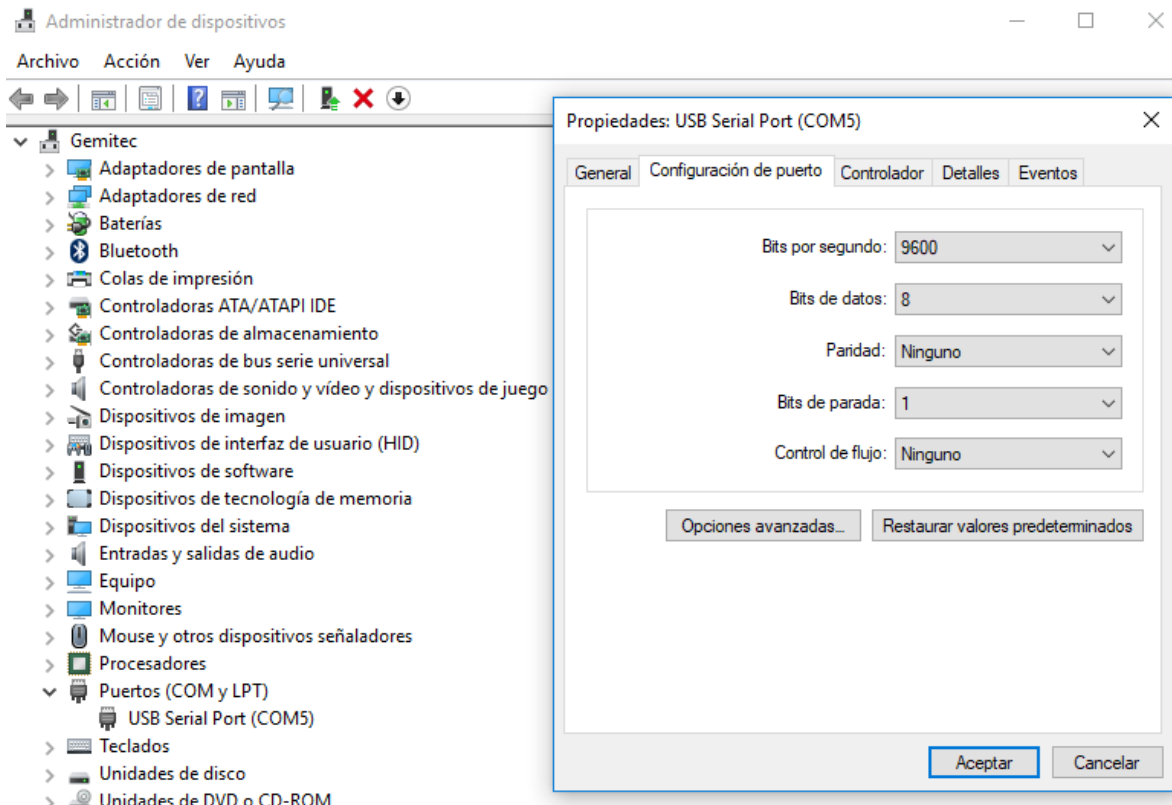


Figura 5.12. Configuración de parámetros a través del administrador de dispositivos.

Posteriormente se configuran los parámetros de transmisión en el firmware de los módulos XBEE, por medio del software XCTU descrito anteriormente.

Finalmente el último lugar donde se establecen los parámetros configurados es en el programa realizado en LABVIEW mostrado en la figura 5.13.

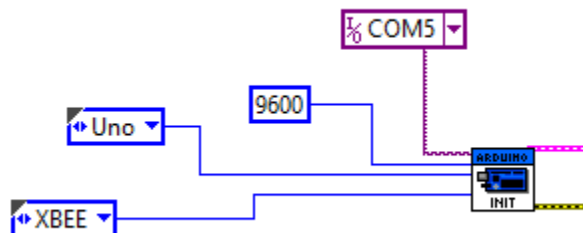


Figura 5.13 Parámetros de inicialización de Arduino en Labview.

5.3 Procesamiento digital de las señales emitidas por los sensores

Encoder

Para contar el número de pulsos se desarrolló el siguiente programa en LABVIEW (mostrado en la figura 5.15), utilizando la librería de interfaz para Arduino.

Donde se establece una velocidad de transmisión de 111111 bits/seg., enviando 15 bytes por paquete enviado, a través del puerto virtual 5, hacia una tarjeta Arduino UNO, por medio de una comunicación inalámbrica XBEE.

Todos estos datos se configuran para poder iniciar la comunicación del dispositivo local conectado a una computadora portátil hacia el dispositivo remoto conectado a la tarjeta de datos Arduino UNO.



Figura 5.14 Diagrama de flujo del encoder.

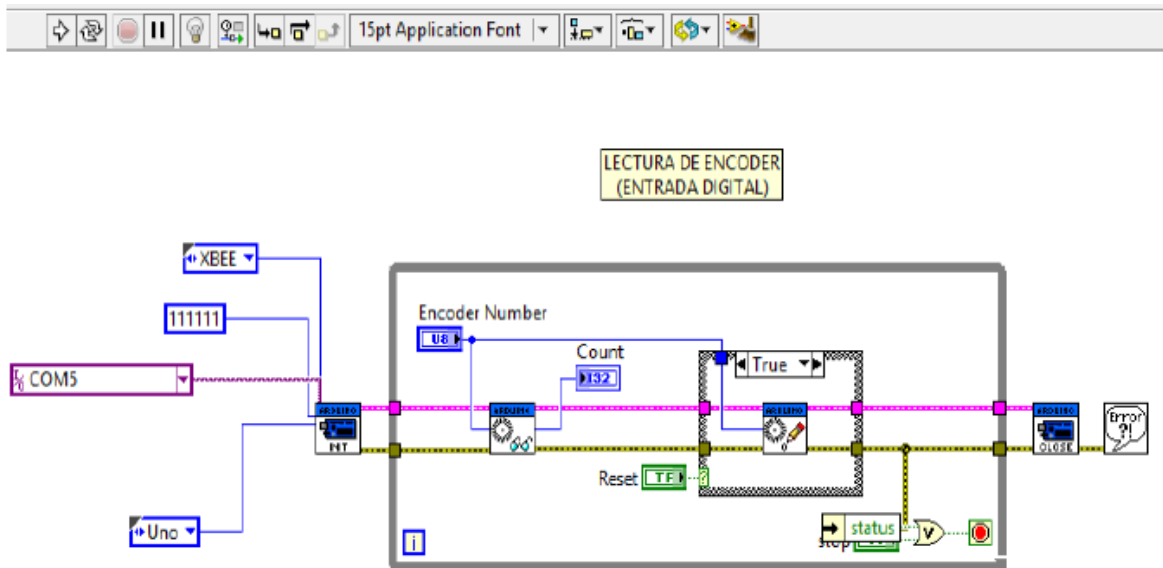


Figura 5.15 Programa para la lectura del Encoder en LABVIEW.

Para obtener la cuenta de los pulsos se utiliza el indicador "count", mostrado en la figura 5.15. Por otra parte para obtener la velocidad de los pulsos se utilizan los registros de corrimiento del software LABVIEW, los cuales nos permiten mediante una diferencia obtener el número de pulsos que se han generado por cada ejecución que realiza el ciclo while.

De esta manera cada ejecución está determinada por un temporizador cuyo valor es el tiempo de muestreo de los sensores que equivale a 50 milisegundos.

Una vez que se obtiene la posición en número de pulsos y la velocidad en número de pulsos/periodo de muestreo, se realiza la conversión de pulsos a metros, teniendo en cuenta que el perímetro de cada llanta es de 0.41 metros y que el Encoder nos proporciona una

resolución de 80 revoluciones/vuelta, la relación para obtener la unidad de la posición y velocidad en metros es 1 pulso = 0.005125 metros.

Potenciómetro

Para medir la dirección del móvil en grados se desarrolló el siguiente programa en LABVIEW (mostrado en la figura 5.17), utilizando la librería de interfaz para Arduino con el protocolo de comunicación serial.

Donde se establece una velocidad de transmisión de 111111 bits/seg., enviando 15 bytes por paquete enviado, a través del puerto virtual 5, hacia una tarjeta Arduino UNO, por medio de una comunicación inalámbrica XBEE.

Todos estos datos se configuran para poder iniciar la comunicación del dispositivo local conectado a una computadora portátil hacia el dispositivo remoto conectado a la tarjeta de datos Arduino UNO.



Figura 5.16 Diagrama de flujo del potenciómetro.

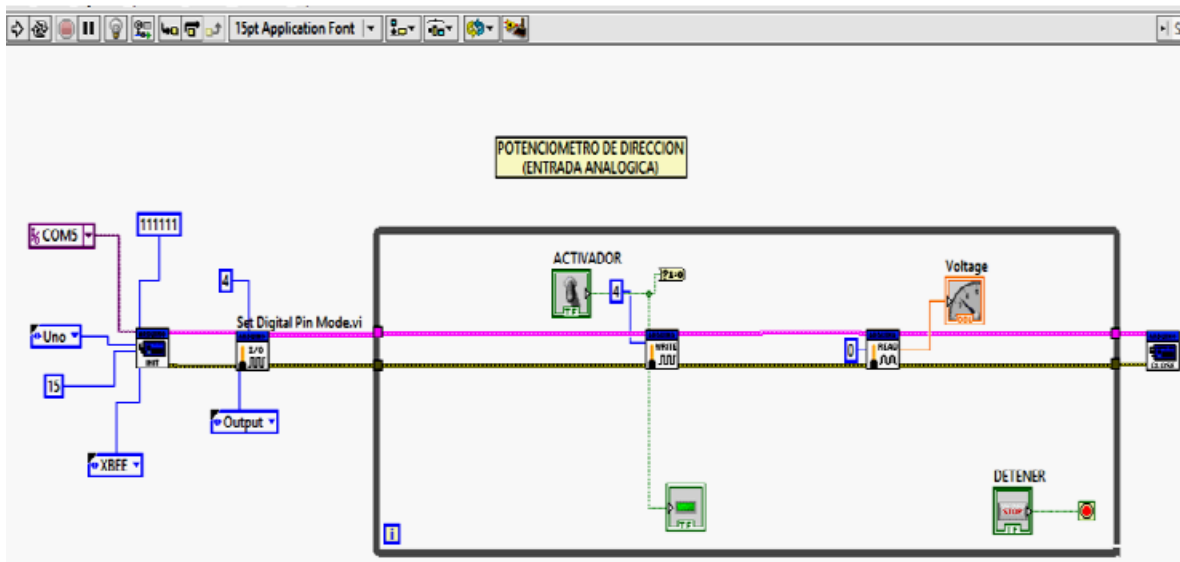


Figura 5.17. Programa para la lectura del Potenciómetro en LABVIEW.

La lectura del voltaje varía de 0.3724V a 0.7252V, con respecto a la posición del eje de las ruedas delanteras, el valor del voltaje variable se lee de forma directa, se mide el valor de voltaje cuando el eje está a centrado (0°), el cual resulta ser de 0.4753V, a partir de este valor se normaliza la escala "Voltaje – Grados", tanto para el sentido de la izquierda como para el sentido de la derecha, tomando como suposición que el ángulo máximo de giro para ambos sentidos de 30°.

Sensor infrarrojo SHARP

Para obtener los datos de la señal de salida del sensor infrarrojo se utiliza el siguiente programa en Labview ilustrado en la figura 5.19.

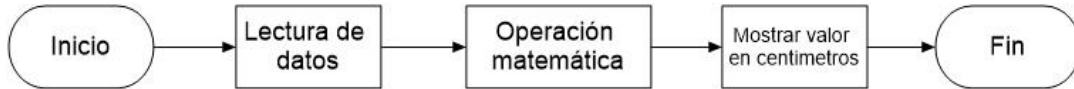


Figura 5.18 Diagrama de flujo del sensor infrarrojo SHARP.

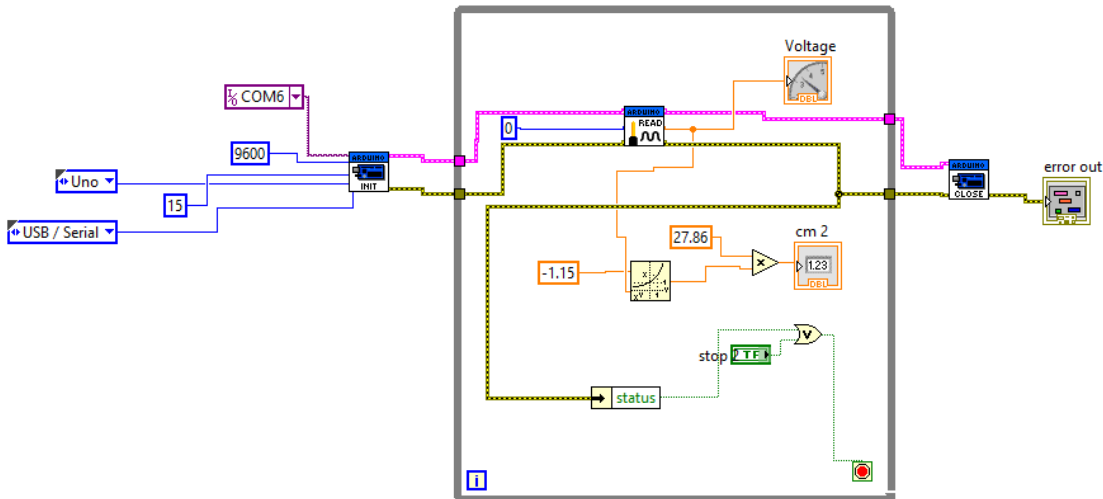


Figura 5.19 Programa para lectura del sensor infrarrojo.

En el programa se obtiene la lectura del sensor infrarrojo SHARP por el puerto analógico y después se convierte este valor a centímetros. Después de hacer la caracterización de este sensor (figura 5.20) y de realizar pruebas con el programa de Labview se determina que el sensor permite lecturas adecuadas a partir de los 9 cm.

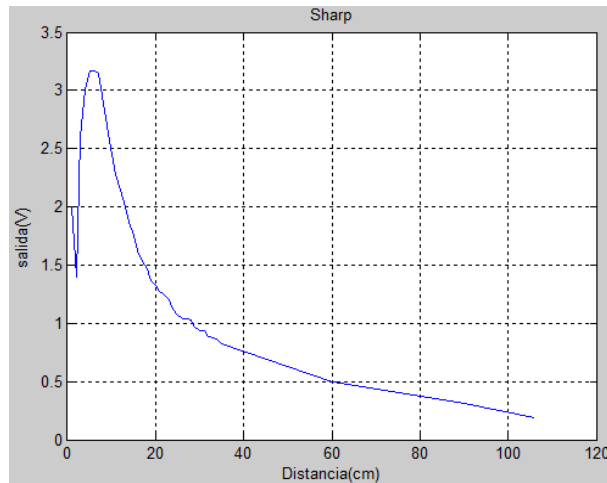


Figura 5.20 Gráfica de respuesta del sensor infrarrojo SHARP

Sensor IMU-6050

Para obtener los datos crudos del sensor IMU-6050 se ocupa el programa de Labview mostrado en la figura 5.22.

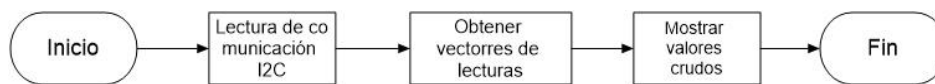


Figura 5.21 Diagrama de flujo de la lectura del sensor IMU-6050.

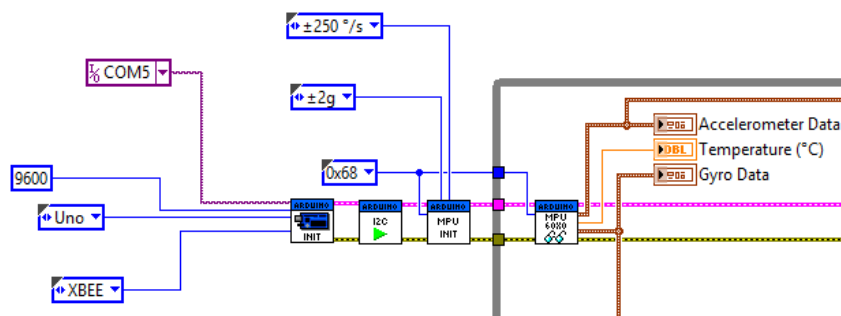


Figura 5.22 Lectura de los datos crudos del sensor IMU-6050.

En este programa se obtienen las lecturas de los datos crudos del sensor IMU, por medio del método de comunicación I²C, del acelerómetro, del giroscopio y de la temperatura para luego realizar las operaciones necesarias para convertir los datos a ángulos o posiciones dentro de un plano (x, y, z).

Ya convirtiendo los datos crudos a reales el acelerómetro da m/s^2 y el giroscopio rad/s , los cuales se tienen que integrar para obtener la posición y los ángulos del dispositivo lo cual genera un error que se va acumulando dentro de las operaciones matemáticas que una lectura errónea. Para eliminar estos errores se ocupan filtros complementarios que ayudan a obtener lecturas más fiables al disminuir los errores antes mencionados.

En la figura 5.19 se muestra a interfaz de usuario del programa de Labview para poder analizar y estudiar el funcionamiento de este sensor.

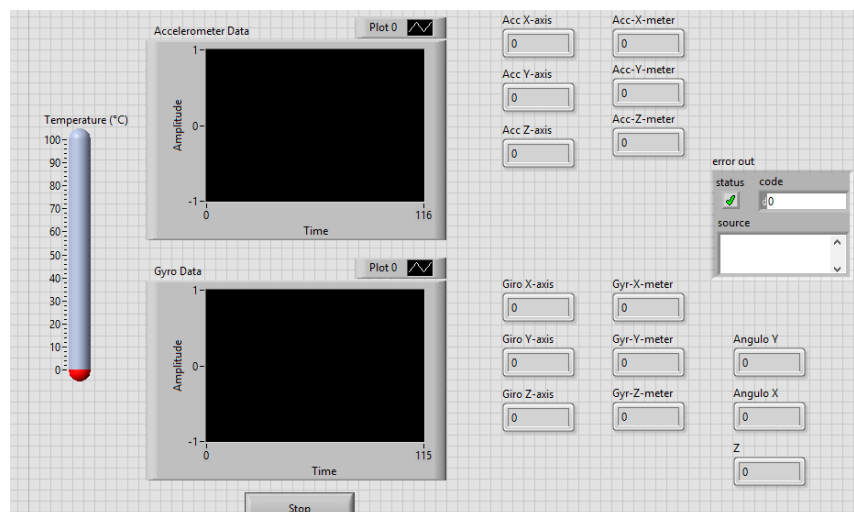


Figura 5.23 Interfaz de usuario de la adquisición de datos del sensor IMU-6050.

5.4 Procesamiento de señales digitales de control de motores

Controlador de dirección

Para controlar el motor de dirección, se determina la diferencia del ángulo deseado (θ_d) con respecto al ángulo del robot móvil (θ_r) por medio de la siguiente ecuación

$$\alpha_d = k(\theta_d - \theta_r) \quad (48)$$

donde k es una ganancia positiva constante y α_d es el ángulo de dirección deseado al que se quiere llegar. A partir de esta variable se compara con respecto a el ángulo de dirección del robot (α_r) para determinar en qué dirección debe girar el eje de dirección del robot y para determinar la fuerza se ocupa la variable F_θ para regular el PWM del motor de dirección como se muestra en la ecuación (49), la variable F_θ se explica en el capítulo 6 en la ecuación 60.

$$PWM = abs(F_\theta) \quad (49)$$

Control de tracción

Para determinar la fuerza proporcionada al motor de tracción se emplean las ecuaciones (59, 58, 61) del capítulo 6 para obtener la variable F_p para ocupar la ecuación 50.

$$PWM = abs(F_p) \quad (50)$$

5.4 Descripción general de la implementación de control

A continuación, se presenta el diagrama de flujo que describe el concepto general del control propuesto a implementar sobre el robot móvil HUMMER H2.

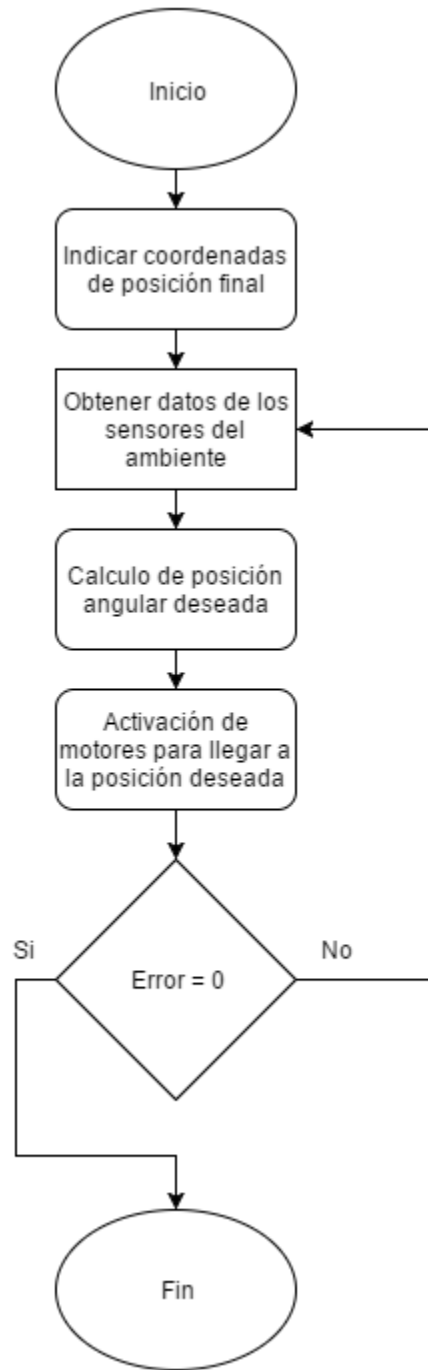


Figura 5.24 Diagrama de flujo del concepto general del control propuesto

Capítulo 6

Simulaciones

Para comprobar que el control planteado en el capítulo 3 funcione en el modelo dinámico del robot móvil descrito en el capítulo 2 se realizan simulaciones en el ambiente de programación MATLAB Simulink, los códigos empleados para las simulaciones se encuentran en el apéndice C y sigue la lógica del diagrama de flujo de la figura 6.1.

Las simulaciones se realizaron bajo un tiempo de 5 segundos con el método de integración ode45 establecido en la simulación con el objetivo de llegar a punto de coordenadas especificado estableciendo un obstáculo en el camino de la trayectoria al punto meta. Las especificaciones físicas pedidas por el modelo dinámico son las mismas que posee el robot móvil HUMMER H2, no son consideradas las dimensiones físicas del robot móvil y el obstáculo durante la simulación; no obstante, para la forma del obstáculo se toman las formas matemáticas planteadas en [11] de las cuales se escogió la forma de una curva simulando un círculo, los resultados obtenidos de la simulación se describen a continuación con sus gráficas correspondientes.

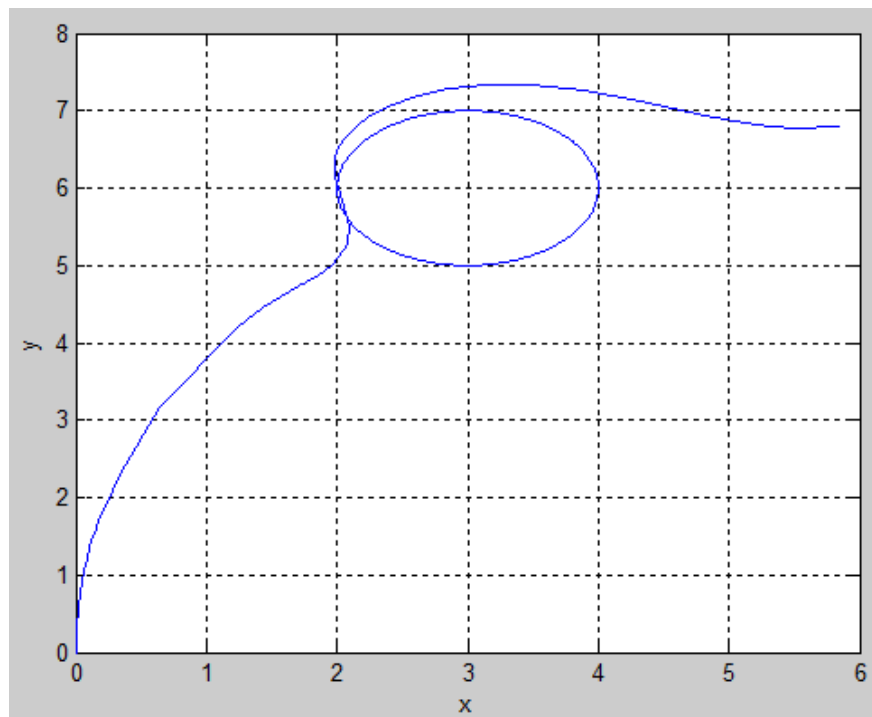


Figura 6.2 Gráfica de evasión de un obstáculo.

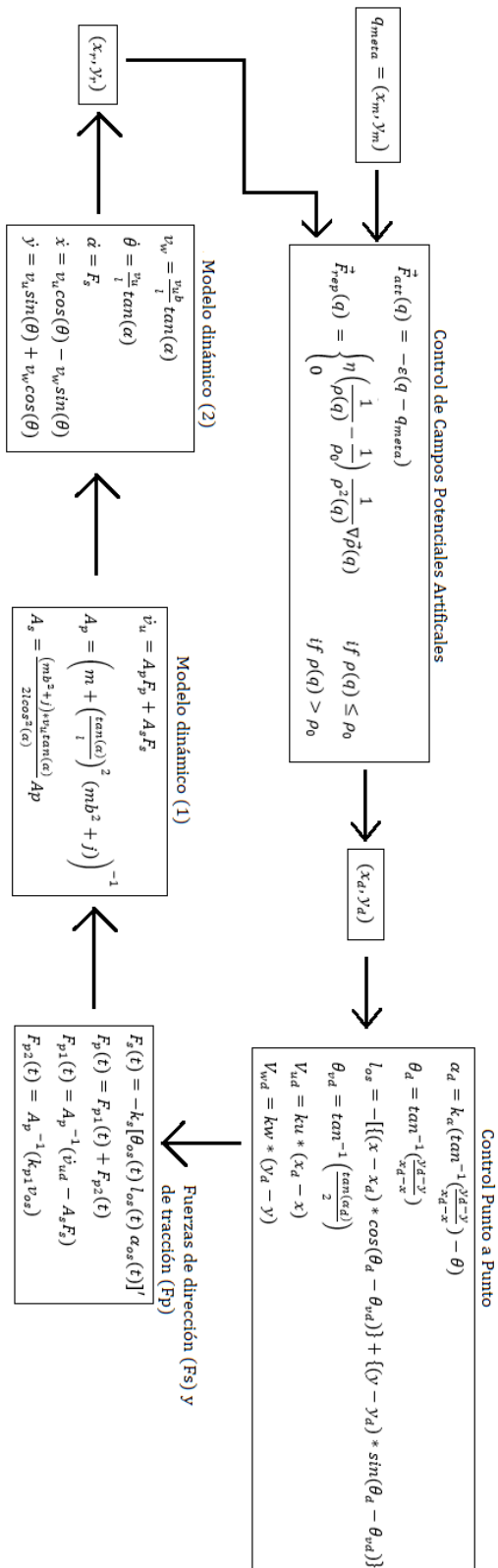


Figura 6.1 Diagrama de flujo del código de la simulación.

En la figura 6.2 se realiza la simulación del robot móvil evadiendo un obstáculo localizado en la coordenada $x=3$ y la coordenada $y=6$ con punto de destino las coordenadas $x=6$ y $y=7$, en la gráfica se ilustra como el robot móvil detecta el obstáculo y empieza a realizar las maniobras de evasión pertinentes y una vez que no lo detecta continua con la trayectoria hacia el punto meta.

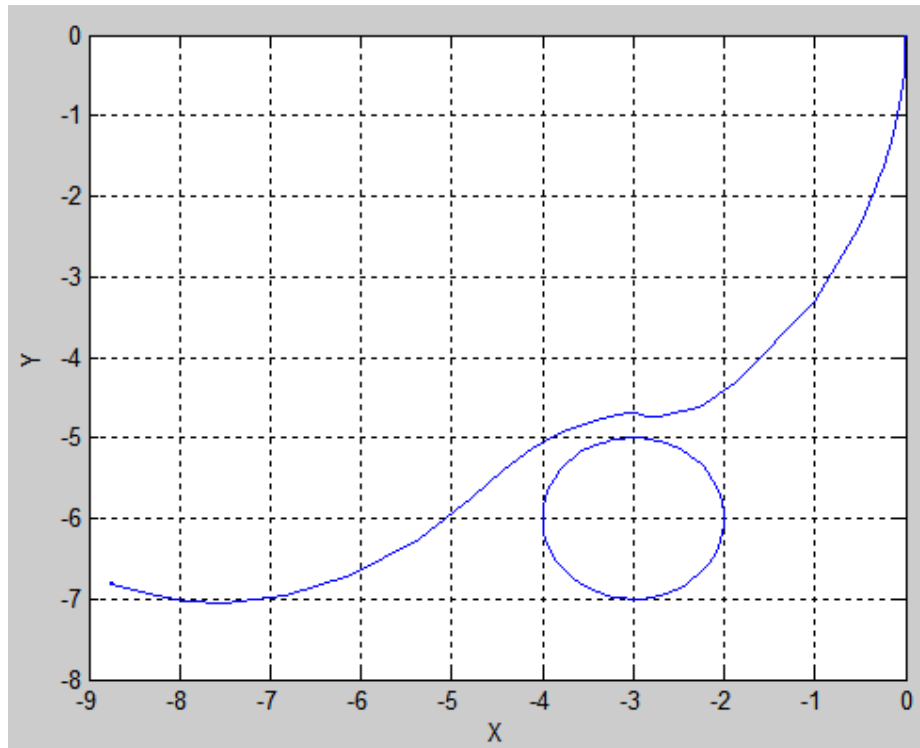


Figura 6.3 Gráfica de evasión de un obstáculo en coordenadas $(-x, -y)$.

Para prueba de la eficiencia del control se propuso cambiar las coordenadas de meta a llegar y del obstáculo a $(-9, -7)$ y $(-3, -6)$ respectivamente. Como lo ilustra la gráfica de la figura 6.3 el control en esta situación fue más suave al no realizar una vuelta tan amplia como en la figura 6.2.

En las figuras 6.4 y 6.5 se hace la simulación con las mismas especificaciones pero cambiando en ellas las coordenadas por $(-x, y)$ y $(x, -y)$ respectivamente.

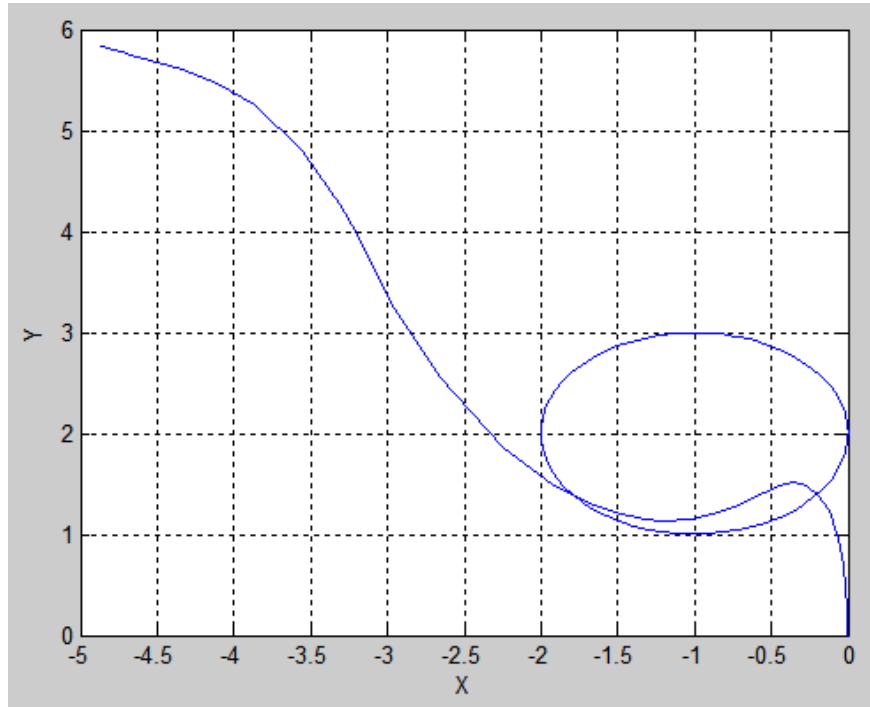


Figura 6.4 Gráfica de evasión de un obstáculo en coordenadas $(-x, y)$.

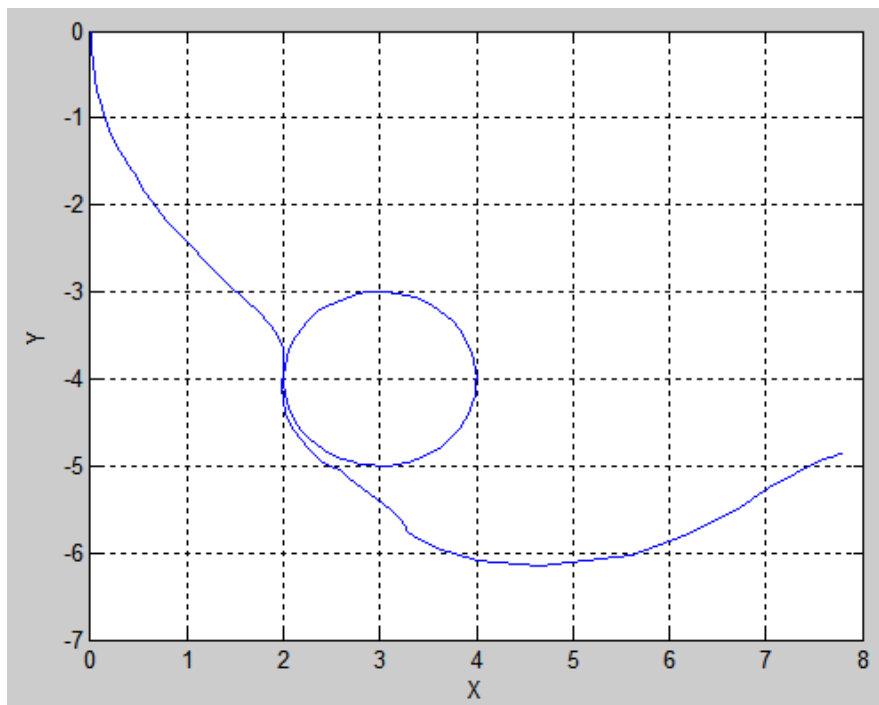


Figura 6.5 Gráfica de evasión de un obstáculo en coordenadas $(x, -y)$.

En las figuras 6.4 y 6.5 el control vuelve a mostrar una trayectoria suave evitando el obstáculo y llegando al punto de meta, para la figura 6.4 el punto meta está localizado en $(-5,6)$ y el obstáculo en $(-1,2)$ y para la figura 6.5 el punto meta en $(8,-5)$ y el obstáculo en $(3,-4)$.

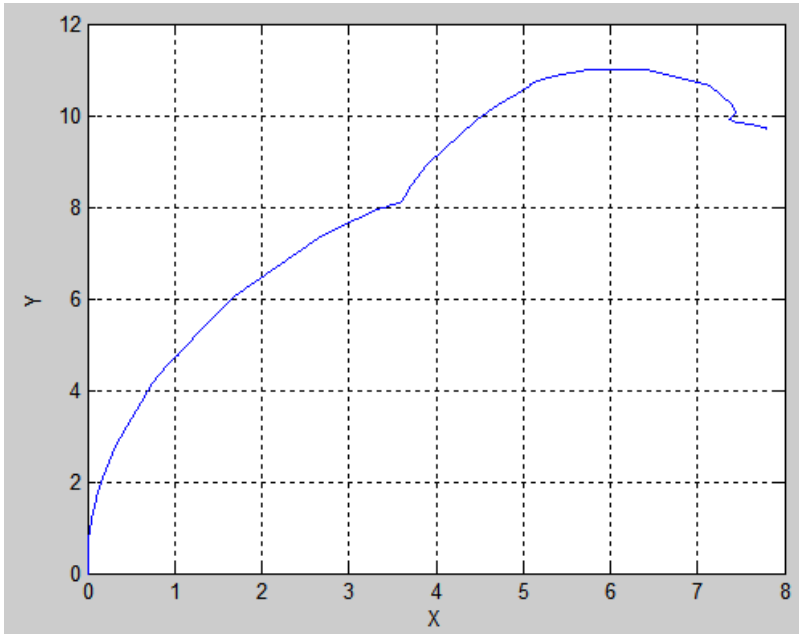


Figura 6.6 Gráfica de trayectoria original.

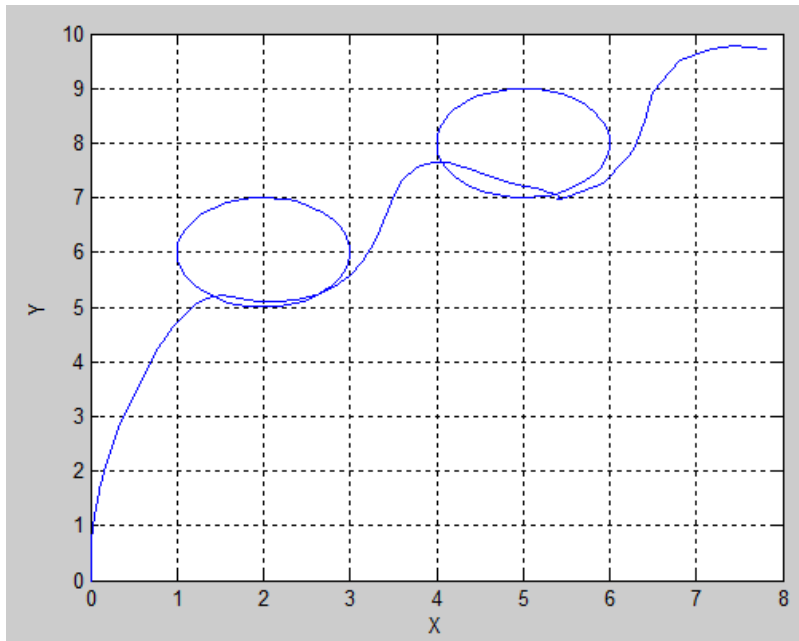


Figura 6.7 Gráfica de evasión con dos obstáculos.

En la figura 6.6 se realiza la trayectoria al punto meta $(7,9)$ sin tener obstáculo presente, se realiza esta simulación para poder comparar la trayectoria que realiza el robot móvil sin obstáculos contra la trayectoria que realiza estableciendo dos obstáculos en medio de la trayectoria original, la cual se ilustra en la figura 6.7. En la figura 6.7 se muestra que al detectar el primer obstáculo en $(1,5)$ el robot móvil empieza a realizar la evasión del mismo dirigiéndose al punto meta $(7,9)$ y en cuanto detecta el obstáculo en $(4,8)$ realiza una ligera desviación para evitarlo, porque no se encontraba en colisión directa con el obstáculo, llegando al punto meta mencionado anteriormente sin problema alguno.

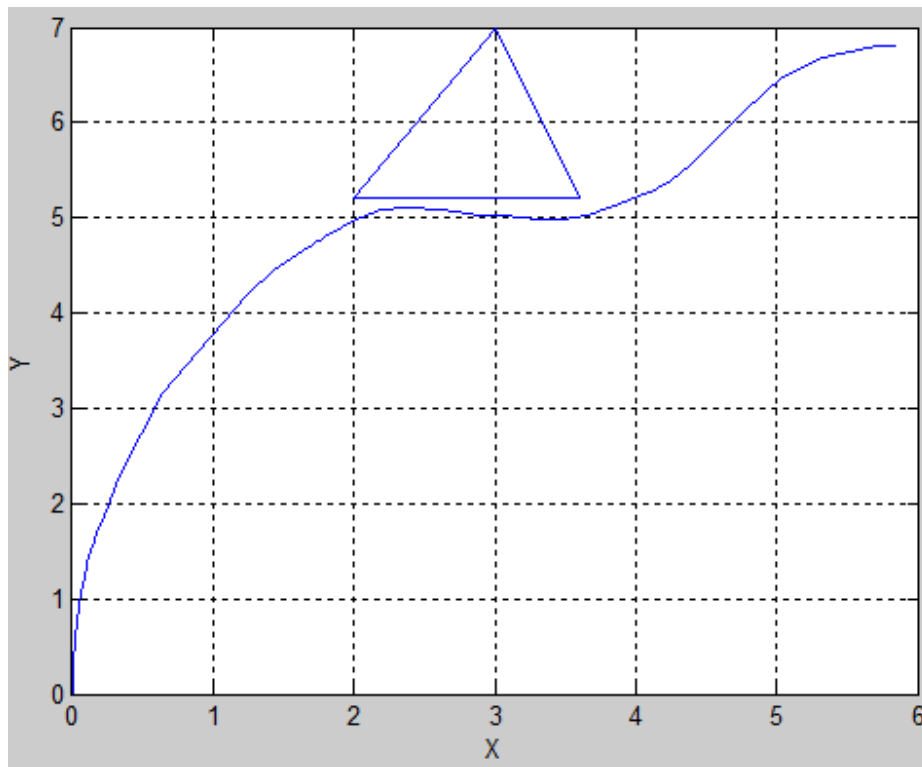


Figura 6.8 Gráfica de evasión de obstáculo con fórmula de obstáculo diferente.

En la figura 6.8 se realiza la simulación con las características de la figura 6.2 cambiando la fórmula que describe la forma del obstáculo. En la figura 6.2 el obstáculo tiene una forma curva emulando un cilindro, mientras que en la simulación de la figura 6.7 se plantea que el robot móvil se está encontrando con una esquina del obstáculo, como la punta de un triángulo, realizando una trayectoria distinta a la simulación original cumpliendo con el objetivo de llegar al punto meta designado.

Con las simulaciones realizadas se comprueba que el método de control propuesto es eficiente y se procede a realizar el montaje de los sensores y pruebas pertinentes en el robot móvil para la evasión de obstáculos.

Pruebas experimentales

Se empiezan a realizar las pruebas comprobando que la plataforma HUMMER H2 se encuentre en buen estado revisando el control sobre los motores, durante estas pruebas se comprueba que al brindarle energía al sistema sin arrancar el programa de Labview provoca que el motor de dirección se active y continúe hasta el tope mecánico del eje de dirección. Para prevenir esto se coloca un switch, el cual se ilustra en la figura 6.9, entre la fuente de energía y el puente H que activa los motores para evitar la activación de los motores antes de que inicie el control sobre el robot.



Figura 6.9 Switch para evitar el paso de energía.

Después de inspeccionar el motor que estaba acoplado en el eje de dirección después del incidente, se localizó un daño irreparable en los dientes del engranaje que permitían al motor mover el eje de dirección, con lo que se procedió a conseguir uno nuevo.

Con el motor nuevo se realizaron más pruebas identificando otro problema mecánico sobre el eje de dirección, el cual era que al llegar al máximo ángulo permitido en el eje de dirección, tanto en el derecho como en el izquierdo, el motor no tenía el suficiente torque para poder girar las llantas en el sentido contrario; se procede a colocar unos switch de fin de carrera en los costados del robot móvil con el motivo de indicar al control que deje de suministrarle energía a los motores para evitar estas situaciones no deseadas.

Con el control manual sobre los motores comprobado; se prosigue a montar los sensores de distancia infrarrojos en el robot móvil, para ello se diseña una plataforma hecha con dos placas de acrílico de forma cuadrada, ilustrada en la figura 6.10, para poder colocar el marco donde se montarán los sensores de forma distribuida para detectar los obstáculos, el marco se ilustra en la figura 6.11.

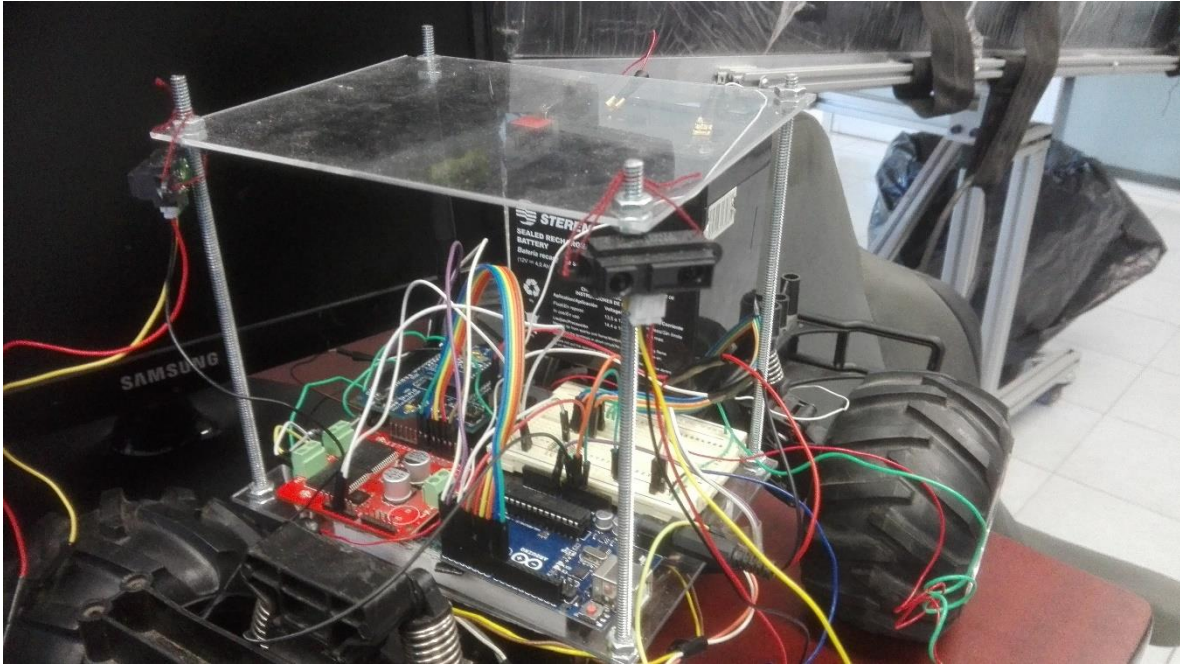


Figura 6.10 Plataforma creada con las dos placas de acrílico.

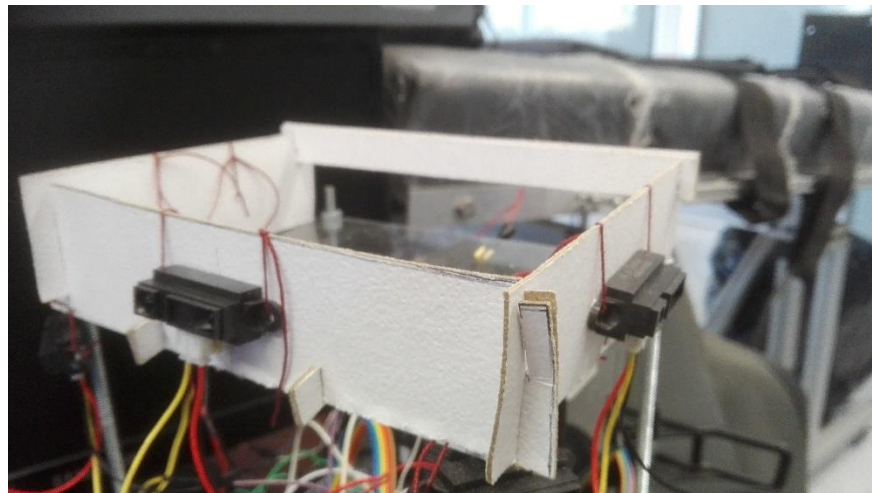


Figura 6.11 Marco de soporte para los sensores infrarrojos.

Después de realizar los montajes se realizan pruebas en el IMU-6050 para obtener el ángulo del robot móvil con respecto al plano horizontal del ambiente de trabajo en el cual trabajará el robot a través del giroscopio del IMU-6050, para ello se plantean las ecuaciones matemáticas pertinentes en Labview mostradas en la figura 6.12.

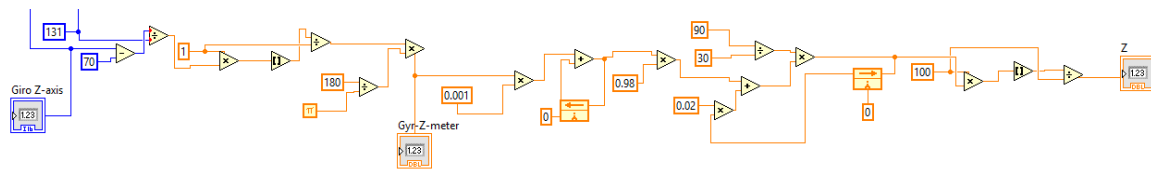


Figura 6.12 Programa de Labview para determinar el ángulo del robot móvil con respecto al plano horizontal del ambiente de trabajo.

En el programa mostrado en la figura 6.12 se aplica la conversión de los datos obtenidos del IMU-6050 a datos reales, posteriormente se hace el cambio de radianes a grados para poder así aplicar el método de integración de Euler. Una vez que se obtiene el dato de la integración se aplica un filtro para reducir el error generado de la operación de integración y un redondeo a dos décimas para tener una mejor visualización de la lectura tomada.

Con los sensores preparados se realiza el diseño y fabricación de los obstáculos, que en este caso se decide por tres obstáculos de color blanco donde uno es de forma circular, el segundo de forma hexagonal y el tercero es un obstáculo conformado de caras planas y caras curvas, los obstáculos se ilustran en las figuras 6.13, 6.14 y 6.15 respectivamente.

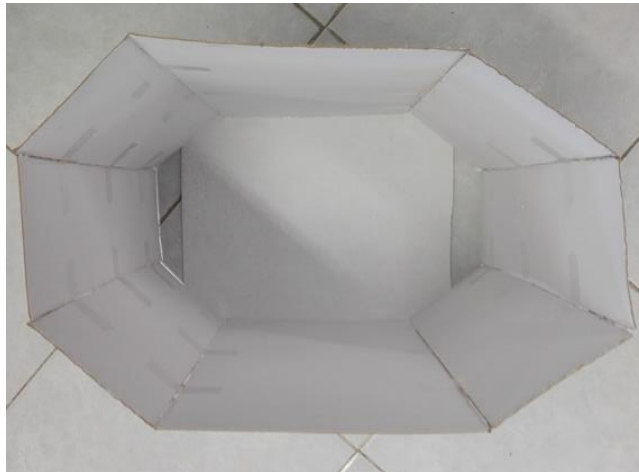


Figura 6.13 Obstáculo con forma hexagonal de caras planas.



Figura 6.14. Obstáculo con forma circular



Figura 6.15 Obstáculo con forma hexagonal de caras curvas.

Control Punto a Punto para llegar a un punto deseado

Se diseña un control punto a punto para que el robot móvil llegue a un punto meta deseado en coordenadas (x, y) . Para realizar este experimento se propone ocupar las losas del piso del laboratorio de la zona de trabajo como unidad de medida, se ilustra en la figura 6.16.

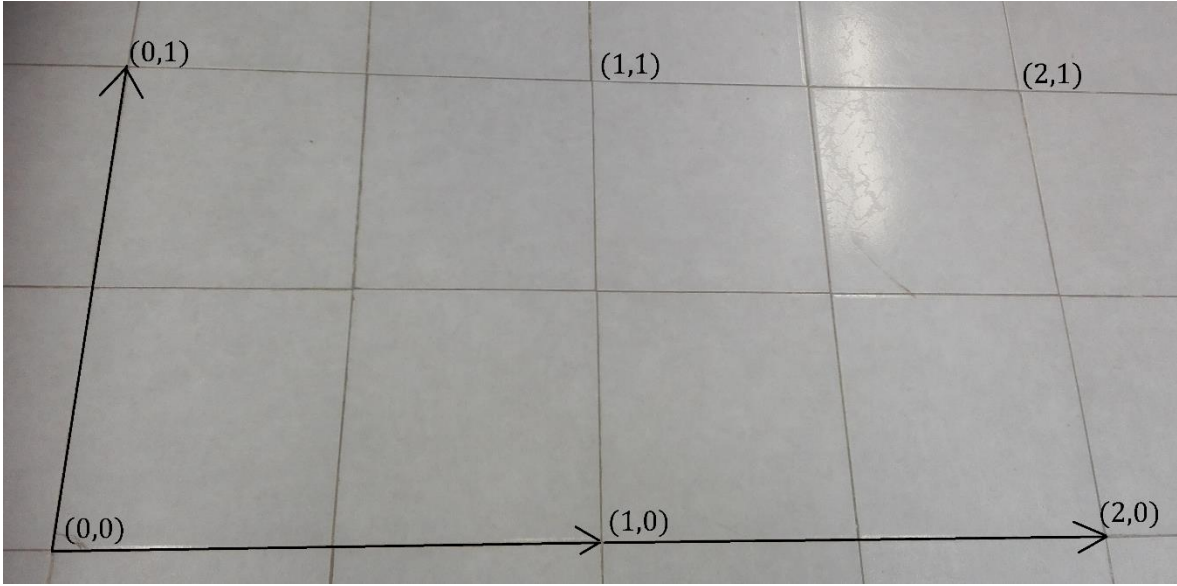


Figura 6.16. Unidad de medida ejemplificada en las losas para el control propuesto.

Con la unidad base establecida para el mapa de coordenadas (x,y) se procede a implementar el siguiente algoritmo de control basado en la tangente hiperbólica con entradas de control los errores de posición en los ejes de coordenadas x, y y sus respectivas derivadas.

$$\varepsilon_x = x_d - x_r \quad (51) \quad \varepsilon_y = y_d - y_r \quad (52)$$

$$\theta_d = \text{atan2}(\varepsilon_y, \varepsilon_x) \quad (53) \quad \varepsilon_\theta = \theta_d - \theta_r \quad (54)$$

$$\dot{x} = \frac{x_{rk} - x_{rk-1}}{h} \quad (55) \quad \dot{y} = \frac{y_{rk} - y_{rk-1}}{h} \quad (56)$$

$$\dot{\theta} = \frac{\theta_{rk} - \theta_{rk-1}}{h} \quad (57)$$

$$F_x = k_{p1} \tanh(\varepsilon_x) - k_{v1} \tanh(\dot{x}) \quad (58)$$

$$F_y = k_{p2} \tanh(\varepsilon_y) - k_{v2} \tanh(\dot{y}) \quad (59)$$

$$F_\theta = k_{p3} \tanh(\varepsilon_\theta) - k_{v2} \tanh(\dot{\theta}) \quad (60)$$

$$F_p = \|F\| = \sqrt{F_x^2 + F_y^2} \quad (61)$$

donde

x_r, y_r, θ_r	Son las coordenadas x, y, θ del robot
x_d, y_d, θ_d	Son las coordenadas x, y, θ deseadas del robot
$\dot{x}, \dot{y}, \dot{\theta}$	Son las derivadas de las coordenadas x, y, θ del robot
$\ F\ $	Nos indica la fuerza con la que se mueve el motor de tracción
F_θ	Nos indica la fuerza con la que se mueve el motor de dirección

Tabla 6.1 Ecuaciones de control del robot móvil.

Las ecuaciones 6.1 a 6.11 son implementadas en el ambiente de programación de Labview con la interfaz de usuario mostrada en la figura 6.17, donde en el bloque “File name” se asigna el nombre del archivo donde se guardan los puntos por los que pasa el robot móvil para después graficar la trayectoria que realizó el mismo en el plano de coordenadas (x,y) el cual se visualiza en la figura 6.18.

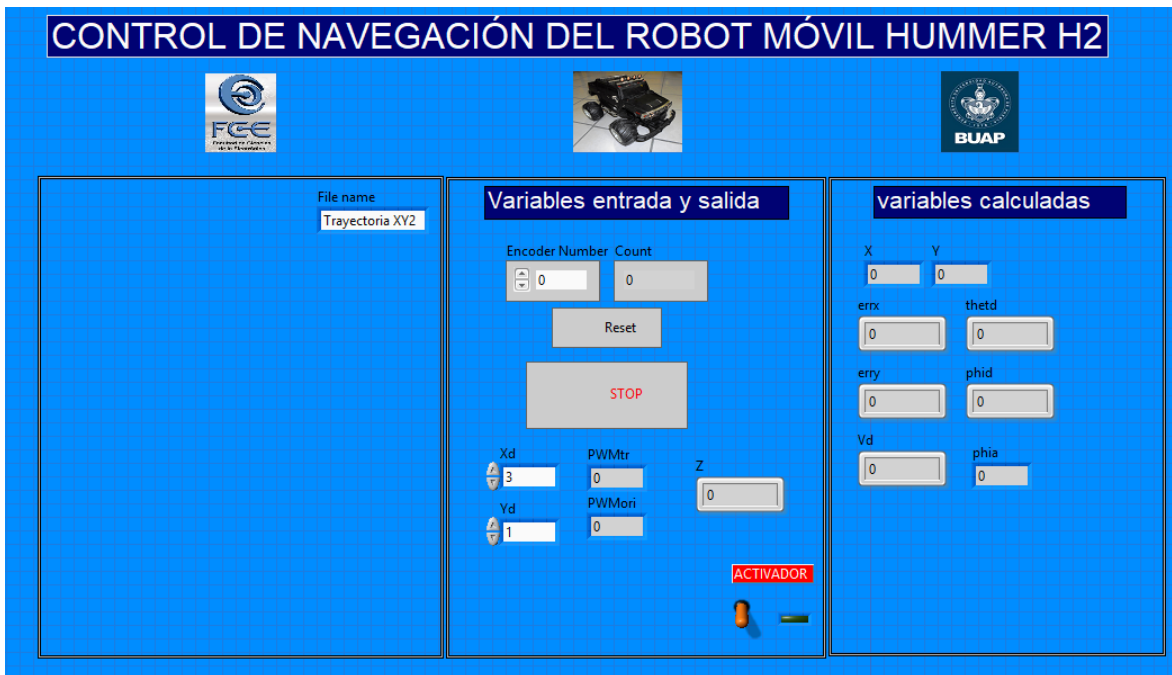


Figura 6.17 Interfaz de usuario en Labview para el control punto a punto implementado.

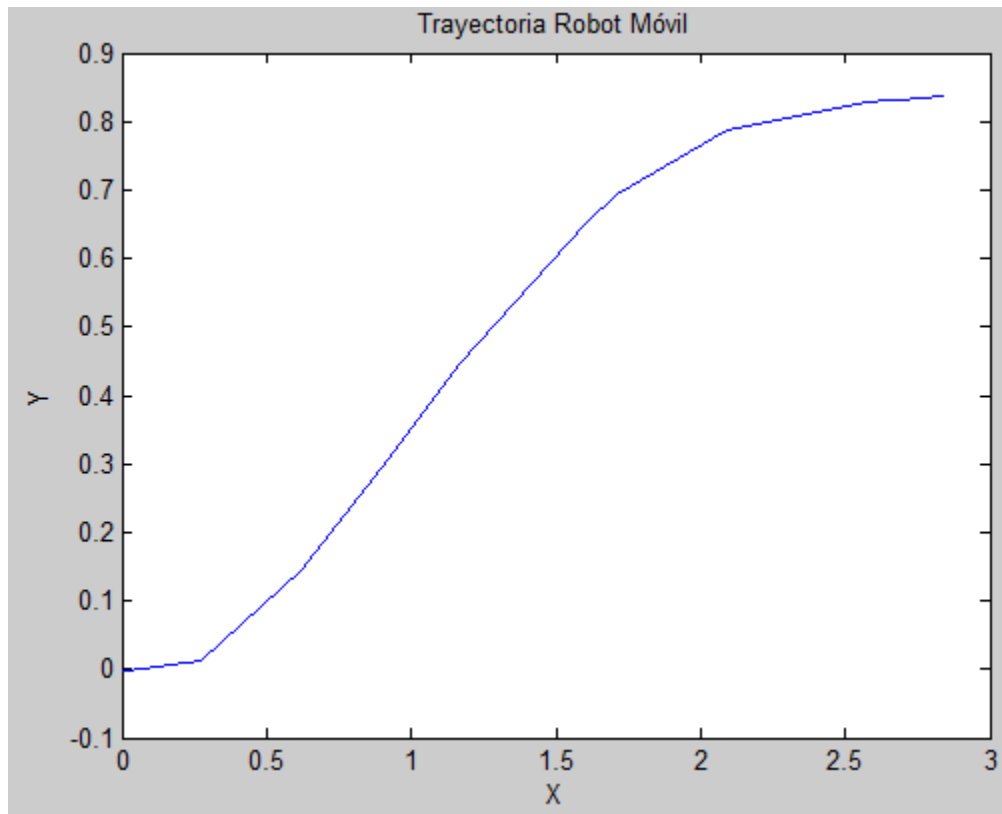


Figura 6.18. Trayectoria Realizada por el Robot Móvil.

En esta gráfica se toman las coordenadas (x, y) en intervalos de tiempo de 1 segundo por lo que se gráfica una trayectoria de cierra, cabe recalcar que la trayectoria realizada por el robot no es tan errática, para visualizar la trayectoria que el robot realiza se utiliza el método de mínimos cuadrados de cuarto orden proporcionado por las herramientas de Matlab mostrando la trayectoria que realizo el robot móvil en color verde.

Conclusiones

El método de control de campos potenciales artificiales es muy eficiente para realizar tareas de evasiones de obstáculos cuando la forma geométrica de los obstáculos no es muy compleja, ya que en algunos casos el control no puede decidir la dirección que debe seguir por la forma de los obstáculos.

La simulación del control de campos potenciales sobre el modelo dinámico del robot móvil fue de gran ayuda ya que permitió un mejor entendimiento de cómo funciona el control para así poder aplicarlo a la plataforma HUMMER H2 de forma eficiente.

Los motores de la plataforma HUMMER H2, en especial el encargado de darle tracción, deben ser reemplazados por el desgaste que llevan por su uso en los 2 trabajos en que se usaron.

La lectura de múltiples sensores en la plataforma Arduino genera la acumulación de retrasos de lectura en el ambiente de programación Labview, por lo que para este tipo de proyectos es recomendable programar directamente sobre el Arduino el control o cambiar la tarjeta de adquisición de datos, en este caso Arduino, por una que ayude a obtener lecturas más precisas y confiables.

Por cuestión de tiempo y dificultades técnicas que ocurrieron en el transcurso del desarrollo del proyecto no se logró concretar la prueba de evasión de obstáculos dentro del laboratorio de robótica de la maestría en ciencias de la electrónica opción automatización, aunque gracias a las simulaciones hechas se concreta que el experimento se concretaría de forma satisfactoria.

Apéndices

Apéndice A Manual HUMMER H2

® RadioShack
2003 Hummer H2
60-4358

What's Included	
Hummer H2	Transmitter
Plastic Antenna Tube	Box Wrench
	Owner's Manual

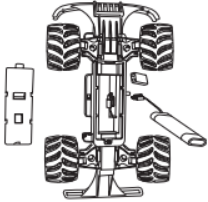
Step One

Installing the Batteries

You will need...
 One 9V Battery for the Transmitter
 One 9V Battery for the Hummer H2
 One 7.2V Rechargeable Battery Pack

Install Batteries in Hummer H2

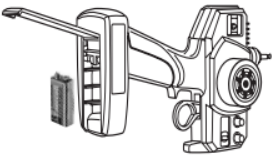
- Turn off the Hummer H2.
- Remove the battery compartment cover.
- Place the 9V battery in the battery compartment, matching the polarity symbols (+ and -) as marked inside.
- Place the battery pack in the recessed area with the battery pack's connector on the same side as the Hummer H2's matching connector.
- Attach the battery pack's connector to the Hummer H2's matching connector. They fit together only one way. Do not force them.
- Close the cover.



You must charge the battery pack with a battery charger (not supplied), available at your local RadioShack store.

Install Battery in the Transmitter

- Turn the transmitter off.
- Slide the battery compartment cover in the direction of the arrow to open it.
- Place the 9V battery in the compartment, matching the polarity symbols (+ and -) as marked inside.
- Replace the cover and snap it shut.



Step Two

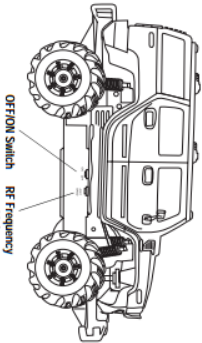
Installing the Antenna Tube

- Route the antenna wire's loose end through the supplied antenna tube.
- Insert the antenna tube securely into the Hummer H2's antenna tube hole at the top of the vehicle.

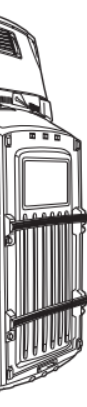
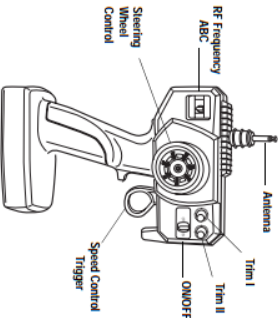
Step Three

Driving the Hummer H2

- Extend the transmitter's antenna to its full length for best reception.
- Turn on the transmitter and the Hummer H2. The headlights and the LED indicator illuminate.
- Select the specific frequency by setting the Band ABC switch on the transmitter and the Hummer H2 to the same band.
- Use the transmitter's speed control trigger to move forward and reverse. Pull the trigger toward the handle to move forward. The headlights illuminate. Pull all the way back for maximum speed. Push the trigger away from the handle to move backward. The tail lights illuminate. Release to stop.



- Use the transmitter's steering wheel control to turn the Hummer H2 left and right. Release the steering wheel to drive straight.
- Turn the transmitter and the vehicle off when you finish playing with it.



Use the box wrench to remove and/or replace the Hummer H2's wheels.

Step Four

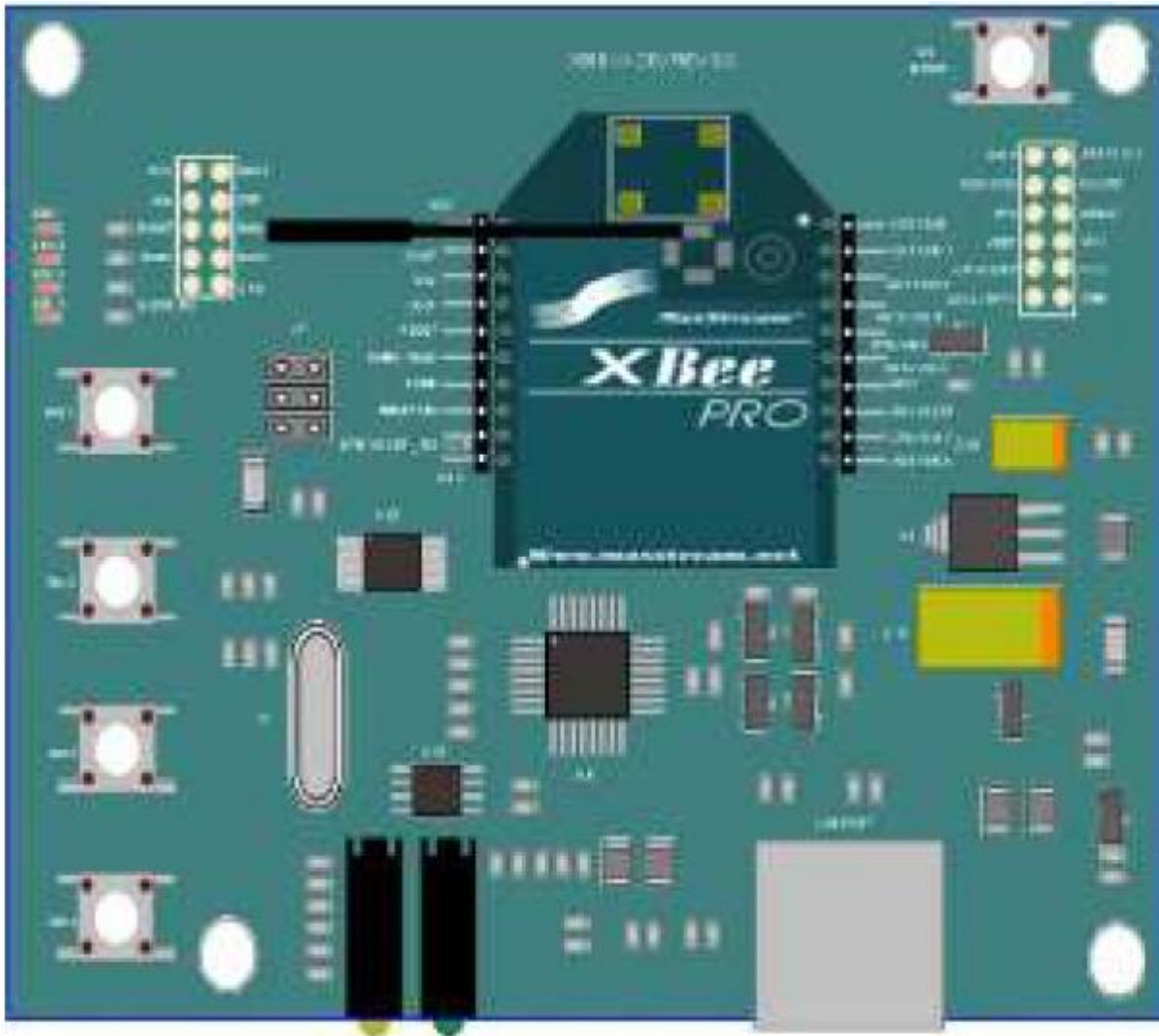
Adjusting the Trim Controls

Trim I
 The transmitter's neutral trim control lets you fine-tune the speed control so the Hummer H2 stops completely when you release the speed control to its center position. If Trim I is set too far to the left, the vehicle will move backward. If it is set too far to the right, the vehicle may continue forward.

Trim II
 The transmitter's steering trim control lets you fine-tune the steering so the Hummer H2 drives in a straight line. If it is set too far to the left or right, the vehicle may drive in those directions.

- Turn the Hummer H2 over on its side so the tires do not touch the ground.
 - Turn on the transmitter and the Hummer H2.
 - If the tires move, turn **Trim I** until the tires stop.
 - If the front tires appear to angle left or right, adjust **Trim II** until the tires appear to be straight.
 - When you finish your adjustments, turn off the transmitter and the Hummer H2.
- Do not touch the tires or the steering control during this adjustment. You may have to fine-tune the trim controls after using the Hummer H2 for a while.
- Never set **Trim I** to make the Hummer H2 continually move.

Apéndice B Componentes para instrumentar el robot móvil



Arquitectura de la tarjeta de interfaz XBIB-U-DEV

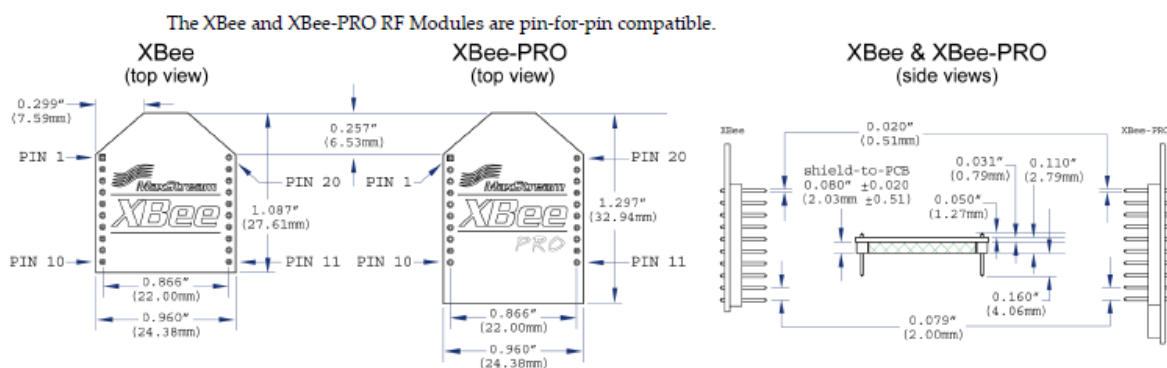
Especificaciones Modulo XBEE

Specification	XBee	XBee-PRO
Performance		
Indoor/Urban Range	Up to 100 ft (30 m)	Up to 300 ft. (90 m), up to 200 ft (60 m) International variant
Outdoor RF line-of-sight Range	Up to 300 ft (90 m)	Up to 1 mile (1600 m), up to 2500 ft (750 m) international variant
Transmit Power Output (software selectable)	1mW (0 dBm)	63mW (18dBm)* 10mW (10 dBm) for International variant
RF Data Rate	250,000 bps	250,000 bps
Serial Interface Data Rate (software selectable)	1200 bps - 250 kbps (non-standard baud rates also supported)	1200 bps - 250 kbps (non-standard baud rates also supported)
Receiver Sensitivity	-92 dBm (1% packet error rate)	-100 dBm (1% packet error rate)
Power Requirements		
Supply Voltage	2.8 – 3.4 V	2.8 – 3.4 V
Transmit Current (typical)	45mA (@ 3.3 V)	250mA (@3.3 V) (150mA for international variant) RPSMA module only: 340mA (@3.3 V) (180mA for international variant)
Idle / Receive Current (typical)	50mA (@ 3.3 V)	55mA (@ 3.3 V)
Power-down Current	< 10 μ A	< 10 μ A
General		
Operating Frequency	ISM 2.4 GHz	ISM 2.4 GHz
Dimensions	0.960" x 1.087" (2.438cm x 2.761cm)	0.960" x 1.297" (2.438cm x 3.294cm)
Operating Temperature	-40 to 85° C (industrial)	-40 to 85° C (industrial)
Antenna Options	Integrated Whip, Chip or U.FL Connector, RPSMA Connector	Integrated Whip, Chip or U.FL Connector, RPSMA Connector
Networking & Security		
Supported Network Topologies	Point-to-point, Point-to-multipoint & Peer-to-peer	
Number of Channels (software selectable)	16 Direct Sequence Channels	12 Direct Sequence Channels
Addressing Options	PAN ID, Channel and Addresses	PAN ID, Channel and Addresses
Agency Approvals		
United States (FCC Part 15.247)	OUR-XBEE	OUR-XBEEPRO
Industry Canada (IC)	4214A XBEE	4214A XBEEPRO
Europe (CE)	ETSI	ETSI (Max. 10 dBm transmit power output)*
Japan	R201WW07215214	R201WW082151111 (Max. 10 dBm transmit power output)*
Australia	C-Tick	C-Tick

Características eléctricas del módulo XBEE

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
V _{IL}	Input Low Voltage	All Digital Inputs	-	-	0.35 * VCC	V
V _{IH}	Input High Voltage	All Digital Inputs	0.7 * VCC	-	-	V
V _{OL}	Output Low Voltage	I _{OL} = 2 mA, VCC >= 2.7 V	-	-	0.5	V
V _{OH}	Output High Voltage	I _{OH} = -2 mA, VCC >= 2.7 V	VCC - 0.5	-	-	V
I _{IN}	Input Leakage Current	V _{IN} = VCC or GND, all inputs, per pin	-	0.025	1	μA
I _{OZ}	High Impedance Leakage Current	V _{IN} = VCC or GND, all I/O High-Z, per pin	-	0.025	1	μA
TX	Transmit Current	VCC = 3.3 V	-	45 (XBee) 215, 140 (PRO, Int)	-	mA
RX	Receive Current	VCC = 3.3 V	-	50 (XBee) 55 (PRO)	-	mA
PWR-DWN	Power-down Current	SM parameter = 1	-	< 10	-	μA
V _{REFH}	VREF - Analog-to-Digital converter reference range		2.08	-	V _{DDAD} *	V
I _{REF}	VREF - Reference Supply Current	Enabled	-	200	-	μA
		Disabled or Sleep Mode	-	< 0.01	0.02	μA
V _{INDC}	Analog Input Voltage ¹		V _{SSAD} - 0.3	-	V _{DDAD} + 0.3	V
R _{AS}	Source Impedance at Input ²		-	-	10	kΩ
V _{AIN}	Analog Input Voltage ³		V _{REFL}	-	V _{REFH}	V
RES	Ideal Resolution (1 LSB) ⁴	2.08V ≤ V _{DDAD} ≤ 3.6V	2.031	-	3.516	mV
DNL	Differential Non-linearity ⁵		-	±0.5	±1.0	LSB
INL	Integral Non-linearity ⁶		-	±0.5	±1.0	LSB
E _{ZS}	Zero-scale Error ⁷		-	±0.4	±1.0	LSB
F _{FS}	Full-scale Error ⁸		-	±0.4	±1.0	LSB
E _{IL}	Input Leakage Error ⁹		-	±0.05	±5.0	LSB
E _{TU}	Total Unadjusted Error ¹⁰		-	±1.1	±2.5	LSB

Esquemáticos Modulo XBEE



Asignaciones de los pines del módulo XBEE

Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate / AD5 / DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS / AD6 / DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital I/O 0

Especificaciones Puente H



VNH2SP30-E

Automotive fully integrated H-bridge motor driver

Features

Type	$R_{DS(on)}$	I_{out}	V_{CCmax}
VNH2SP30-E	19m Ω max (per leg)	30A	41V

- 5V logic level compatible inputs
- Undervoltage and overvoltage shut-down
- Overvoltage clamp
- Thermal shut down
- Cross-conduction protection
- Linear current limiter
- Very low stand-by power consumption
- PWM operation up to 20 kHz
- Protection against loss of ground and loss of V_{CC}
- Current sense output proportional to motor current
- Package: ECOPACK®

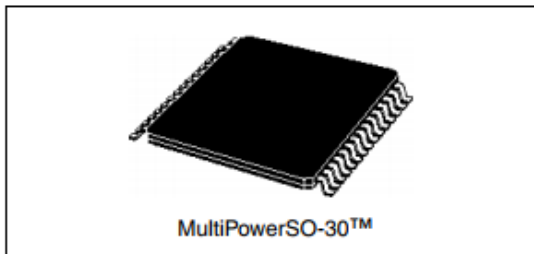
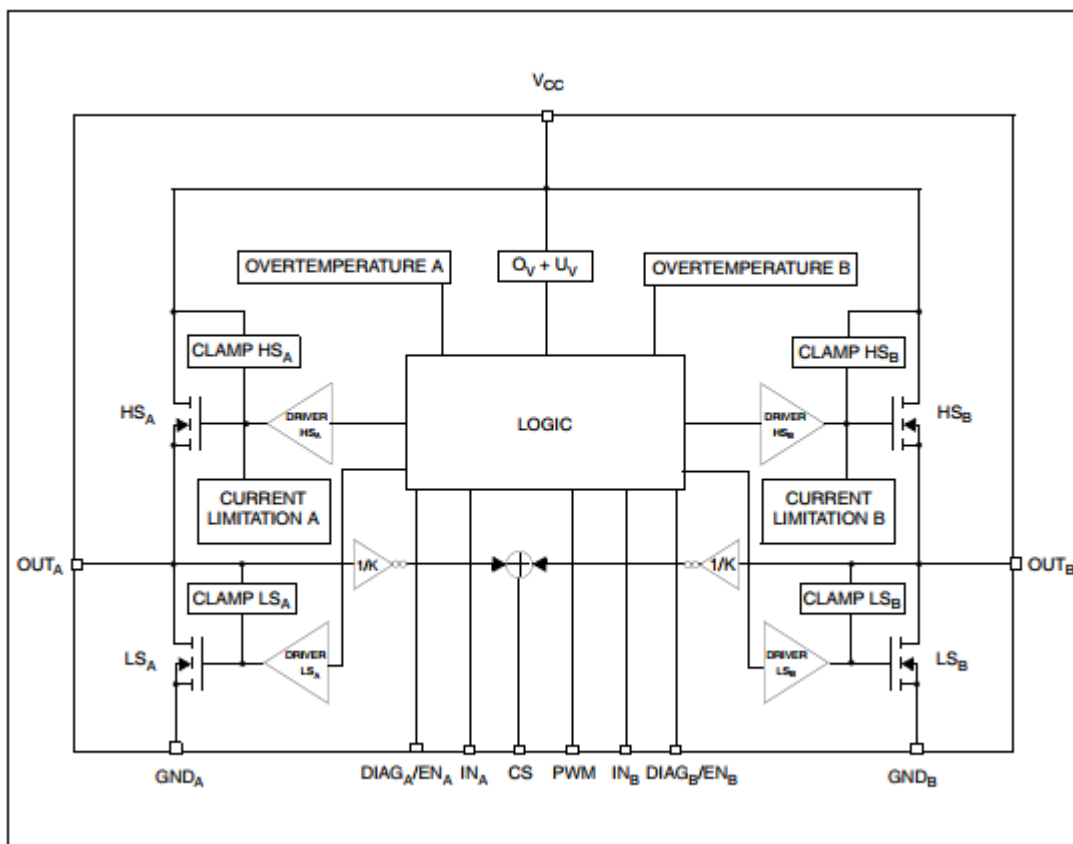


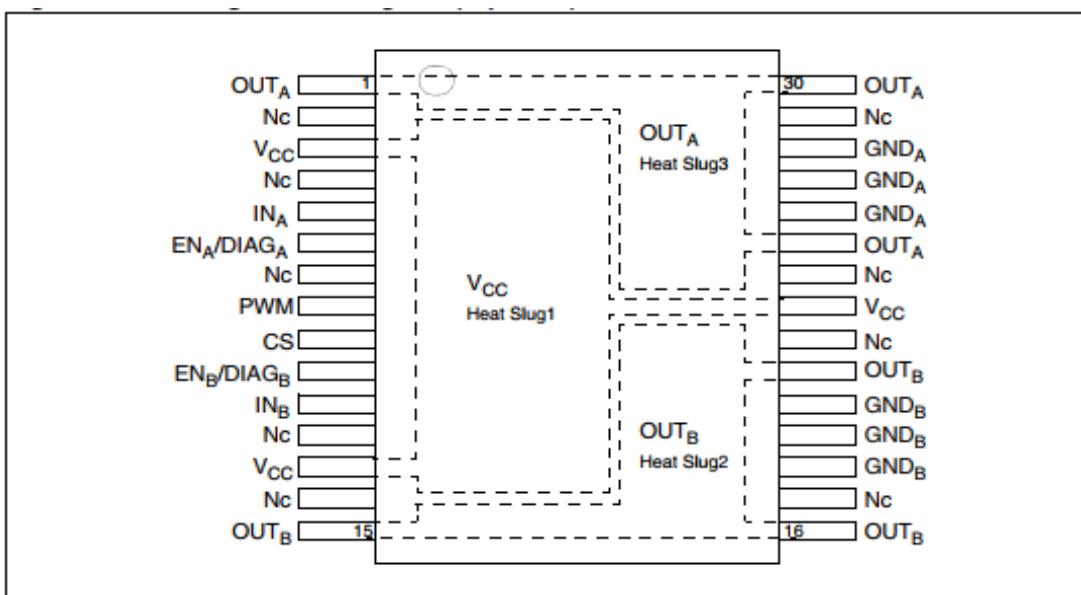
Diagrama a bloques



Descripción de los bloques

Name	Description
Logic control	Allows the turn-on and the turn-off of the high side and the low side switches according to the truth table
Ovoltage + undervoltage	Shuts down the device outside the range [5.5V..16V] for the battery voltage
High side and low side clamp voltage	Protects the high side and the low side switches from the high voltage on the battery line in all configurations for the motor
High side and low side driver	Drives the gate of the concerned switch to allow a proper $R_{DS(on)}$ for the leg of the bridge
Linear current limiter	Limits the motor current by reducing the high side switch gate-source voltage when short-circuit to ground occurs
Overtemperature protection	In case of short-circuit with the increase of the junction's temperature, shuts down the concerned high side to prevent its degradation and to protect the die
Fault detection	Signals an abnormal behavior of the switches in the half-bridge A or B by pulling low the concerned $EN_x/DIAG_x$ pin

Diagrama de configuración



Definición de pines y funciones

Pin No	Symbol	Function
1, 25, 30	OUT _A , Heat Slug3	Source of high side switch A / Drain of low side switch A
2, 4, 7, 12, 14, 17, 22, 24, 29	NC	Not connected
3, 13, 23	V _{CC} , Heat Slug1	Drain of high side switches and power supply voltage
6	EN _A /DIAG _A	Status of high side and low side switches A; open drain output
5	IN _A	Clockwise input
8	PWM	PWM input
9	CS	Output of current sense
11	IN _B	Counter clockwise input
10	EN _B /DIAG _B	Status of high side and low side switches B; open drain output
15, 16, 21	OUT _B , Heat Slug2	Source of high side switch B / Drain of low side switch B
26, 27, 28	GND _A	Source of low side switch A ⁽¹⁾
18, 19, 20	GND _B	Source of low side switch B ⁽¹⁾

Descripción de la funciones de los pines

V_{CC}	Battery connection
GND_A, GND_B	Power grounds; must always be externally connected together
OUT_A, OUT_B	Power connections to the motor
IN_A, IN_B	Voltage controlled input pins with hysteresis, CMOS compatible. These two pins control the state of the bridge in normal operation according to the truth table (brake to V_{CC} , brake to GND, clockwise and counterclockwise).
PWM	Voltage controlled input pin with hysteresis, CMOS compatible. Gates of low side FETs are modulated by the PWM signal during their ON phase allowing speed control of the motor.
$EN_A/DIAG_A, EN_B/DIAG_B$	Open drain bidirectional logic pins. These pins must be connected to an external pull up resistor. When externally pulled low, they disable half-bridge A or B. In case of fault detection (thermal shutdown of a high side FET or excessive ON state voltage drop across a low side FET), these pins are pulled low by the device (see truth table in fault condition).
CS	Analog current sense output. This output sources a current proportional to the motor current. The information can be read back as an analog voltage across an external resistor.

Especificaciones eléctricas

Symbol	Parameter	Value	Unit
V_{CC}	Supply voltage	+41	V
I_{max}	Maximum output current (continuous)	30	A
I_R	Reverse output current (continuous)	-30	
I_{IN}	Input current (IN_A and IN_B pins)	± 10	mA
I_{EN}	Enable input current ($DIAG_A/EN_A$ and $DIAG_B/EN_B$ pins)	± 10	
I_{pw}	PWM input current	± 10	
V_{CS}	Current sense maximum voltage	-3/+15	V
V_{ESD}	Electrostatic discharge ($R = 1.5k\Omega, C = 100pF$)		
	– CS pin	2	kV
	– logic pins	4	kV
	– output pins: OUT_A, OUT_B, V_{CC}	5	kV
T_j	Junction operating temperature	Internally limited	°C
T_c	Case operating temperature	-40 to 150	
T_{STG}	Storage temperature	-55 to 150	

Symbol	Parameter	Test conditions	Min	Typ	Max	Unit
V_{CC}	Operating supply voltage		5.5		16	V
I_S	Supply current	Off state with all Fault Cleared & $EN_X=0$ $I_{N_A} = I_{N_B} = PWM = 0$; $T_j = 25^\circ C$; $V_{CC} = 13V$ $I_{N_A} = I_{N_B} = PWM = 0$ Off state: $I_{N_A} = I_{N_B} = PWM = 0$		12	30 60	μA μA mA
		On state: I_{N_A} or $I_{N_B} = 5V$, no PWM			10	mA
R_{ONHS}	Static high side resistance	$I_{OUT} = 15A$; $T_j = 25^\circ C$			14	m Ω
		$I_{OUT} = 15A$; $T_j = -40$ to $150^\circ C$			28	
R_{ONLS}	Static low side resistance	$I_{OUT} = 15A$; $T_j = 25^\circ C$			5	
		$I_{OUT} = 15A$; $T_j = -40$ to $150^\circ C$			10	
V_f	High side free-wheeling diode forward voltage	$I_f = 15A$		0.8	1.1	V
$I_{L(off)}$	High side off state output current (per channel)	$T_j = 25^\circ C$; $V_{OUTX} = EN_X = 0V$; $V_{CC} = 13V$			3	μA
		$T_j = 125^\circ C$; $V_{OUTX} = EN_X = 0V$; $V_{CC} = 13V$			5	
I_{RM}	Dynamic cross-conduction current	$I_{OUT} = 15A$ (see Figure 7)		0.7		A

Entradas lógicas

Symbol	Parameter	Test conditions	Min	Typ	Max	Unit
V_{IL}	Input low level voltage	Normal operation ($DIAG_X/EN_X$ pin acts as an input pin)			1.25	V
V_{IH}	Input high level voltage		3.25			
V_{IHYST}	Input hysteresis voltage		0.5			
V_{ICL}	Input clamp voltage	$I_{IN} = 1mA$	5.5	6.3	7.5	
		$I_{IN} = -1mA$	-1.0	-0.7	-0.3	
I_{INL}	Input low current	$V_{IN} = 1.25V$	1			μA
I_{INH}	Input high current	$V_{IN} = 3.25V$			10	
V_{DIAG}	Enable output low level voltage	Fault operation ($DIAG_X/EN_X$ pin acts as an output pin); $I_{EN} = 1mA$			0.4	V

Switches

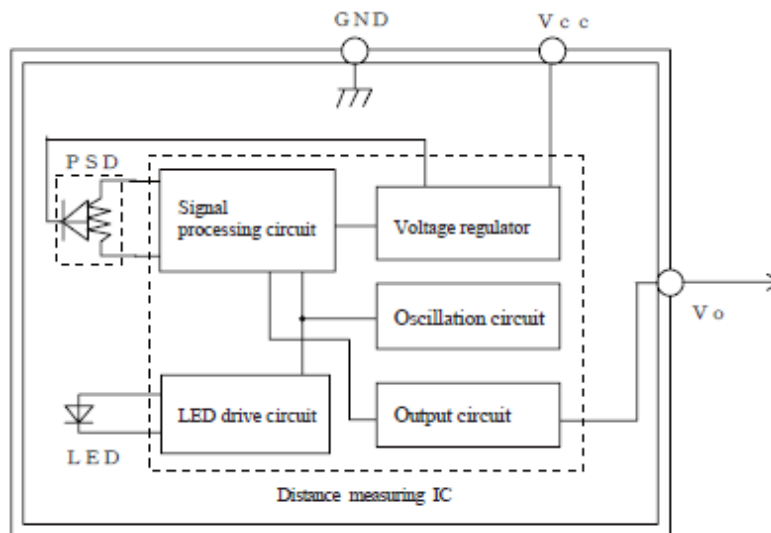
Symbol	Parameter	Test conditions	Min	Typ	Max	Unit
f	PWM frequency		0		20	kHz
t _{d(on)}	Turn-on delay time	Input rise time < 1μs (see Figure 6)			250	μs
t _{d(off)}	Turn-off delay time	Input rise time < 1μs (see Figure 6)			250	
t _r	Rise time	(see Figure 5)		1	1.6	
t _f	Fall time	(see Figure 5)		1.2	2.4	
t _{DEL}	Delay time during change of operating mode	(see Figure 4)	300	600	1800	
t _{rr}	High side free wheeling diode reverse recovery time	(see Figure 7)		110		ns
t _{off(min)} ⁽¹⁾	PWM minimum off time	9V < V _{CC} < 16V; T _j = 25°C; L = 250μH; I _{OUT} = 15A			6	μs

PWM

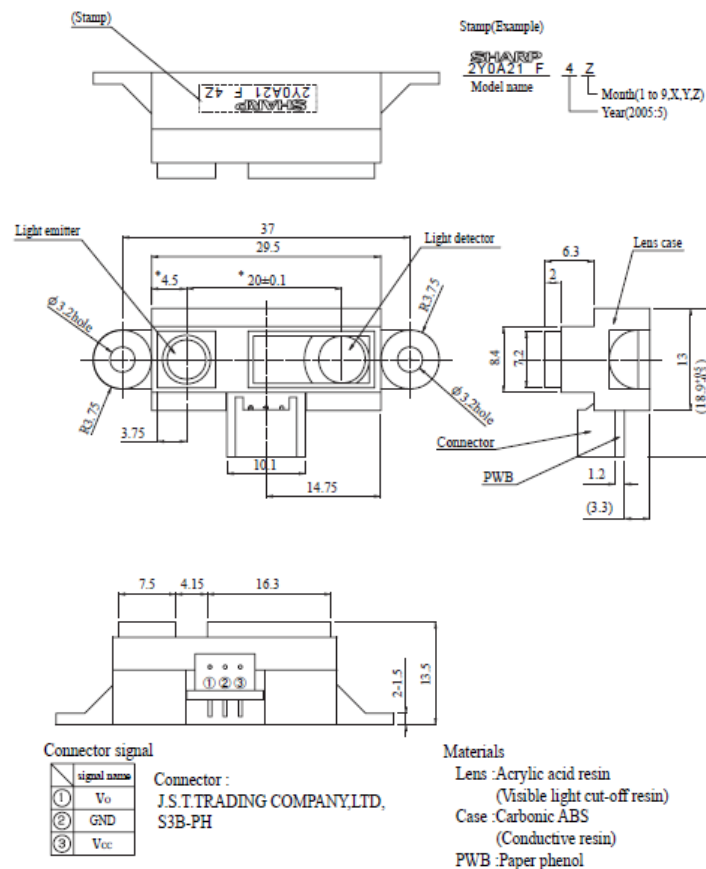
Symbol	Parameter	Test conditions	Min	Typ	Max	Unit
V _{pwl}	PWM low level voltage				1.25	V
I _{pwl}	PWM pin current	V _{pwl} = 1.25V	1			μA
V _{pwh}	PWM high level voltage		3.25			V
I _{pwh}	PWM pin current	V _{pwh} = 3.25V			10	μA
V _{pwhyst}	PWM hysteresis voltage		0.5			V
V _{pwcl}	PWM clamp voltage	I _{pw} = 1mA	V _{CC} + 0.3	V _{CC} + 0.7	V _{CC} + 1.0	
		I _{pw} = -1mA	-6.0	-4.5	-3.0	
C _{INPWM}	PWM pin input capacitance	V _{IN} = 2.5V			25	pF

Especificaciones Sensor Infrarrojo SHARP GP2Y0A21YK0F

Diagrama de bloques



Dimensiones físicas



Especificaciones eléctricas

■ Absolute Maximum Ratings ($T_a=25^{\circ}\text{C}, V_{CC}=5\text{V}$)

Parameter	Symbol	Rating	Unit
Supply voltage	V_{CC}	-0.3 to +7	V
Output terminal voltage	V_O	-0.3 to $V_{CC}+0.3$	V
Operating temperature	T_{opr}	-10 to +60	$^{\circ}\text{C}$
Storage temperature	T_{stg}	-40 to +70	$^{\circ}\text{C}$

■ Electro-optical Characteristics ($T_a=25^{\circ}\text{C}, V_{CC}=5\text{V}$)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Average supply current	I_{CC}	L=80cm (Note 1)	—	30	40	mA
Distance measuring	ΔL	(Note 1)	10	—	80	cm
Output voltage	V_O	L=80cm (Note 1)	0.25	0.4	0.55	V
Output voltage differential	ΔV_O	Output voltage difference between L=10cm and L=80cm (Note 1)	1.65	1.9	2.15	V

* L : Distance to reflective object

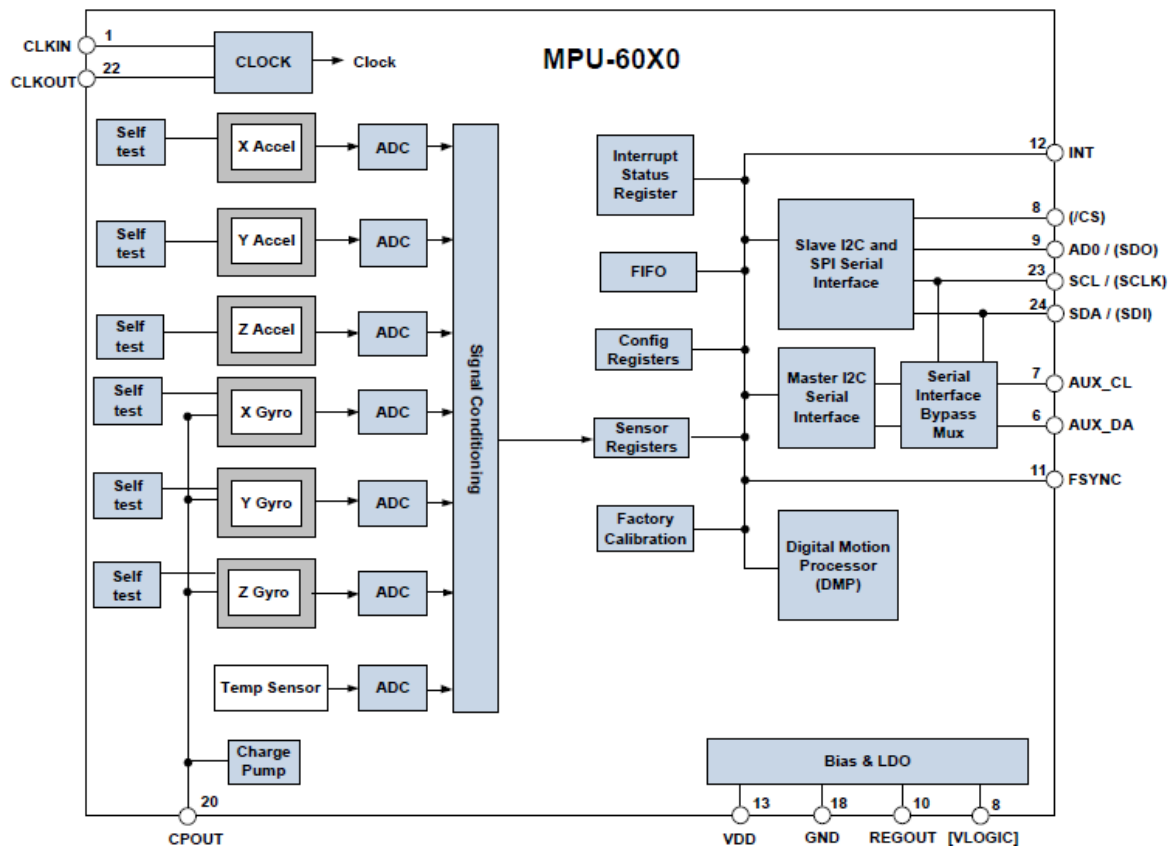
Note 1 : Using reflective object : White paper (Made by Kodak Co., Ltd. gray cards R-27·white face, reflectance; 90%)

■ Recommended operating conditions

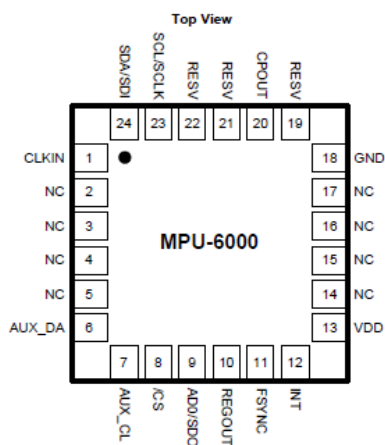
Parameter	Symbol	Rating	Unit
Supply voltage	V_{CC}	4.5 to 5.5	V

Especificaciones Sensor IMU-6050 ó MPU-6050

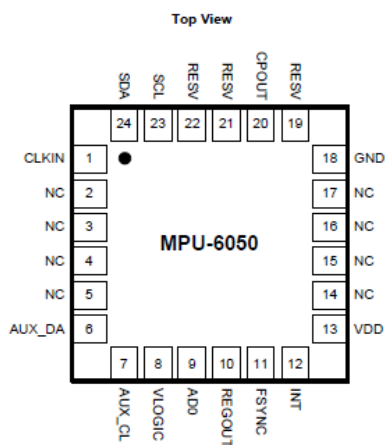
Diagrama de bloques



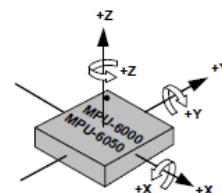
Descripción de pines



QFN Package
24-pin, 4mm x 4mm x 0.9mm



QFN Package
24-pin, 4mm x 4mm x 0.9mm



Orientation of Axes of Sensitivity and
Polarity of Rotation

Pin Number	MPU-6000	MPU-6050	Pin Name	Pin Description
1	Y	Y	CLKIN	Optional external reference clock input. Connect to GND if unused.
6	Y	Y	AUX_DA	I ² C master serial data, for connecting to external sensors
7	Y	Y	AUX_CL	I ² C Master serial clock, for connecting to external sensors
8	Y		/CS	SPI chip select (0=SPI mode)
8		Y	VLOGIC	Digital I/O supply voltage
9	Y		AD0 / SDO	I ² C Slave Address LSB (AD0); SPI serial data output (SDO)
9		Y	AD0	I ² C Slave Address LSB (AD0)
10	Y	Y	REGOUT	Regulator filter capacitor connection
11	Y	Y	FSYNC	Frame synchronization digital input. Connect to GND if unused.
12	Y	Y	INT	Interrupt digital output (totem pole or open-drain)
13	Y	Y	VDD	Power supply voltage and Digital I/O supply voltage
18	Y	Y	GND	Power supply ground
19, 21	Y	Y	RESV	Reserved. Do not connect.
20	Y	Y	CPOUT	Charge pump capacitor connection
22	Y	Y	RESV	Reserved. Do not connect.
23	Y		SCL / SCLK	I ² C serial clock (SCL); SPI serial clock (SCLK)
23		Y	SCL	I ² C serial clock (SCL)
24	Y		SDA / SDI	I ² C serial data (SDA); SPI serial data input (SDI)
24		Y	SDA	I ² C serial data (SDA)
2, 3, 4, 5, 14, 15, 16, 17	Y	Y	NC	Not internally connected. May be used for PCB trace routing.

Especificaciones eléctricas

VDD = 2.375V-3.46V, VLOGIC (MPU-6050 only) = 1.8V±5% or VDD, T_A = 25°C

PARAMETER	CONDITIONS	MIN	TYP	MAX	Units	Notes
TEMPERATURE SENSOR						
Range			-40 to +85		°C	
Sensitivity	Untrimmed		340		LSB/°C	
Temperature Offset	35°C		-521		LSB	
Linearity	Best fit straight line (-40°C to +85°C)		±1		°C	
VDD POWER SUPPLY						
Operating Voltages		2.375		3.46	V	
Normal Operating Current	Gyroscope + Accelerometer + DMP		3.9		mA	
	Gyroscope + Accelerometer (DMP disabled)		3.8		mA	
	Gyroscope + DMP (Accelerometer disabled)		3.7		mA	
	Gyroscope only (DMP & Accelerometer disabled)		3.6		mA	
	Accelerometer only (DMP & Gyroscope disabled)		500		µA	
Accelerometer Low Power Mode Current	1.25 Hz update rate		10		µA	
	5 Hz update rate		20		µA	
	20 Hz update rate		70		µA	
	40 Hz update rate		140		µA	
Full-Chip Idle Mode Supply Current			5		µA	
Power Supply Ramp Rate	Monotonic ramp. Ramp rate is 10% to 90% of the final value			100	ms	
VLOGIC REFERENCE VOLTAGE						
Voltage Range	MPU-6050 only VLOGIC must be ≤VDD at all times	1.71		VDD	V	
Power Supply Ramp Rate	Monotonic ramp. Ramp rate is 10% to 90% of the final value			3	ms	
Normal Operating Current			100		µA	
TEMPERATURE RANGE						
Specified Temperature Range	Performance parameters are not applicable beyond Specified Temperature Range	-40		+85	°C	

SERIAL INTERFACE SPI Operating Frequency, All Registers Read/Write SPI Operating Frequency, Sensor and Interrupt Registers Read Only I ² C Operating Frequency	MPU-6000 only, Low Speed Characterization		100 ±10%		kHz	
	MPU-6000 only, High Speed Characterization		1 ±10%		MHz	
	MPU-6000 only		20 ±10%		MHz	
	All registers, Fast-mode All registers, Standard-mode			400 100	kHz kHz	
I²C ADDRESS	AD0 = 0 AD0 = 1		1101000 1101001			
DIGITAL INPUTS (SDI/SDA, AD0, SCLK/SCL, FSYNC, /CS, CLKIN)						
V _{IH} , High Level Input Voltage	MPU-6000 MPU-6050	0.7*VDD 0.7*VLOGIC			V V	
V _{IL} , Low Level Input Voltage	MPU-6000 MPU-6050			0.3*VDD 0.3*VLOGIC	V V	
C _I , Input Capacitance			< 5		pF	
DIGITAL OUTPUT (SDO, INT)						
V _{OH} , High Level Output Voltage	R _{LOAD} =1MΩ; MPU-6000 R _{LOAD} =1MΩ; MPU-6050	0.9*VDD 0.9*VLOGIC			V V	
V _{OL1} , LOW-Level Output Voltage	R _{LOAD} =1MΩ; MPU-6000 R _{LOAD} =1MΩ; MPU-6050			0.1*VDD 0.1*VLOGIC	V V	
V _{OLINT1} , INT Low-Level Output Voltage	OPEN=1, 0.3mA sink Current			0.1	V	
Output Leakage Current	OPEN=1		100		nA	
t _{INT} , INT Pulse Width	LATCH_INT_EN=0		50		μs	

Parameters	Conditions	Typical			Units	Notes
Primary I²C I/O (SCL, SDA)						
V _{IL} , LOW-Level Input Voltage	MPU-6000	-0.5 to 0.3*VDD			V	
V _{IH} , HIGH-Level Input Voltage	MPU-6000	0.7*VDD to VDD + 0.5V			V	
V _{hys} , Hysteresis	MPU-6000	0.1*VDD			V	
V _{IL} , LOW Level Input Voltage	MPU-6050	-0.5V to 0.3*VLOGIC			V	
V _{IH} , HIGH-Level Input Voltage	MPU-6050	0.7*VLOGIC to VLOGIC + 0.5V			V	
V _{hys} , Hysteresis	MPU-6050	0.1*VLOGIC			V	
V _{OL1} , LOW-Level Output Voltage	3mA sink current	0 to 0.4			V	
I _{OL} , LOW-Level Output Current	V _{OL} = 0.4V	3			mA	
	V _{OL} = 0.6V	5			mA	
Output Leakage Current		100			nA	
t _{of} , Output Fall Time from V _{IHmax} to V _{ILmax}	C _b bus capacitance in pF	20+0.1C _b to 250			ns	
C _i , Capacitance for Each I/O pin		< 10			pF	
Auxiliary I²C I/O (AUX_CL, AUX_DA)	MPU-6050: AUX_VDDIO=0					
V _{IL} , LOW-Level Input Voltage		-0.5V to 0.3*VLOGIC			V	
V _{IH} , HIGH-Level Input Voltage		0.7*VLOGIC to VLOGIC + 0.5V			V	
V _{hys} , Hysteresis		0.1*VLOGIC			V	
V _{OL1} , LOW-Level Output Voltage	VLOGIC > 2V; 1mA sink current	0 to 0.4			V	
V _{OL3} , LOW-Level Output Voltage	VLOGIC < 2V; 1mA sink current	0 to 0.2*VLOGIC			V	
I _{OL} , LOW-Level Output Current	V _{OL} = 0.4V	1			mA	
	V _{OL} = 0.6V	1			mA	
Output Leakage Current		100			nA	
t _{of} , Output Fall Time from V _{IHmax} to V _{ILmax}	C _b bus capacitance in pF	20+0.1C _b to 250			ns	
C _i , Capacitance for Each I/O pin		< 10			pF	
INTERNAL CLOCK SOURCE	CLK_SEL=0,1,2,3					
Gyroscope Sample Rate, Fast	DLPFCFG=0 SAMPLERATEDIV = 0		8		kHz	
Gyroscope Sample Rate, Slow	DLPFCFG=1,2,3,4,5, or 6 SAMPLERATEDIV = 0		1		kHz	
Accelerometer Sample Rate			1		kHz	
Clock Frequency Initial Tolerance	CLK_SEL=0, 25°C	-5		+5	%	
	CLK_SEL=1,2,3; 25°C	-1		+1	%	
Frequency Variation over Temperature	CLK_SEL=0		-15 to +10		%	
	CLK_SEL=1,2,3		±1		%	
PLL Settling Time	CLK_SEL=1,2,3		1	10	ms	
EXTERNAL 32.768kHz CLOCK	CLK_SEL=4					
External Clock Frequency			32.768		kHz	
External Clock Allowable Jitter	Cycle-to-cycle rms		1 to 2		µs	
Gyroscope Sample Rate, Fast	DLPFCFG=0 SAMPLERATEDIV = 0		8.192		kHz	
Gyroscope Sample Rate, Slow	DLPFCFG=1,2,3,4,5, or 6 SAMPLERATEDIV = 0		1.024		kHz	
Accelerometer Sample Rate			1.024		kHz	
PLL Settling Time			1	10	ms	
EXTERNAL 19.2MHz CLOCK	CLK_SEL=5					
External Clock Frequency			19.2		MHz	
Gyroscope Sample Rate	Full programmable range	3.9		8000	Hz	
Gyroscope Sample Rate, Fast Mode	DLPFCFG=0 SAMPLERATEDIV = 0		8		kHz	
Gyroscope Sample Rate, Slow Mode	DLPFCFG=1,2,3,4,5, or 6 SAMPLERATEDIV = 0		1		kHz	
Accelerometer Sample Rate			1		kHz	
PLL Settling Time			1	10	ms	

Valores máximos de operación

Parameter	Rating
Supply Voltage, VDD	-0.5V to +6V
VLOGIC Input Voltage Level (MPU-6050)	-0.5V to VDD + 0.5V
REGOUT	-0.5V to 2V
Input Voltage Level (CLKIN, AUX_DA, AD0, FSYNC, INT, SCL, SDA)	-0.5V to VDD + 0.5V
CPOUT (2.5V ≤ VDD ≤ 3.6V)	-0.5V to 30V
Acceleration (Any Axis, unpowered)	10,000g for 0.2ms
Operating Temperature Range	-40°C to +105°C
Storage Temperature Range	-40°C to +125°C
Electrostatic Discharge (ESD) Protection	2kV (HBM); 250V (MM)
Latch-up	JEDEC Class II (2), 125°C ±100mA

Especificaciones de Giroscopio

VDD = 2.375V-3.46V, VLOGIC (MPU-6050 only) = 1.8V±5% or VDD, T_A = 25°C

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
GYROSCOPE SENSITIVITY						
Full-Scale Range	FS_SEL=0		±250		°/s	
	FS_SEL=1		±500		°/s	
	FS_SEL=2		±1000		°/s	
	FS_SEL=3		±2000		°/s	
Gyroscope ADC Word Length			16		bits	
Sensitivity Scale Factor	FS_SEL=0		131		LSB/(°/s)	
	FS_SEL=1		65.5		LSB/(°/s)	
	FS_SEL=2		32.8		LSB/(°/s)	
	FS_SEL=3		16.4		LSB/(°/s)	
Sensitivity Scale Factor Tolerance	25°C	-3		+3	%	
Sensitivity Scale Factor Variation Over Temperature			±2		%	
Nonlinearity	Best fit straight line; 25°C		0.2		%	
Cross-Axis Sensitivity			±2		%	
GYROSCOPE ZERO-RATE OUTPUT (ZRO)						
Initial ZRO Tolerance	25°C		±20		°/s	
ZRO Variation Over Temperature	-40°C to +85°C		±20		°/s	
Power-Supply Sensitivity (1-10Hz)	Sine wave, 100mVpp; VDD=2.5V		0.2		°/s	
Power-Supply Sensitivity (10 - 250Hz)	Sine wave, 100mVpp; VDD=2.5V		0.2		°/s	
Power-Supply Sensitivity (250Hz - 100kHz)	Sine wave, 100mVpp; VDD=2.5V		4		°/s	
Linear Acceleration Sensitivity	Static		0.1		°/s/g	
SELF-TEST RESPONSE						
Relative	Change from factory trim	-14		14	%	1
GYROSCOPE NOISE PERFORMANCE						
Total RMS Noise	FS_SEL=0 DLPFCFG=2 (100Hz)		0.05		°/s-rms	
Low-frequency RMS noise	Bandwidth 1Hz to10Hz		0.033		°/s-rms	
Rate Noise Spectral Density	At 10Hz		0.005		°/s/√Hz	
GYROSCOPE MECHANICAL FREQUENCIES						
X-Axis		30	33	36	kHz	
Y-Axis		27	30	33	kHz	
Z-Axis		24	27	30	kHz	
LOW PASS FILTER RESPONSE						
	Programmable Range	5		256	Hz	
OUTPUT DATA RATE						
	Programmable	4		8,000	Hz	
GYROSCOPE START-UP TIME						
ZRO Settling (from power-on)	DLPFCFG=0 to ±1% of Final		30		ms	

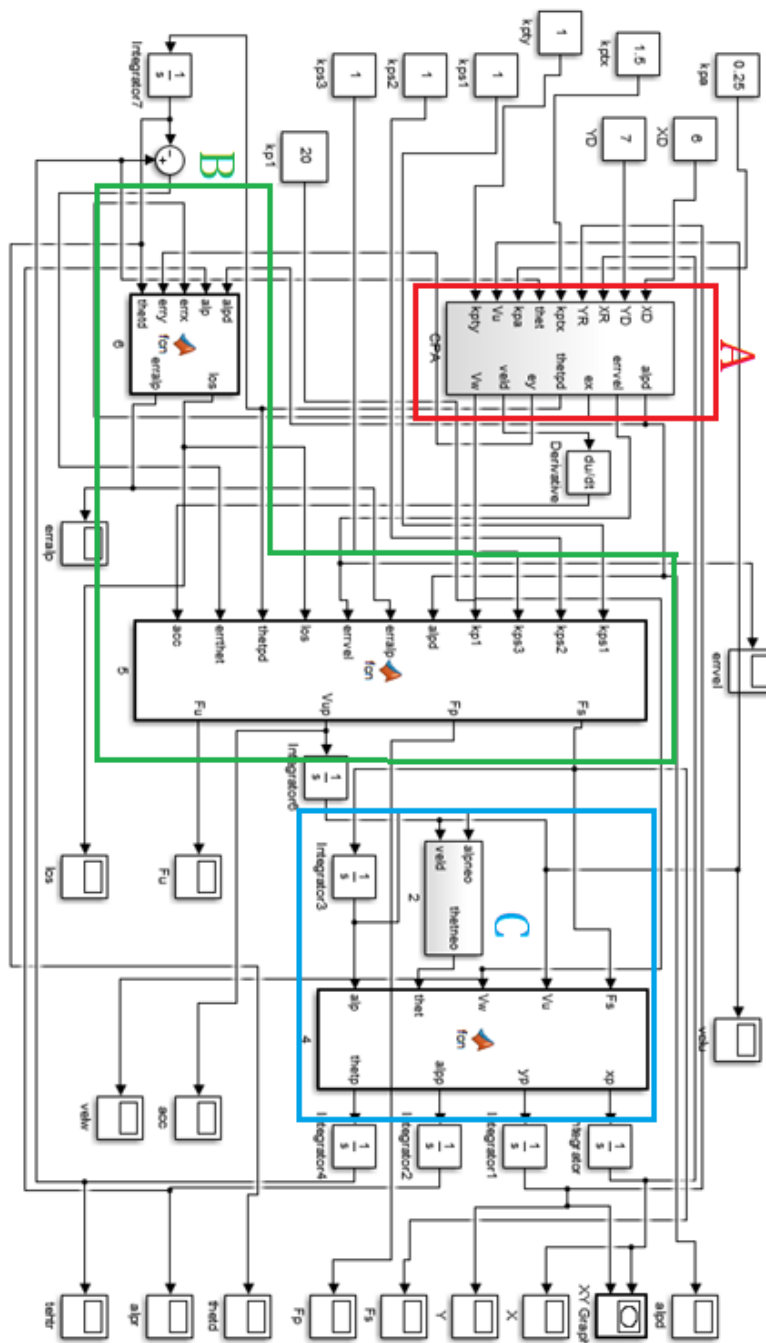
Especificaciones de acelerómetro

VDD = 2.375V-3.46V, V_{LOGIC} (MPU-6050 only) = 1.8V±5% or VDD, T_A = 25°C

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
ACCELEROMETER SENSITIVITY						
Full-Scale Range	AFS_SEL=0		±2		g	
	AFS_SEL=1		±4		g	
	AFS_SEL=2		±8		g	
	AFS_SEL=3		±16		g	
ADC Word Length	Output in two's complement format		16		bits	
Sensitivity Scale Factor	AFS_SEL=0		16,384		LSB/g	
	AFS_SEL=1		8,192		LSB/g	
	AFS_SEL=2		4,096		LSB/g	
	AFS_SEL=3		2,048		LSB/g	
Initial Calibration Tolerance			±3		%	
Sensitivity Change vs. Temperature	AFS_SEL=0, -40°C to +85°C		±0.02		%/°C	
Nonlinearity	Best Fit Straight Line		0.5		%	
Cross-Axis Sensitivity			±2		%	
ZERO-G OUTPUT						
Initial Calibration Tolerance	X and Y axes		±50		mg	1
	Z axis		±80		mg	
Zero-G Level Change vs. Temperature	X and Y axes, 0°C to +70°C		±35		mg	
	Z axis, 0°C to +70°C		±60		mg	
SELF TEST RESPONSE						
Relative	Change from factory trim	-14		14	%	2
NOISE PERFORMANCE						
Power Spectral Density	@10Hz, AFS_SEL=0 & ODR=1kHz		400		μg/√Hz	
LOW PASS FILTER RESPONSE						
	Programmable Range	5		260	Hz	
OUTPUT DATA RATE						
	Programmable Range	4		1,000	Hz	
INTELLIGENCE FUNCTION INCREMENT			32		mq/LSB	

Apéndice C Programación Simulink

Programa general

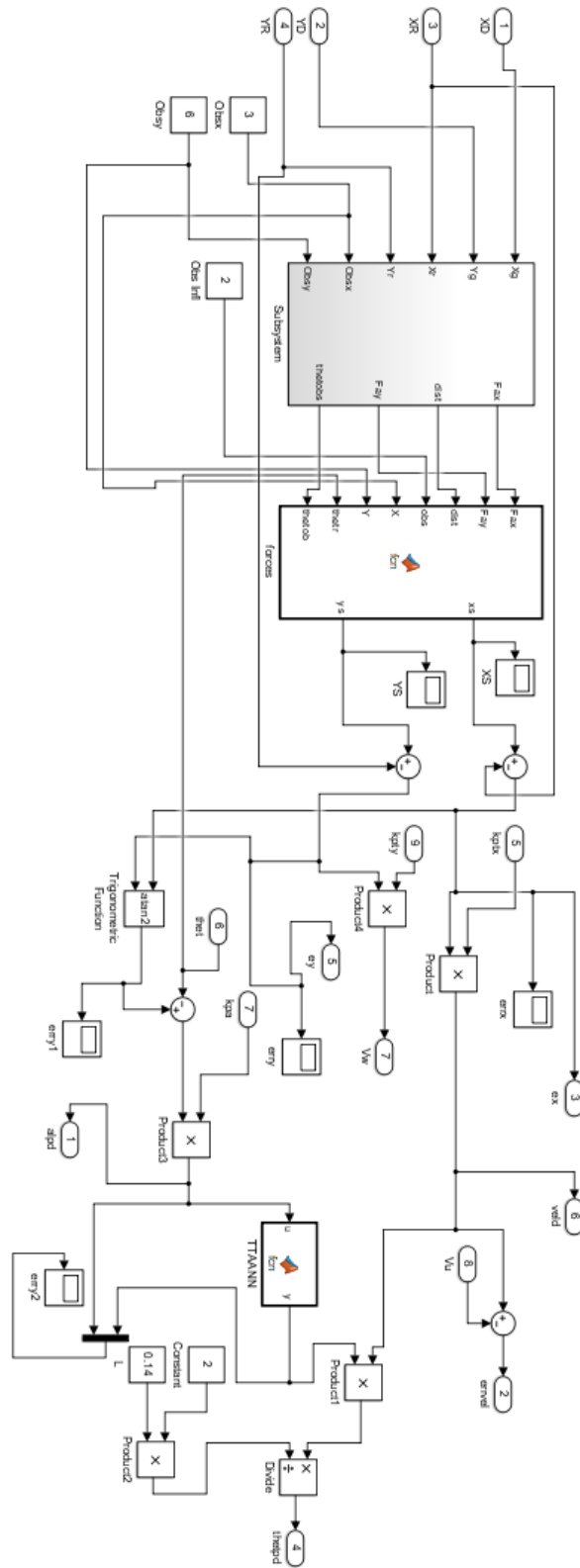


A Bloque de Campos Potenciales Artificiales

B Bloque de Control Punto a Punto

C Bloque de Modelo Dinámico

Bloque A Código Control de Campos Potenciales Artificiales



Bloque B Código Control Punto a Punto

```
function [los, erralp] = fcn(alpd, alp, errx, erry, thetd)

thetv=atand((tand(alpd))/2);
los=(errx*(sind(thetd+thetv)))+(erry*(cosd(thetd+thetv)));

erralp=alp-alpd;
end

function [Fs, Fp, Vup, Fu] = fcn(kps1, kps2, kps3, kp1, alpd, erralp,
errvel, los, thetpd, errthet, acc)

J=5.6;
L=0.14;
m=5.4;

Fs=-(kps1*(erralp))-(kps2*(errthet))-(kps3*(los));

y=(m+(((tand(alpd))/2*L)^2)*(m*(L^2)+J));
gp=y^-1;
z=cosd(alpd);

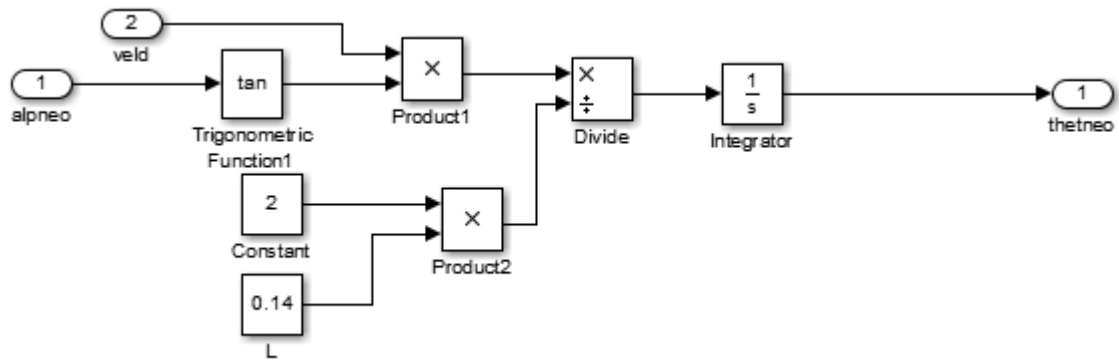
gu=(gp/z);
x=(2*L*(cosd(alpd))^2);

gs=gp*(m*(L^2)+J)*(thetpd/x);
Fu=(acc-(kp1*(errvel))-(2*gs*Fs))/(2*gu);

Fp1=(gp^-1)*(acc-(gs*Fs)-(gu*Fu));
Fp2=(gp^-1)*(kp1*(errvel));
Fp=Fp1+Fp2;

Vup=(gp*Fp)+(gs*Fs)+(gu*Fu);
end
```

Bloque C Código de Modelo Dinámico



```
function [xp, yp, alpp, thetp] = fcn(Fs, Vu, Vw, thet, alp)
```

```
L=0.14;
```

```
thetp=Vu*tand(alp)/(2*L);
```

```
alpp=Fs;
```

```
xp=(Vu*cosd(thet))-(Vw*sind(thet));
```

```
yp=(Vu*sind(thet))+(Vw*cosd(thet));
```

```
end
```

Apéndice D Publicaciones



EL TECNOLÓGICO NACIONAL DE MÉXICO A TRAVÉS DEL INSTITUTO TECNOLÓGICO DE LA PAZ

OTORGA EL PRESENTE

RECONOCIMIENTO

A

HERMES JAVIER RAMIREZ GARCÍA, AMPARO DORA PALOMINO MERINO Y
MARIA AURORA DIOZCORA VARGAS TREVIÑO

POR SU ARTÍCULO: CONTROL DE UN ROBOT MÓVIL TIPO ACKERMAN POR MEDIO DEL
MODELO DINÁMICO
PRESENTADO EN EL CUARTO CONGRESO INTERNACIONAL DE
ROBÓTICA Y COMPUTACIÓN
CELEBRADO DEL 2 AL 4 DE MAYO DEL AÑO EN CURSO.

LA PAZ, B.C.S., 4 DE MAYO DE 2017.

ATENTAMENTE
“CIENCIA ES VERDAD, TÉCNICA ES LIBERTAD”


ING. JESÚS DAVID ESTRADA RUÍZ
DIRECTOR


SECRETARÍA DE EDUCACIÓN PÚBLICA
TECNOLÓGICO NACIONAL
DE MÉXICO
INSTITUTO TECNOLÓGICO
DE LA PAZ
DIRECCIÓN

17035

TOEFL ITP Score Report

Name of Institution: FACULTAD DE LENGUAS DE LA BUAP

Name: RAMIREZ HERMES J

Student Number:

DOB: 05/31/1990

Sex: M Degree:

Times Taken TOEFL: None

Native Country: Mexico

Native Language: Spanish

Scaled Scores:

Listening Comprehension: 58

Test Date: 03/03/2017

Structure & Written Expression: 60

Form: TOEFL ITP

Reading Comprehension: 57

Total Score: 583



The face of this document has a security background. The back contains a watermark. Hold at an angle to view.

The TOEFL® ITP Assessment Series is designed to be used for placement, progress monitoring, and exit purposes. TOEFL® ITP scores can also be used for admissions to programs and institutions where English is not the dominant language of instruction for content courses. Learn more at www.ets.org/toefl_itp/use.

100116-10573 • FB1214R200 • Printed in U.S.A.

I.N. 770462

Copyright © 2012 by Educational Testing Service.

**Student's File Copy
Do Not Copy**



EXTENDS ITS APPRECIATION

C. Hermes Javier Ramírez García

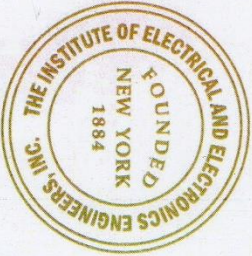
For having presented the Technical Paper:

Control de un Robot Móvil tipo Ackerman por Modelo Dinámico
in the "Mexican Humanitarian Technology Conference (MHTC) 2017"

Puebla, Mexico, March 31, 2017

Dr. German A. Muñoz-Hernandez
IEEE Puebla Section Chair

Dr. Jaime J. Cid-Monjaraz
Organizing Committee MHTC 2017 Chair



Control de un Robot Móvil Tipo Ackerman por medio del Modelo Dinámico

Ramirez García Hermes Javier, Palomino Merino Amparo, Vargas Treviño Maria Aurora Diozcora
Facultad de Ciencias de la Electrónica
Benemérita Universidad Autónoma de Puebla
Puebla, México

hermesjrg@outlook.com, palomino@ece.buap.mx, auroravargast@hotmail.com

Resumen— En este documento se presenta el control de un robot móvil tipo Ackerman, por su modelo dinámico, a diferencia de la gran mayoría de enfoques, dirigidos a partir del modelo cinemático; con el fin de llevar el robot a una posición deseada. En el modelado se establecen las restricciones a considerar para tal sistema y finalmente se presentan las pruebas realizadas mediante simulaciones en Matlab/simulink, las cuales arrojan resultados satisfactorios.

Palabras Clave—Ackerman, Cinemática, Dinámica, Control Punto a Punto

I. INTRODUCCIÓN

Desde el inicio de la revolución industrial con la creación de máquinas que ayudan a aumentar la producción y velocidad de diferentes procesos, el hombre ha buscado la forma de mejorar estas máquinas lo cual ha llevado a la construcción de robots móviles.

Con la creación de los robots móviles se empezó la investigación de métodos de control para que ejecutaran una tarea deseada, una de las más básicas es que el robot móvil llegue a un punto fijo deseado. De la investigación de cómo controlar estos robots surgen los diferentes métodos de control que hoy en día existen y se siguen mejorando. El problema radica desde la concepción física del robot, para el caso en que se desea realizar una implementación real, o desde la concepción matemática del mismo, lo cual, en cualquiera de los dos casos, lleva al problema de modelar matemáticamente el robot. En este planteamiento se deben tomar en cuenta diversas consideraciones o restricciones, en función de la descripción del modelo.

En una gran cantidad de trabajos enfocados a robots móviles [1], [2] y [3] se presenta el control de robot móviles por medio de su modelo cinemático, lo cual presenta ciertos inconvenientes cuando se requiere una mayor precisión en cuando al desempeño general del sistema para llegar a la posición deseada, ya que dicho control no proporciona una fuerza o torque directamente aplicado a los motores que mueven las ruedas. Este tipo de control sólo se puede lograr en base al modelo dinámico del robot móvil. Algunas propuestas donde involucran el modelo dinámico se pueden encontrar en [4] y [5].

El objetivo de este trabajo es el estudio y análisis del comportamiento de las variables que determinan la posición del robot móvil para comprender el posible comportamiento del robot móvil al ejecutar la tarea deseada y así evitar accidentes durante la implementación implementando un control punto a punto en el modelo dinámico del robot móvil.

II. MODELADO

Modelar un sistema en sus ecuaciones cinemáticas y dinámicas permite comprender al investigador las propiedades y comportamiento del sistema a partir de diferentes parámetros definidos.

El modelo cinemático describe el movimiento del sistema sin considerar las fuerzas externas que lo causan, generalmente se toman en cuenta las siguientes consideraciones para obtener el modelo cinemático para robots móviles con ruedas:

- El robot se mueve sobre una superficie plana.
- No existen elementos flexibles en la estructura del robot (incluyendo las ruedas).
- Las ruedas poseen un eje de direccionamiento, que siempre es perpendicular al suelo.
- Se desprecia todo tipo de fricción en los elementos móviles del robot móvil contra el suelo.

En contra posición con el modelo dinámico, el cual describe el movimiento del robot a partir de las fuerzas externas que mueven al robot, comúnmente es el torque que proporcionan los motores a las ruedas del robot móvil.

III. MODELO CINEMÁTICO

En particular, este trabajo se enfocará al robot móvil tipo Ackerman por su semejanza a la configuración de un vehículo estándar, ya que proporciona una gran estabilidad y es aplicable a la vida cotidiana.

La configuración Ackerman se conforma de 4 ruedas distribuidas en 2 ejes que poseen la misma longitud b a una

distancia l entre ellos, con las llantas delanteras indicando la dirección en la que se moverá el robot móvil (Fig. 1).

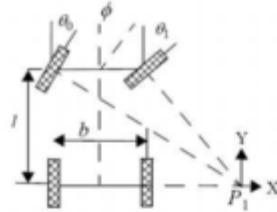


Fig. 1 Robot Móvil tipo Ackerman

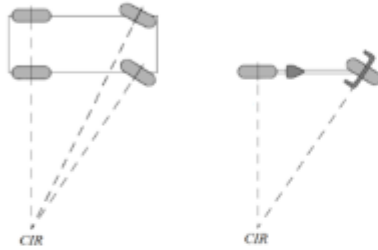


Fig. 2. Simplificación del modelo del robot móvil.

Para obtener el modelo cinemático de este tipo de robot móvil se emplean las restricciones no holonómicas del sistema, las restricciones asumen que no existe deslizamiento en las ruedas del robot móvil y las mismas se encuentran relacionadas con las velocidades del vehículo. Como resultado la ecuación general de las restricciones no holonómicas para un robot móvil tipo Ackerman queda [6]

$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) = 0 \quad (1)$$

donde \dot{x} y \dot{y} son las velocidades lineales de las llantas dentro del sistema de coordenadas (x, y) y θ es el ángulo del vehículo con respecto al eje x . Con las restricciones holonómicas establecidas se procede a modelar el robot móvil en el sistema de coordenadas (x, y) simplificando el modelo al de una bicicleta como se ilustra en la Fig. 2.

Bajo estas condiciones se plantean las coordenadas generales que permitirán describir al sistema en la Fig. 3.

Se plantea (x, y) como la posición del centro de gravedad del vehículo, θ la orientación del vehículo con respecto al eje x y α el ángulo de orientación de la rueda delantera con respecto al eje del cuerpo del robot. Sea (x_1, y_1) la posición de la rueda trasera y (x_2, y_2) la posición de la rueda delantera, definidas por (2) a (5)

$$x_1 = x - b \cos(\theta) \quad x_2 = x + a \cos(\theta) \quad (2)$$

$$y_1 = y - b \sin(\theta) \quad y_2 = y + a \sin(\theta) \quad (3)$$

$$\dot{x}_1 = \dot{x} + \dot{\theta} b \sin(\theta) \quad \dot{x}_2 = \dot{x} - \dot{\theta} a \sin(\theta) \quad (4)$$

$$\dot{y}_1 = \dot{y} - \dot{\theta} b \cos(\theta) \quad \dot{y}_2 = \dot{y} + \dot{\theta} a \cos(\theta) \quad (5)$$

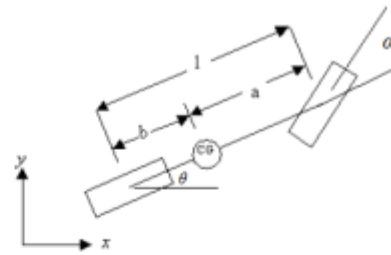


Fig. 3. Coordenadas Generales del robot móvil.

Las restricciones holonómicas para las ruedas trasera y delantera están descritas de la siguiente forma:

$$\dot{x}_1 \sin(\theta) - \dot{y}_1 \cos(\theta) = 0 \quad (6)$$

$$\dot{x}_1 \sin(\theta + \alpha) - \dot{y}_2 \cos(\theta + \alpha) = 0 \quad (7)$$

Sustituyendo la definición de (\dot{x}_1, \dot{y}_1) y (\dot{x}_2, \dot{y}_2) en (6) y (7) obtenemos:

$$\dot{x} \sin(\theta) - \dot{y} \cos(\theta) + b \dot{\theta} = 0 \quad (8)$$

$$\dot{x} \sin(\theta + \alpha) - \dot{y} \cos(\theta + \alpha) + a \dot{\theta} \cos(\theta) = 0 \quad (9)$$

Usando las coordenadas en el centro de masa del vehículo, definidas en la Fig. 3, se define el eje u a lo largo del eje central del robot móvil y el eje w perpendicular al eje u , ejemplificado en la Fig. 4, se obtienen (10) y (11).

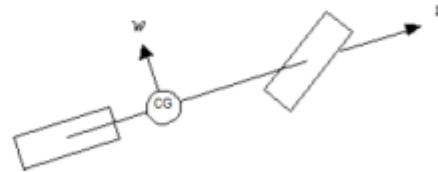


Fig. 4. Coordenadas del robot móvil.

$$\dot{x} = v_u \cos(\theta) - v_w \sin(\theta) \quad (10)$$

$$\dot{y} = v_u \sin(\theta) + v_w \cos(\theta) \quad (11)$$

donde v_u y v_w son las velocidades del centro de gravedad (CG) a lo largo de los ejes u y w respectivamente. Sustituyendo (10) y (11) en (8) y (9) se obtienen (12) y (13).

$$v_w = b \dot{\theta} \quad (12)$$

$$\dot{\theta} = \frac{v_w}{l} \tan(\alpha) \quad (13)$$

IV. MODELO DINÁMICO

Con el modelo cinemático definido, se procede a determinar el modelo dinámico usando la metodología de Newton-Euler, donde se tienen las fuerzas longitudinales y laterales ejercidas sobre las llantas, dadas por F_{ui} y F_{wi} ($i = 1$ para las llantas delanteras e $i = 2$ para las llantas traseras) respectivamente. La fuerza de propulsión, F_p , es aplicada por medio de las llantas traseras en la dirección del eje u de las ruedas, m la masa del robot móvil, J el momento de inercia y las distancias a , b , l definidas en la cinemática; la dinámica del robot móvil están dadas por (14) a (19) [7]

$$\dot{x} = v_u \cos(\theta) - v_w \sin(\theta) \quad (14)$$

$$\dot{y} = v_u \sin(\theta) + v_w \cos(\theta) \quad (15)$$

$$\dot{\theta} = \frac{v_w}{l} \tan(\alpha) \quad (16)$$

$$\dot{v}_u = v_w \Omega + \frac{F_{w1} \cos(\alpha)}{m} - \frac{F_{w1} \sin(\alpha)}{m} + \frac{F_{u1} + F_p}{m} \quad (17)$$

$$\dot{v}_w = -v_u \Omega + \frac{F_{u1} \sin(\alpha)}{m} + \frac{F_{w1} \cos(\alpha)}{m} + \frac{F_{w2}}{m} \quad (18)$$

$$\dot{\Omega} = \frac{F_{u1} a \sin(\alpha)}{j} + \frac{F_{w1} a \cos(\alpha)}{j} - \frac{F_{w2} b}{j} \quad (19)$$

Este modelo dinámico se complementa con la dinámica de la rueda de dirección la cual se denota por $\dot{\alpha} = F_s$, donde F_s es la variable de control para modificar el ángulo de orientación del robot móvil.

Considerando que no existe deslizamiento en las ruedas, el modelo dinámico del robot móvil tipo Ackerman, queda descrito por (20) a (28):

$$\dot{v}_u = A_p F_p + A_s F_s \quad (20)$$

$$v_w = \frac{v_u b}{l} \tan(\alpha) \quad (21)$$

$$\Omega = \frac{v_w}{l} \tan(\alpha) \quad (22)$$

$$\dot{\alpha} = F_s \quad (23)$$

$$\dot{x} = v_u \cos(\theta) - v_w \sin(\theta) \quad (24)$$

$$\dot{y} = v_u \sin(\theta) + v_w \cos(\theta) \quad (25)$$

$$\dot{\theta} = \Omega \quad (26)$$

$$A_p = \left(m + \left(\frac{\tan(\alpha)}{l} \right)^2 (mb^2 + j) \right)^{-1} \quad (27)$$

$$A_s = \frac{mb^2 + j}{l \cos^2(\alpha)} \Omega A_p \quad (28)$$

V. CONTROL PUNTO A PUNTO

Para llevar a cabo el control punto a punto del robot móvil se plantea como variables de entrada las coordenadas de un punto fijo al cual se desea llegar (posición deseada) y como salida las fuerzas de control F_p y F_s , para ello se plantean (29) a (32) para llevar a cabo la tarea deseada [7] [8].

El control de dirección está dado por (29) y el control de propulsión por (28).

$$F_s(t) = -k_s [\theta_{os}(t) \quad l_{os}(t) \quad \alpha_{os}(t)]^T \quad (29)$$

$$F_p(t) = F_{p1}(t) + F_{p2}(t) \quad (30)$$

$$F_{p1}(t) = A_p^{-1} (\dot{v}_{ud} - A_s F_s) \quad (31)$$

$$F_{p2}(t) = A_p^{-1} (k_{p1} v_{os}) \quad (32)$$

donde

\dot{v}_{ud} es la aceleración deseada del robot móvil

k_s es un vector de ganancias constantes positivas.

$\theta_{os}(t) = \theta(t) - \theta_d(t)$ es el error en theta

$\alpha_{os}(t) = \alpha(t) - \alpha_d(t)$ es el error de alfa

$l_{os}(t) = l(t) - l_d(t)$ es el error de posición

k_{p1} y k_{p2} son ganancias constantes positivas

para obtener los valores de θ_d , l_{os} y α_d se ocupan (33) a (36).

$$\alpha_d = k_a \left(\tan^{-1} \left(\frac{y_d - y}{x_d - x} \right) - \theta \right) \quad (33)$$

$$\theta_d = \tan^{-1} \left(\frac{y_d - y}{x_d - x} \right) \quad (34)$$

$$l_{os} = -((x - x_d) \cos(\theta_d - \theta_{ud})) + ((y - y_d) \sin(\theta_d - \theta_{ud})) \quad (35)$$

donde

k_a es una constante de ganancia positiva

y es la posición actual del robot en el eje y

x es la posición actual del robot en el eje x

y_d es la posición deseada del robot en el eje y

x_d es la posición deseada del robot en el eje x

$$\theta_{vd} = \tan^{-1} \left(\frac{\tan(\alpha_d)}{2} \right) \quad (36)$$

VI. SIMULACIÓN DEL MODELO DINÁMICO CON EL CONTROL PROPUESTO

Las simulaciones se desarrollan en el ambiente de programación de simulink de Matlab para simular el recorrido

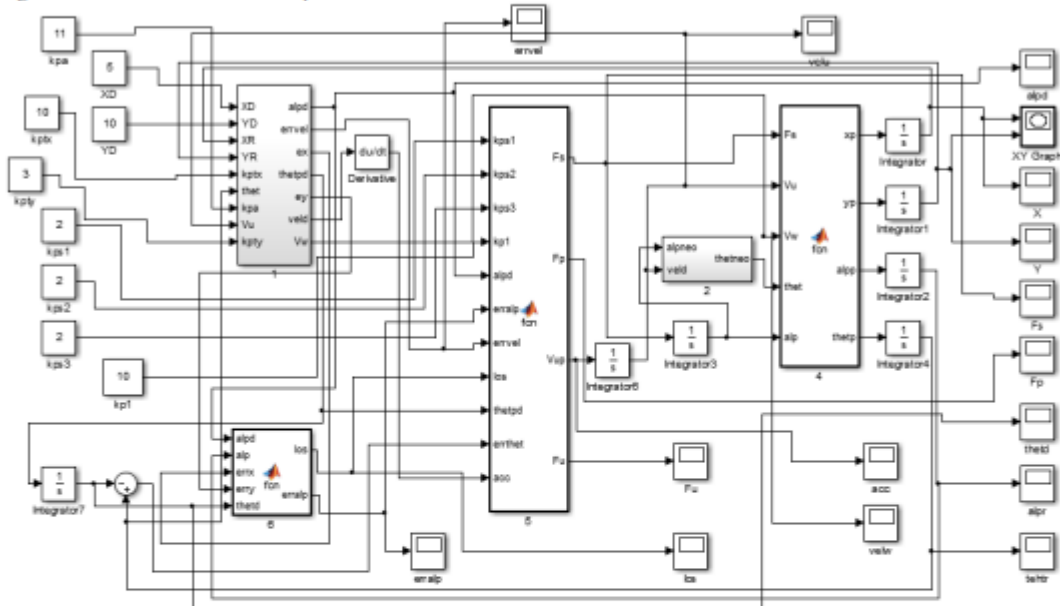


Fig. 5. Diagrama de bloques de simulación.

Para el análisis de resultados se grafica la posición del robot en un plano de coordenadas (x, y) , el comportamiento de los ejes x y y , el error en los ejes x , y y los ángulos α y θ del robot móvil.

A. Resultados de simulación

Los resultados obtenidos de la simulación se muestran en un tiempo de 5 segundos con el vector de condiciones iniciales $[x \ y \ \theta \ \alpha] = [0 \ 0 \ 0 \ 0]$ para llegar al punto fijo deseado $P = (5, 10)$, con las ganancias del control propuestas de tal forma que la trayectoria realizada del robot sea suave y en poco tiempo.

En Fig. 6 se muestra la trayectoria que siguió el robot móvil para llegar al punto deseado.

que realiza el robot móvil hacia una posición deseada dada en coordenadas (x, y) .

En base a las ecuaciones del modelo dinámico del robot móvil y del control propuesto se diseña el diagrama de bloques en simulink que se muestra en la Fig. 5.

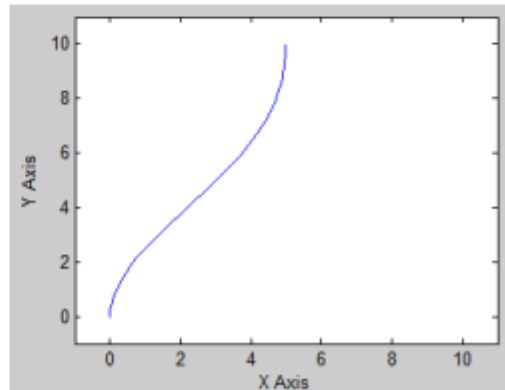


Fig. 6 Trayectoria realizada por el robot móvil a la posición deseada P.

En Figs. 7 y 8 se muestra el comportamiento de los ejes x y y y obtenidos del modelo dinámico.

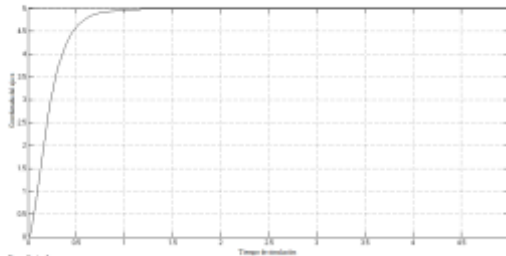


Fig. 7. Coordenada x.

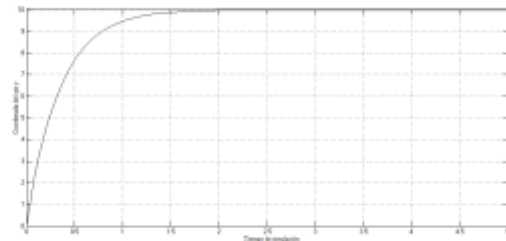


Fig. 8. Coordenada y.

En Figs. 9 y 10 se presentan las gráficas del comportamiento de los ángulos α y θ .

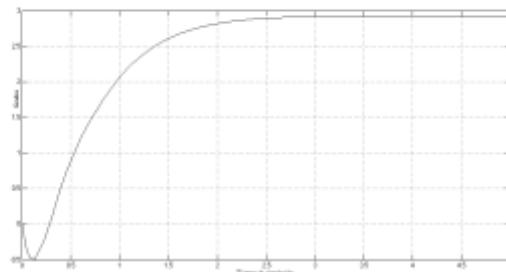


Fig. 9. Ángulo α .

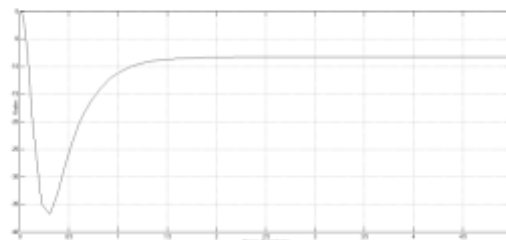


Fig. 10. Ángulo θ .

En Figs. 11 y 12 se presentan las gráficas de los errores de las coordenadas x y y.

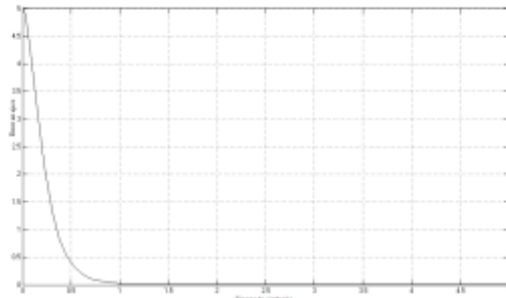


Fig. 11. Error en la coordenada x.

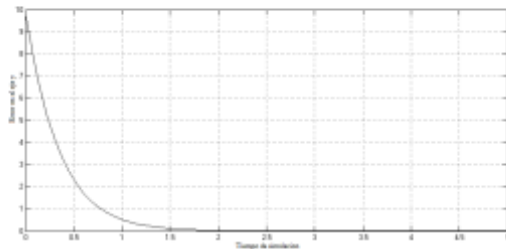


Fig. 12. Error en la coordenada y.

En la Fig. 13 se presenta la gráfica de la velocidad lineal con la que se mueve el robot móvil.

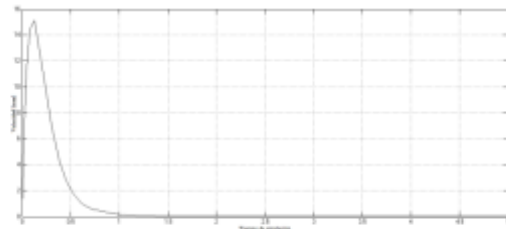


Fig. 13. Velocidad lineal

Se realizaron más simulaciones con posiciones deseadas en coordenadas $(-x, y)$, $(x, -y)$ y $(-x, -y)$ con resultados de la misma naturaleza. En la Fig. 13 se muestra la trayectoria realizada con la posición deseada $G = (-8, -7)$ usando los mismos parámetros mencionados con anterioridad.

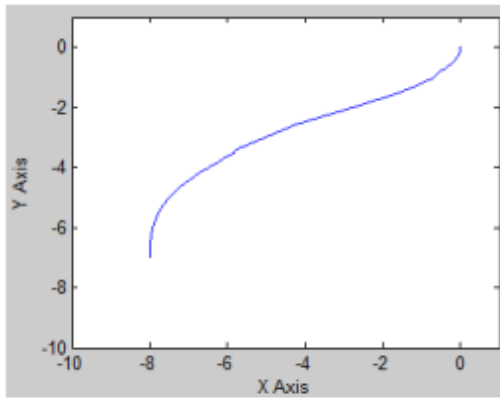


Fig. 13. Trayectoria realizada por el robot móvil a la posición deseada G.

VII. CONCLUSIONES

El desarrollo y obtención de los modelos cinemático y dinámico de un sistema ayuda a comprender, en el caso de los robot móviles, el movimiento a seguir por el sistema e identificar, con ayuda de la simulación si los datos que se están obteniendo como resultado son correctos o incorrectos con el fin de identificar los errores dentro de la simulación.

Al proponer diferentes valores en las ganancias en el control punto a punto permite analizar y entender la manera en que afectan las ganancias al modelo y así asegurar que el robot móvil realice la tarea deseada de la manera más eficaz posible. El desempeño del sistema depende de gran medida de la sintonización de las ganancias, lo cual no es una tarea fácil ya que se ven involucradas distintas ganancias en todo el proceso.

El control propuesto permite comprender de mejor forma el modo en que debe aplicarse a un modelo en la vida real con el propósito de evitar accidentes, tanto para la plataforma en la que se experimenta y el ambiente en el que se trabaja.

Se encuentra en desarrollo la implementación del control punto a punto propuesto en un robot móvil experimental [9].

REFERENCIAS

- [1] B. d'Andréa-Novel, G. Campion, G. bastin; Control of Nonholonomic Wheeled Mobile Robots by State Feedback Linearization; The International Journal of Robotics Research; 1995.
- [2] A. De Luca, G. Oriolo, C. Samson; Feedback Control of a Nonholonomic Car-like Robot; Laboratoire d'Analyse et d'Architecture des Systèmes Centre National de la Recherche Scientifique; p. 83; 1998.
- [3] Giraldo Castillo Oscar, Duran Tolentino Eyner; Localización de un Robot Móvil de Configuración Ackerman usando el Filtro de Kalman; p. 9.
- [4] Noor Hafizah Amer, Hairi Zamzuri, Khisbullah Hudha, Zulkiffli Abdul Kadir; Modelling and Control Strategies in Path Tracking Control for Autonomous Ground Vehicles: A Review of State of the Art and Challenges; p. 30; 2016.

- [5] Junmin Wang, Joe Steiber, Bapiraju Surampudi; Autonomous Ground Vehicle Control System for High-Speed and Safe Operation; p. 6; 2008.
- [6] Erick N Moret; Dynamic Modeling and Control of a Car-Like Robot; Faculty of the Virginia Polytechnic Institute and State University; p. 88; 2003.
- [7] R. M. DeSantis; Path-Tracking for Car-Like Robots with Single and Double Steering; TRANSACTIONS ON VEHICULAR TECHNOLOGY; IEEE; p. 12; 1995.
- [8] Peter Corke; Robotics, Vision and Control; Editorial Springer; p. 558; Año 2011.
- [9] Alfredo García Suarez; Control de Navegación de un Robot Móvil; Facultad de Ciencias de la Electrónica, BUAP; p. 162; 2015.

Bibliografía

- [1] Everett, H. R. «Sensors for Mobile Robots: theory and application». A. K. Peters. USA. 1995.
- [2] Ollero Baturone, Annibal, *Robótica, Manipuladores y robots Móviles*, Editorial MARCOMBO, Edición 1º, Año 2001, págs. 464.
- [3] Andrei Fenesan, Daniel Szöcs, Teodor Pana, Wen-Hua Chen; *Building an Electric Model Vehicle and Implementing an Obstacle Avoidance Algorithm*; páginas 5, 2012.
- [4] Yan Peng, Dong Qu , Yuxuan Zhong, Shaorong Xie, Jun Luo; *The Obstacle Detection and Obstacle Avoidance Algorithm Based on 2-D Lidar*; Department of Mechatronic Engineering Shanghai University, Shanghai, China; páginas 6;2015.
- [5] Luis C. Gonzalez-Sua, Olivia Barron, Rogelio Soto, Leonardo Garrido, Ivan Gonzalez, J. L. Gordillo and Alejandro Garza; *Tecnológico de Monterrey*; páginas 5; 2013.
- [6] Marco Melega, Samuel Lazarus, Al Savvaris, Antonios Tsourdos; *Algoritmo de Evasión y Sensado de Múltiples Objetivos para Obstáculos Estáticos y Dinámicos*; Department of Engineering Physics, School of Engineering, Cranfield University, Bedfordshire, UK; páginas; 2014.
- [7] S.B. Hottman, K.R. Hansen, and M. Berry; *Documento de Revisión de Detección, Censado y Evasión para Sistemas Aéreos no Tripulados*; U.S. Department of Transportation Federal Aviation Administration Air Traffic Organization Operations Planning Office of Aviation Research and Development Washington; páginas 26; 2009.
- [8] Helbert Eduardo Espitia Cuchango; *Propuesta de un Algoritmo para la Planeación de Trayectorias de Robots Móviles Empleando Campos Potenciales y Enjambres de Partículas Activas Brownianas*; Universidad nacional de Colombia, Facultad de Ingeniería; páginas 126; 2011.
- [9] Erick N Moret; *Dynamic Modeling and Control of a Car-Like Robot*; Faculty of the Virginia Polytechnic Institute and State University; p. 88; 2003.
- [10] R. M. DeSantis; *Path-Tracking for Car-Like Robots with Single and Double Steering*; TRANSACTIONS ON VEHICULAR TECHNOLOGY; IEEE; p. 12; 1995
- [11] Oussama Khatib; *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*; Stanford University; páginas 9; 1986.
- [12] Alfredo García Suarez; *Control de Navegación de un Robot Móvil*; Facultad de Ciencias de la Electrónica, BUAP; páginas 162; 2015.