

CONTROL DE DIRECCIÓN DE UN VEHÍCULO AUTÓNOMO

TESIS

Maestría en Automatización

Instituto Tecnológico y de Estudios Superiores de Monterrey

por

Gerardo Javier Palacios Garza

Diciembre del 2000

Instituto Tecnológico y de Estudios Superiores de Monterrey

CAMPUS MONTERREY

DIVISION DE GRADUADOS EN COMPUTACIÓN, INFORMACIÓN Y COMUNICACIONES

Dirección de Programas de Posgrados en Computación, Información y Comunicaciones

Los miembros del comité de tesis recomendamos que la presente tesis del Ing. Gerardo Javier Palacios Garza sea aceptada como requisito parcial para obtener el grado académico de Maestro en Ciencias, especialidad en:

Automatización: Ingeniería de Control

Comité de tesis:

Dr. José Luis Gordillo
Asesor

Dr. Rogelio Soto
Sinodal

Dr. Ricardo A. Ramírez
Sinodal

Dr. Carlos Scheel M.
Director del Programa de Graduados en
Computación, Información y Comunicaciones

Diciembre del 2000

CONTROL DE DIRECCIÓN DE UN VEHÍCULO AUTÓNOMO

por

Gerardo Javier Palacios Garza

Tesis

**Presentada al Programa de Graduados en Computación, Información y
Comunicaciones**

del

Instituto Tecnológico y de Estudios Superiores de Monterrey

Como requisito parcial para obtener el grado de

Maestro en Ciencias

Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Monterrey

Diciembre del 2000

Dedicatoria

A Dios,

A mis padres Hortensia y Enrique por su amor,

A mi novia Susana por su constante dedicación y cariño,

A mis hermanos Claudia, Enrique, Cecilia y Anayantzi por su apoyo.

Agradecimientos

Quiero expresar mi agradecimiento al Departamento de Ingeniería de Control, en especial al Dr. Narvaez, que me permitió trabajar como asistente de docencia durante el último año de mis estudios de maestría. Al Ing. Richer, por darme la flexibilidad de continuar mis estudios, mientras trabajaba en su empresa.

También quiero agradecer a Adriana Cantú, Alfredo Cruz y Claudia Sanchez y todas las personas que de alguna u otra forma me ayudaron en la realización de esta tesis. También, le agradezco al Dr. Ricardo Ramirez y al Dr. Rogelio Soto, por su tiempo y apoyo, y en especial al Dr. Gordillo, por todo el tiempo que ha dedicado en mejorar esta tesis y por todo lo que he aprendido mientras trabajé con él.

Resumen

La presente tesis propone el desarrollo de un vehículo que sigue una trayectoria dada, controlando su dirección y su velocidad. El vehículo se clasifica como semiautónomo, ya que el cálculo de la trayectoria la realiza el usuario. El controlador que aquí se desarrolla se encarga de convertir esa trayectoria, en movimientos reales del vehículo.

La cinemática del vehículo es igual a la de un carro común; se tienen dos ejes, en el eje delantero están las llantas que giran para direccionar el vehículo, mientras que la dirección de las llantas en el eje trasero está fija. La trayectoria se define por medio de puntos en el plano, en pares de coordenadas. La parte central del eje trasero del vehículo es el punto que debe de pasar por los puntos para formar la trayectoria. El control se realiza con un controlador en cascada, cuyo controlador maestro se encarga de controlar la posición del vehículo en el plano, verificando que el vehículo pase por los puntos. Los controladores esclavos controlan la dirección de las llantas delanteras y la velocidad lineal del vehículo. El control maestro va cambiando el valor de referencia en el controlador esclavo de dirección, de tal forma que el vehículo se dirija hacia el siguiente punto, al llegar al último punto de la trayectoria frena el vehículo.

La implementación se hizo en un vehículo comercial de control remoto que ya contaba con motores de corriente directa para la dirección y la tracción. El vehículo se instrumentó con los circuitos necesarios para controlar los motores y para leer las señales de los sensores desde un microcontrolador. El control de los motores de corriente directa utilizó puentes H y la señal de control se envía mediante modulación de ancho de pulso (PWM), para regular la potencia entregada al motor. El sensor de dirección es un potenciómetro unido a la flecha del motor que mueve la dirección, mientras que el sensor de avance es un codificador óptico incremental conectado físicamente con la llanta trasera izquierda.

Para el cálculo de la posición del vehículo en el plano, se estima la posición y la inclinación del cuerpo del vehículo. La estimación se realizó utilizando únicamente los sensores de dirección y de avance; por lo cuál el lazo de control de la trayectoria se cierra interiormente al vehículo y no al exterior.

Contenido

Índice de Figuras	xv
Índice de tablas	xix
Capítulo 1 Introducción	1
Capítulo 2 Modelo del sistema	7
2.1 Comportamiento mecánico	7
2.1.1 Geometría de direccionamiento Ackerman	8
2.2 Modelo cinemático	10
2.2.1 Modelo cinemático del carro común	10
2.2.2 Histéresis de la dirección	12
2.3 Comportamiento Eléctrico	14
2.3.1 Motor de corriente directa	14
2.3.2 Modelo del sistema de dirección	16
2.3.3 Modelo del sistema de velocidad	17
2.4 Lista de términos	18
Capítulo 3 Arquitectura para el control del vehículo	21
3.1 Estructura y módulos de la arquitectura	21
3.2 Instrumentación del vehículo	23
3.2.1 Etapa de potencia en tarjetas de control	24
3.2.2 Sensado y acondicionamiento de dirección	25
3.2.3 Sensado y acondicionamiento de velocidad	27
3.3 Unidad de procesamiento	29
Capítulo 4 Sistema jerárquico de control	31
4.1 Control de dirección	33

4.1.1 Desempeño del controlador	34
4.1.2 Ganancia preprogramada	36
4.2 Control a un punto	37
4.3 Control de velocidad.....	40
4.4 Estimación de parámetros (x, y, θ)	41
4.4.1 Aproximaciones para su implementación	43
4.4.2 Comparación de los métodos de aproximación.....	46
4.5 Control de trayectoria	47
4.5.1 Puntos objetivo inalcanzables	48
4.5.2 Mejoras al algoritmo de control de trayectoria.....	50
Capítulo 5 Implementación y experimentos	55
5.1 Modelo del sistema.....	55
5.2 Arquitectura para el control	57
5.2.1 Tarjeta para el control de dirección.....	57
5.2.2 Tarjeta para el control de velocidad	58
5.2.3 Unidad de procesamiento.....	59
5.2.5 Comunicación serial	60
5.3 Controladores	61
5.3.1 Lazo de control de dirección.....	61
5.3.2 Lazo de control de velocidad	62
5.3.3 Estimación de parámetros (x, y, θ)	64
5.3.4 Lazo de control a un punto.....	66
5.3.5 Lazo de control de trayectoria.....	66
Capítulo 6 Conclusiones	73
Apéndice A Programa de simulación	77
A.1 Comparar las dos estrategias de control de trayectoria	78
A.2 Cambiar ganancia en el controlador a un punto	79
A.3 Efecto de cambiar la longitud del vehículo	81
Apéndice B Programa del microcontrolador AVR	83
B.1 Manejo de variables en el microcontrolador	83
Apéndice C Diagramas eléctricos	87
Apéndice D Tablas para el cálculo de funciones trigonométricas	89

Referencias

91

Vita

93

Índice de Figuras

Figura 1.1: Modelo cinemático del carro común. Se localiza en (x,y) del plano con θ como la inclinación del carro y ϕ como el ángulo de direccionamiento	2
Figura 1.2: Diagrama a bloques de la arquitectura de control propuesta	3
Figura 1.3: Diagrama a bloques del lazo de control propuesto.....	4
Figura 2.1: Un ejemplo de giro con geometría Ackerman. La llanta de adentro gira más que la de afuera del círculo, para que el centro de rotación coincida en un punto	8
Figura 2.2: Forma de lograr la Geometría Ackerman. El ángulo que forman las conexiones en la barra coinciden en un punto localizado en el centro del eje trasero.....	9
Figura 2.3: Mecanismo de direccionamiento del vehículo. Se presentan además las conexiones mecánicas que cumplen con la geometría de Ackerman	9
Figura 2.4: Modelo cinemático del carro común. Se localiza en (x,y) del plano con θ como la inclinación del carro y ϕ como el ángulo de direccionamiento	11
Figura 2.5: Prueba de histéresis en la dirección de la llanta izquierda. El ángulo de giro real, contra la lectura en el convertidor.....	13
Figura 2.6: Prueba de histéresis en la dirección de la llanta derecha. El ángulo de giro real, contra la lectura en el convertidor.....	13
Figura 2.7: Circuito equivalente del motor de CD para obtención del modelo.....	15
Figura 2.8: Circuito equivalente del motor de CD con la carga y sensor de posición, para el sistema de dirección.....	16
Figura 3.1: Diagrama a bloques de los módulos que componen la arquitectura de control en términos de electrónica y flujo de señales del vehículo	22
Figura 3.2: Los dos estados posibles en el <i>punte H</i> . Cuando los contactos $S1$ y $S4$ se encuentran cerrados el motor gira en un sentido; mientras que con $S2$ y $S3$ cerrados, el giro es en sentido contrario.....	24

Figura 3.3. Ejemplo de señales <i>PWM</i> . La señal permanece en estado alto un 25%, 50% y 75% del tiempo, entregando ese mismo porcentaje de voltaje en promedio	25
Figura 3.4: Ejemplo de codificador óptico absoluto de tres bits en código binario. El ángulo actual se obtiene con una resolución de 45 grados.....	26
Figura 3.5: Configuración para medir ángulo con un potenciómetro. El voltaje en la salida depende directamente del ángulo que forma el devanado central.	27
Figura 3.6: Codificador óptico incremental, con dos señales desfasadas, para obtener la dirección del giro. Durante el movimiento a favor de las manecillas del reloj, la señal del sensor B adelanta a la de A, en contra de la manecillas del reloj, la señal del sensor A adelanta a la de B.....	28
Figura 4.1: Diagrama a bloques del sistema de control propuesto.....	32
Figura 4.2: Lazo de control digital para la dirección. $G_c(s)$ es la función de transferencia del controlador y $G(s)$ es el modelo del sistema	33
Figura 4.3: Pruebas escalón en forma ascendente al controlador PD de la dirección. En la zona de operación de abajo se observa una ganancia grande, mientras que arriba se observa muy poca ganancia.....	34
Figura 4.4: Pruebas escalón en forma descendente al controlador PD de la dirección. En la zona de operación de arriba se observa una ganancia grande, mientras que abajo se observa muy poca ganancia.....	35
Figura 4.5: Mecanismo equivalente al mecanismo de dirección. La pieza que une la flecha del motor con la pieza que mueve las llantas, actúa como un resorte en los dos extremos de los topes mecánicos	35
Figura 4.6: Pruebas escalón en el control de dirección utilizando un controlador PD con ganancia preprogramada. El desempeño mejora en toda la zona de operación y en los dos sentidos de giro (ascendente y descendente).....	36
Figura 4.7: Seguimiento de un punto objetivo F. La variable controlada es θ , la variable manipulada es ϕ , y el error es θ_e . Si el error es cero, el vehículo se dirige directo al punto.....	37
Figura 4.8: Curvas generadas con diferentes sintonías del controlador para el seguimiento a un punto.....	39
Figura 4.9: Curvas generadas con diferentes distancias de muestreo, en el controlador para el seguimiento a un punto.....	40
Figura 4.10: Lazo de control digital para la velocidad; $G_c(s)$ es la función de transferencia del controlador y $G_v(s)$ es el modelo del sistema	40

Figura 4.11: Diagrama geométrico para el cálculo de Δx , Δy , $\Delta\theta$. Se ilustran dos instantes del muestreo: el instante k y el instante $k+1$; se supone que entre esos dos muestreos, el giro es constante, por ello no hubo cambio en ϕ	42
Figura 4.12: Aproximación por el Método 1. Se utiliza un vector con inclinación θ para el cálculo de Δx y Δy	43
Figura 4.13: Aproximación por el Método 2. Se utiliza un vector con inclinación $\theta + \Delta\theta$ para el cálculo de Δx y Δy	44
Figura 4.14: Aproximación por el método 3. Se utiliza un vector con una inclinación $\theta + \Delta\theta/2$ para el cálculo de Δx y Δy . En la gráfica se muestran la posición real y la aproximación	45
Figura 4.15: Prueba en el simulador con el Método 2 y Método 3 de estimación de la posición; utilizando la distancia d de una vuelta de llanta	46
Figura 4.16: Prueba en el simulador con el Método 2 y el Método 3 de estimación de la posición; utilizando la distancia d de 1/3 de vuelta de llanta	47
Figura 4.17: El vehículo no llega con el algoritmo de control propuesto, a un punto objetivo dentro de la zona circular. El radio del círculo está dado por R_{min} que se presenta al tener ϕ_{max}	48
Figura 4.18: Caso 1, el cálculo de la distancia, determina que el punto está en la zona inalcanzable. Caso 2, el cálculo de la distancia, determina que el punto es alcanzable	49
Figura 4.19: Ejemplo de seguimiento a un punto objetivo inalcanzable usando la primer estrategia. Al inicio el vehículo se aleja del punto, hasta el momento en que el punto sale de la zona inalcanzable, definida por los círculos	51
Figura 4.20: Ejemplo de seguimiento a un punto objetivo inalcanzable usando la segunda estrategia.....	52
Figura 4.21: Comparación del seguimiento de una trayectoria usando las dos estrategias. Primer estrategia: El vehículo pasa exactamente por los puntos, sin importar la curva que se genera. Segunda estrategia: La curva es más suave, aunque no pasa exactamente por los puntos	53
Figura 5.1: Fotografía del vehículo que se utilizó para hacer las pruebas	55
Figura 5.2: Esquema del vehículo con las medidas más importantes. Se muestra la distancia entre ejes de las llantas L , como se definió en el modelo cinemático	56
Figura 5.3: Ejemplo de señal PWM utilizando un contador y un registro de comparación ..	59
Figura 5.4: Diagrama de flujo programa principal implementado en el microcontrolador ...	61

Figura 5.5: Gráfica de desempeño del controlador de velocidad. La gráfica se obtuvo con el vehículo estacionado y con las llantas al aire.	63
Figura 5.6: Gráfica que relaciona ϕ con $\Delta\theta$, los datos fueron tomados de forma experimental.	66
Figura 5.7: Seguimiento de trayectoria definida por dos puntos. Al ir el vehículo a baja velocidad las vueltas son más cerradas, al ir a alta velocidad las vueltas son más abiertas.	68
Figura 5.8: Recorrido del vehículo a un punto. Se observa como el vehículo llega al punto, con un error de 25 cm hacia la izquierda	69
Figura 5.9: Recorrido del vehículo con una trayectoria de dos puntos. Se observa como pasa por el primer punto con un error de 5 cm, mientras que queda a 40 cm del segundo punto	70
Figura 5.10: Recorrido del vehículo con una trayectoria de tres puntos. Se observa como pasa por el primer punto con un mínimo error, mientras que por el segundo con un error de aproximadamente 35 cm. El tercer punto es el que tiene más error, ya que el vehículo queda a 75 cm del punto	71
Figura A.1: Pantalla principal del simulador. Presenta el plano donde se mueve el vehículo y todos los controles para hacer las simulaciones	77
Figura A.2: Dibujar trayectoria en el plano	78
Figura A.3: Ejecutar la trayectoria usando cada una de las dos estrategias	79
Figura A.4: Ejecución de trayectoria con una ganancia de 0.60 en el controlador	80
Figura A.5: Ejecución de la trayectoria con una ganancia de 3.0 en el controlador	80
Figura A.6: Se observa el efecto de cambiar la longitud del vehículo. Un vehículo pequeño puede dar vueltas más cerradas que uno más largo. El radio de giro depende de la longitud del vehículo, como se explicó en el Capítulo 2.	81
Figura B.1: Representación de números de 16 bits. El byte más significativo representa el valor entero, y el menos significativo los decimales	84
Figura C.1: Diagrama eléctrico del <i>punteo H</i> para el control de dirección	87
Figura C.2: Diagrama del circuito de conversión, para el sensor de dirección	87
Figura D.1: Representación de una variable fraccional usando 8 bits	89

Índice de tablas

Tabla 5.1: Caracterización de los motores de corriente directa en el vehículo	56
Tabla 5.2: Características estáticas del sensor de dirección.....	56
Tabla 5.3: Rango de operación del convertidor de 8 <i>bits</i> para lectura de la dirección	58
Tabla 5.4: Pruebas en campo con el vehículo, para obtener la relación entre ϕ y $\Delta\theta$	65
Tabla B.1: Lista de variables más importantes utilizadas en el programa, el espacio asignado de memoria, los valores posibles junto con la resolución y la descripción.....	85

Capítulo 1

Introducción

El trabajo más adecuado de los robots es ayudar, a los humanos, para realizar tareas repetitivas o que impliquen un esfuerzo físico muy grande; tareas en donde las condiciones de operación son desagradables e incluso insalubres. Sin embargo, la mayoría de la investigación actual en robótica móvil, se realiza con robots, que por su mismo diseño mecánico, operan bajo condiciones muy limitadas. Es cierto que estos robots cuentan con muchas funciones, sensores de todo tipo y una arquitectura computacional muy avanzada que merecen estudio; pero al analizar su real alcance, encontramos que están destinados a trabajar en laboratorios y difícilmente podrían realizar tareas en un medio ambiente que no esté acondicionado.

Por otro lado, se tiene el desarrollo tecnológico de los automóviles, y otros vehículos de trabajo, que por más de 100 años han servido al hombre para un sinnúmero de tareas y se han adecuado de muchas formas para manipular herramientas útiles en trabajos rudos, tales como la exploración petrolera y la minería. Si se lograra automatizar estos vehículos, de tal forma que sean autónomos en su nivel de aplicación, se tendrían robots más útiles y capaces de trabajar en condiciones severas.

Según el almanaque mundial, la producción actual de vehículos de pasajeros, sobrepasa los 35 millones anualmente; nada más en México, la producción actual es de casi un millón anual. Si se aprovecha esta infraestructura, en cuanto a diseño mecánico y producción, las aplicaciones al automatizar estos vehículos, van desde la transportación de personas, la transportación de material, el trabajo en la industria de la construcción, agrícola, minera, petrolera, etc. Estos vehículos también serían útiles en la exploración terrestre, para acceder en lugares inaccesibles o tóxicos, e incluso, para la exploración de otros planetas.

Actualmente se diferencia el trabajo de investigación realizado con “robots móviles” al realizado con “vehículos autónomos”; puesto que en este último caso se adecúa y automatiza un vehículo, inicialmente diseñado para ser conducido por un humano, como un robot capaz de realizar, de forma autónoma, tareas rudas acordes con su definición original. El controlador de un *vehículo autónomo*, requiere sustituir al operador humano en la ejecución de la tarea.

Se pretende generar un método para automatizar un vehículo y convertirlo en un vehículo que realice de forma autónoma tareas especializadas. Para ello, el vehículo planifica los movimientos para entrar y salir del sitio de operación, navegar con sensores dentro del espacio de trabajo, para finalmente posicionarse y manejar las herramientas.

El objetivo de esta tesis, consiste en el desarrollo de un controlador, que traduzca las trayectorias calculadas por el planificador, en movimientos reales del vehículo, controlando su dirección y velocidad.

La metodología de desarrollo, se separa en tres partes: primero se obtiene el modelo del vehículo, después se propone la arquitectura de control, sobre la cuál finalmente, se implementan los algoritmos de control. Para validar dicha metodología, se propone la adecuación de un vehículo, el cuál originalmente era un carro de control remoto, para transformarlo en un *vehículo autónomo*.

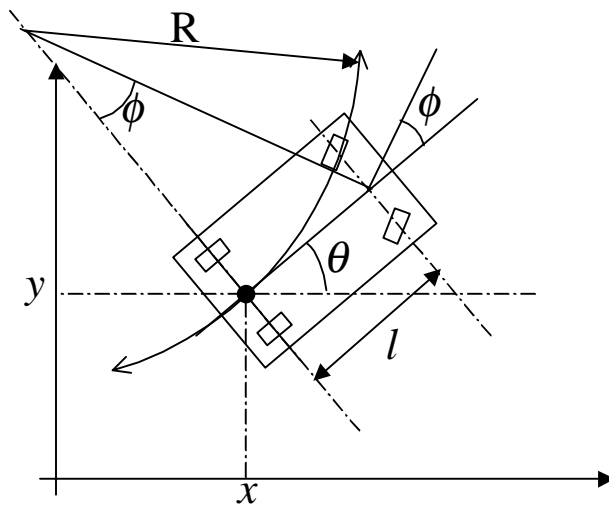


Figura 1.1: Modelo cinemático del carro común. Se localiza en (x,y) del plano con θ como la inclinación del carro y ϕ como el ángulo de direccionamiento

El modelo cinemático del vehículo, se obtiene de la configuración de los elementos mecánicos que le dan la capacidad de navegación. El vehículo se compone del cuerpo principal, unido por dos ejes con dos llantas cada uno. La dirección de las llantas, en el eje trasero, están fijas, mientras que las del eje delantero se inclinan para cambiar la dirección del movimiento. Se trata de un sistema no-holonómico, porque no se puede cambiar la dirección del cuerpo del vehículo, sin cambiar también su posición. En la figura 1.1 se presenta gráficamente la configuración del vehículo que se utiliza a lo largo de este trabajo de tesis.

El espacio de trabajo del vehículo se define por un plano, en donde se especifican las trayectorias, por medio de una serie de puntos con coordenadas (x,y) . El controlador recibe la lista de puntos que definen la trayectoria y manipula los actuadores necesarios para mover

el vehículo, de tal forma que pase por cada uno de los puntos de la trayectoria. La arquitectura de control se basa en la estrategia de control cascada, donde se tiene un controlador maestro, en este caso el que sigue la trayectoria, y dos esclavos, para el control de la velocidad y el volante de dirección.

El control, para el seguimiento de trayectorias de sistemas holonómicos, es más sencillo que para sistemas no-holonómicos, ya que para los primeros, la posibilidad de girar sobre su propio eje permite que cualquier trayectoria sea alcanzable; mientras que para los segundos, las restricciones cinemáticas que presentan, hacen que una trayectoria no sea necesariamente alcanzable.

El modelo del vehículo también incluye los modelos dinámicos de los sistemas de velocidad y dirección, individualmente. El vehículo que se automatizó, utiliza motores de corriente directa, como actuadores, en los mecanismos de velocidad y dirección.

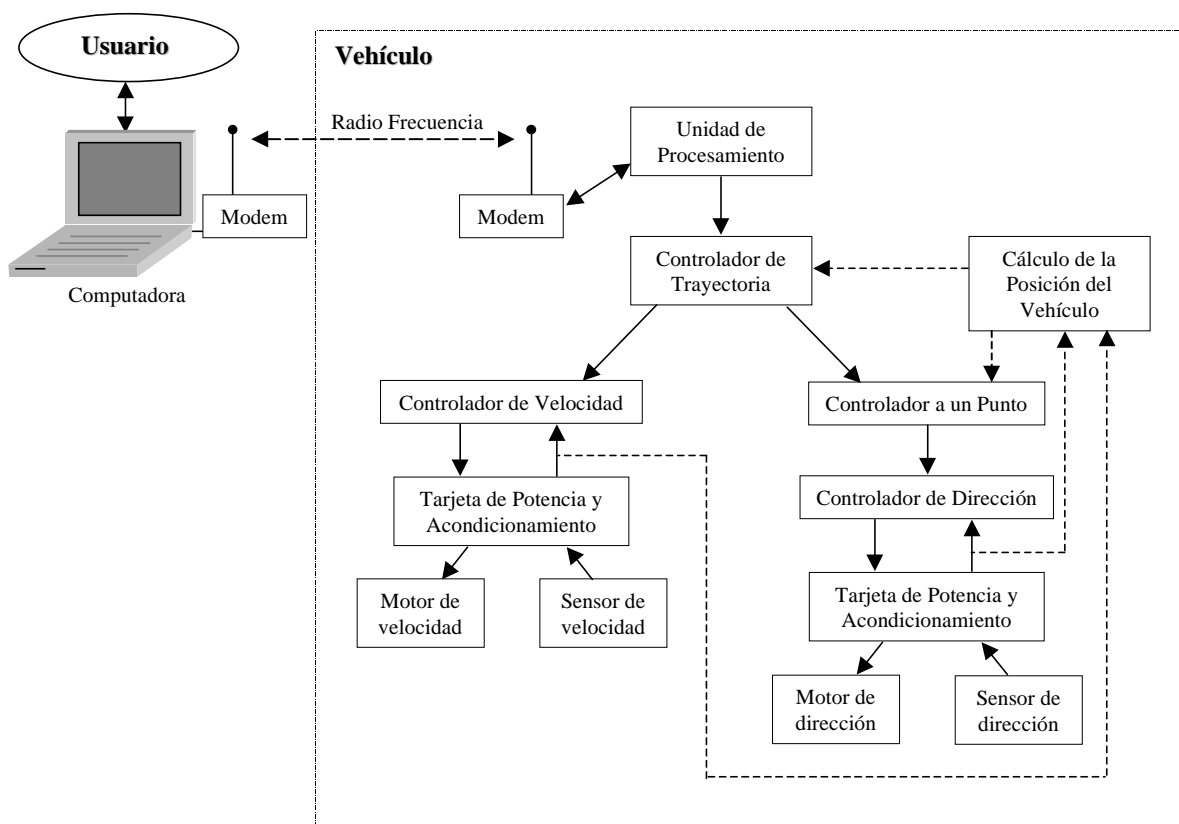


Figura 1.2: Diagrama a bloques de la arquitectura de control propuesta

La arquitectura de control define los elementos que se requieren para controlar el vehículo; tales elementos se componen de actuadores, sensores, tarjetas de potencia, tarjetas de acondicionamiento de las señales de los sensores, así como la unidad de procesamiento. La arquitectura también define la estructura del controlador; en este caso se utiliza la estrategia de control en cascada; por lo tanto, se separa el control de velocidad del control de dirección. La figura 1.2 muestra la arquitectura de control propuesta.

Los algoritmos de control dan funcionalidad a la arquitectura de control. El lazo de control propuesto para el seguimiento de trayectorias, se compone por varios controladores en configuración de cascada. El controlador de trayectoria es el controlador principal; se encarga de verificar la posición actual del vehículo para que siga la trayectoria; recibe como entrada una serie de puntos que definen la trayectoria. Conforme avanza, el vehículo toma cada uno de los puntos. La salida de este controlador son las coordenadas del siguiente punto, a donde debe avanzar el vehículo. El controlador a un punto recibe las coordenadas del siguiente punto y calcula la inclinación de las llantas, para orientar el vehículo en dirección hacia dicho punto. El controlador de dirección, recibe la inclinación en las llantas, calculada por el controlador a un punto, y manipula el motor en el mecanismo de dirección para lograr la inclinación deseada. Mientras que el controlador de velocidad lleva y mantiene el vehículo a una velocidad fija, durante todo el recorrido. La figura 1.3 muestra el diagrama a bloques del lazo de control propuesto.

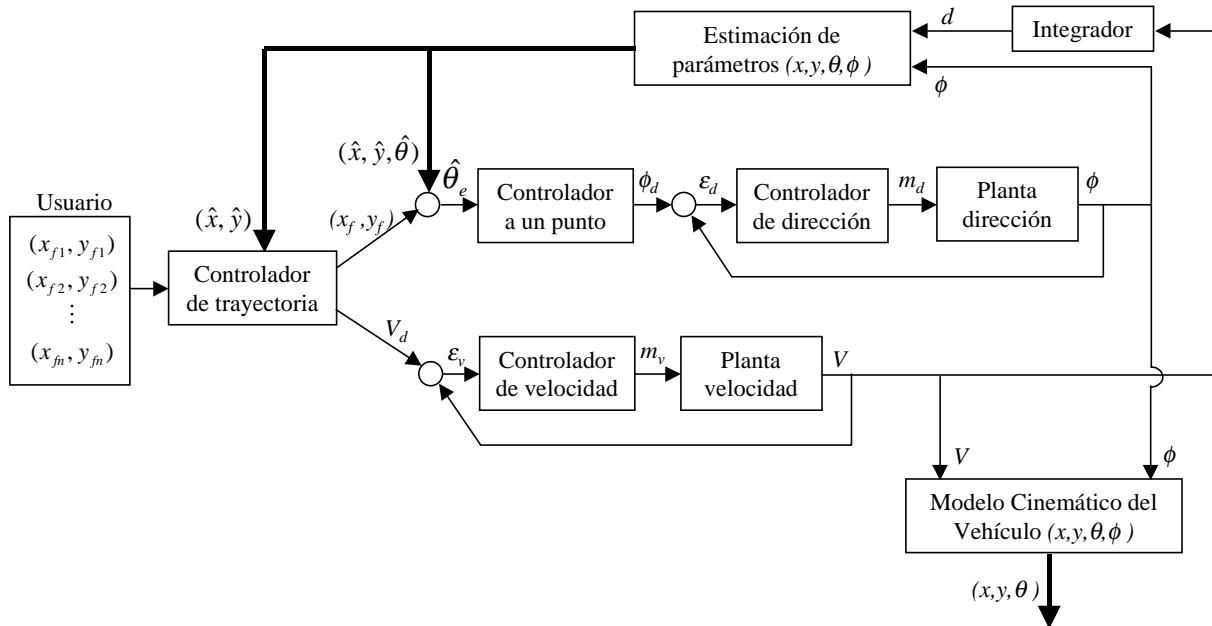


Figura 1.3: Diagrama a bloques del lazo de control propuesto

En la figura se observa que, además de los controladores, existe un bloque en donde se estiman los parámetros que definen la posición y orientación del vehículo. Este bloque es necesario, ya que no se cuenta con un sensor que proporcione directamente la posición y orientación del vehículo. La estimación de la posición del vehículo se basa únicamente en el sensor de avance y en el sensor en el mecanismo de dirección; por lo tanto, si el vehículo derrapa, el algoritmo que calcula la posición no lo detecta y como resultado se obtiene una posición errónea. El desarrollo se limitará para el seguimiento de trayectorias libres de obstáculos, donde además el contacto es puramente rotacional entre las llantas y el piso.

La arquitectura de control es modular; siendo así, se podrían cambiar los algoritmos contenidos en los bloques de control por otro que tuvieran las mismas entradas y salidas. Por ejemplo, el módulo que estima la posición del vehículo se podría cambiar por un sensor que

entregara la posición real del vehículo, directamente.

El área de investigación en vehículos autónomos es muy extensa. Este trabajo ataca el problema de actuación; el desarrollo se podría utilizar en construir un vehículo con un mayor grado de autonomía, si se le agregaran los algoritmos para el cálculo de trayectorias. El controlador que aquí se desarrolla, se puede utilizar con cualquier vehículo semejante a un carro común, independientemente de la tarea que vaya a desarrollar. Además, la modularidad de la arquitectura de control, le da universalidad en cuanto al tipo de vehículo que se quiera controlar y el medio ambiente donde se requiera trabajar.

La arquitectura de control jerárquica, permite descomponer un problema complejo, en subproblemas más sencillos de resolver.

Para describir la tesis, el Capítulo 2 analiza los componentes mecánicos y eléctricos para obtener el modelo del sistema que se quiere controlar. El Capítulo 3 propone una arquitectura de control y se analizan los elementos que la componen. El Capítulo 4 presenta una estructura de control y se analizan los algoritmos de control. El Capítulo 5 trata sobre la implementación del controlador en el vehículo. El Capítulo 6 presenta las Conclusiones. Así mismo el Apéndice A describe el programa computacional desarrollado para hacer las simulaciones. El Apéndice B describe el programa desarrollado para el microcontrolador. El Apéndice C presenta los diagramas eléctricos. Finalmente, el apéndice D explica las tablas utilizadas para los cálculos trigonométricos en el microcontrolador.

Capítulo 2

Modelo del sistema

Se pretende controlar un vehículo a lo largo de una trayectoria, definida dentro del espacio de trabajo. Para lograr el objetivo, primeramente se define el comportamiento cinemático del vehículo, enseguida se obtiene un modelo que describe el movimiento del vehículo en el plano, dependiendo de las entradas. Cuando se cuenta con más de una entrada que afecta una o más salidas, se dice que el sistema es multivariable. En este caso, como se trata de un vehículo común, las entradas, son las velocidades de desplazamiento y dirección, las salidas, son la posición e inclinación del vehículo en el plano.

La característica primordial de los vehículos que por su configuración mecánica se asemejan a la de un carro común, es que no pueden desplazarse libremente en el plano, es decir, no puede rotar sobre su propio eje o desplazarse hacia los lados, únicamente puede avanzar o retroceder con cierta dirección, la cuál, a su vez, está limitada con un tope mecánico; sin embargo, puede tomar cualquier posición y orientación en el plano. A esta clase de vehículos se les denomina no-holonómicos.

La configuración de los elementos mecánicos, definen el modelo cinemático. En este capítulo se obtendrán las ecuaciones cinemáticas de un vehículo tipo carro común no-holonómico. También se explicará la geometría de direccionamiento Ackerman, la cuál utilizan la mayoría de los vehículos de este tipo.

En un lazo típico de control, el modelo del sistema incluye también, el modelo del actuador. El vehículo utiliza motores de corriente directa para actuar sobre los elementos mecánicos de la tracción y dirección; al final del capítulo se obtendrá el modelo del sistema de tracción y dirección incluyendo los motores de corriente directa.

2.1 Comportamiento mecánico

Los elementos mecánicos, definen el comportamiento cinemático del vehículo. Con base en el modelo, se determina el número de actuadores requeridos para seguir una trayectoria, así como las variables necesarias a medir. Las ecuaciones utilizadas para el cálculo de la

posición del vehículo en el plano, dependen directamente del modelo cinemático.

En esta sección se analiza el comportamiento mecánico del vehículo. La geometría de direccionamiento de Ackerman, permite que los elementos en el mecanismo de la dirección, se diseñen de tal forma que al maniobrar el vehículo, pueda desplazarse sin que exista un deslizamiento lateral en el contacto con el piso de las cuatro llantas.

2.1.1 Geometría de direccionamiento Ackerman

Al girar las llantas delanteras del vehículo fuera de su posición de línea recta, el diseño mecánico del actuador determina que las llantas permanecen paralelas; o bien, que una llanta gire más que la otra, procurando que no exista deslizamiento en el contacto de las llantas con el piso.

Para bajas aceleraciones laterales (vehículos de baja velocidad) es común utilizar la geometría Ackerman [Milliken, Milliken 1995]. Como se muestra en la figura 2.1, esta geometría permite que todas las llantas giren sin deslizamientos, ya que las llantas se posicionan de modo que su centro de rotación coincide en un mismo punto. Se observa como la llanta interna gira más que la llanta de afuera del círculo que describe. La geometría de Ackerman permite simplificar el modelo del carro, al modelo de una bicicleta, lo cuál ayudará a simplificar las ecuaciones del modelo cinemático. El ángulo de dirección, que se toma en el modelo de la bicicleta, es el promedio del ángulo de las dos llantas, el cuál se obtiene al trazar una línea recta del centro de rotación al centro del eje delantero.

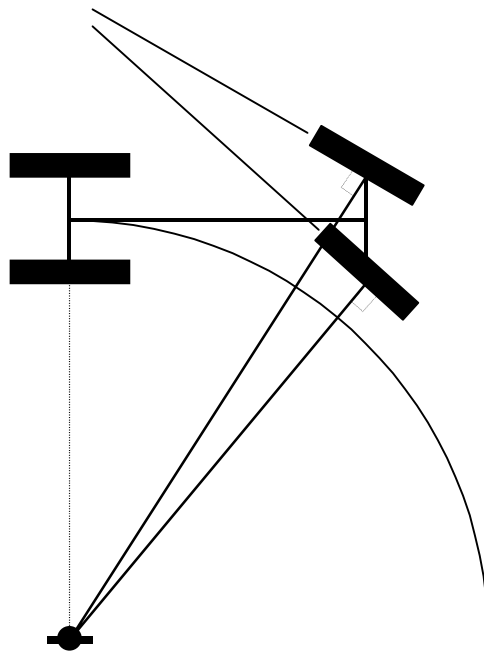


Figura 2.1: Un ejemplo de giro con geometría Ackerman. La llanta de adentro gira más que la de afuera del círculo, para que el centro de rotación coincida en un punto

Se obtiene una aproximación a esta geometría, si el mecanismo cumple con ciertas restricciones, según se muestra en la figura 2.2. El ángulo que forman los brazos que hacen girar las llantas, coinciden en el centro del eje trasero.

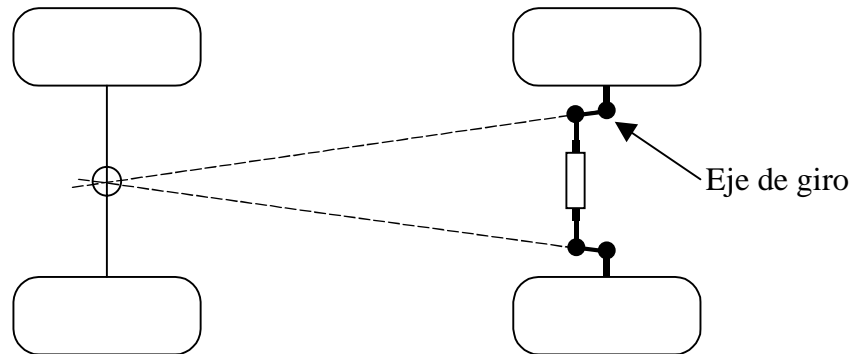


Figura 2.2: Forma de lograr la Geometría Ackerman. El ángulo que forman las conexiones en la barra coinciden en un punto localizado en el centro del eje trasero

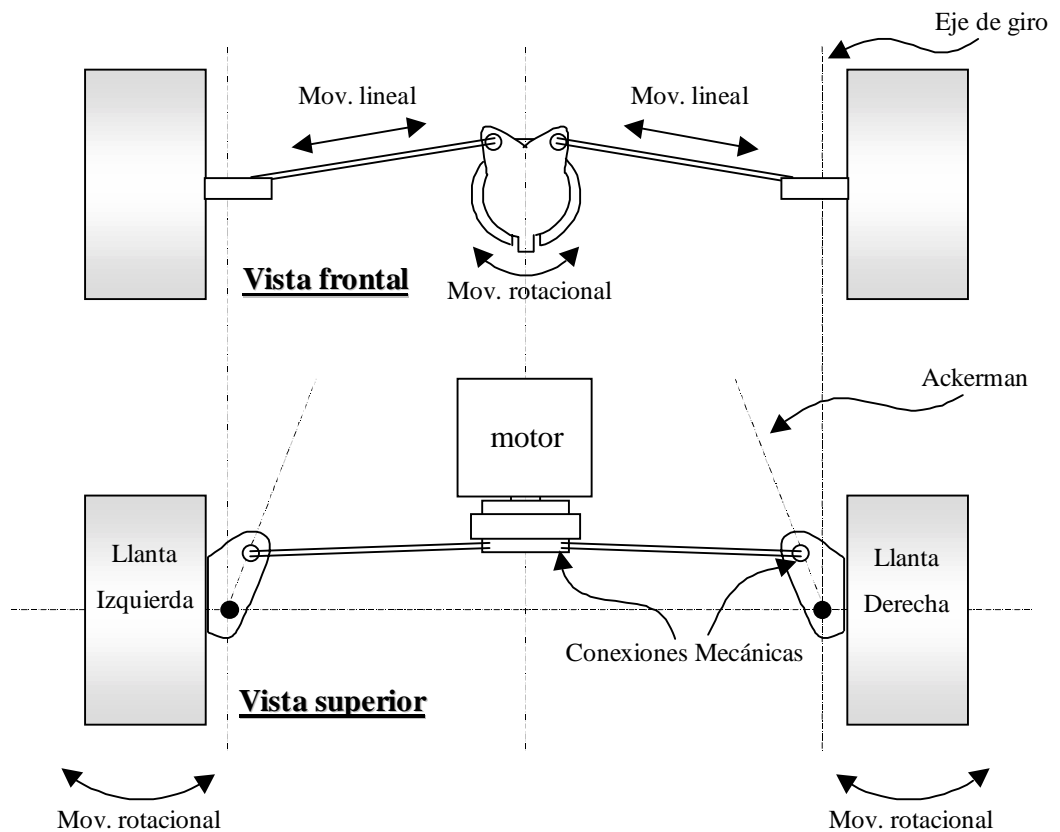


Figura 2.3: Mecanismo de direccionamiento del vehículo. Se presentan además las conexiones mecánicas que cumplen con la geometría de Ackerman

El vehículo modelado utiliza la geometría de Ackerman, aunque en el lugar del diferencial, se encuentra un motor de corriente directa. Este motor está conectado a un mecanismo que convierte el movimiento rotacional de la flecha a un movimiento lineal en las varillas, las cuáles, a su vez, hacen girar las llantas para dar sentido a la dirección (ver figura 2.3).

2.2 Modelo cinemático

Las restricciones cinemáticas establecen que un objeto no puede trasladarse y rotar libremente en el espacio de trabajo. El movimiento de un robot móvil, tipo carro, es un ejemplo de estas restricciones. Un vehículo que se mueve hacia adelante y hacia atrás, puede cambiar de dirección, pero no se desplaza hacia los lados, y sus vueltas son usualmente limitadas por topes mecánicos en el mecanismo de dirección [Latombe 1991].

Las restricciones que afectan el movimiento de un vehículo, con las características de un carro común, se clasifican como no-holonómicas, porque no reducen la dimensión del espacio de configuración. El vehículo puede tomar cualquier posición y orientación en el plano; aunque las restricciones limitan el espacio de velocidad, ya que no puede desplazarse hacia los lados, únicamente hacia adelante y hacia atrás. Por ello, para alcanzar una posición contigua, un carro debe realizar maniobras de dirección, avance y retroceso. Tal es el caso del estacionamiento en espacio lateral (diente de engrane). La planeación y ejecución de trayectorias con restricciones no-holonómicas es un área de investigación relativamente nueva y tiene un gran interés dada la aplicación a robots móviles [Latombe 1991].

2.2.1 Modelo cinemático del carro común

En el caso de los vehículos de tipo carro común se tienen cuatro llantas, dos fijas en el eje trasero y dos en la parte frontal las cuáles giran para determinar la dirección de movimiento. El rasgo principal del modelo de los automóviles es la presencia de restricciones no-holonómicas, debidas a la condición que existe de rotación puramente rotacional sin que exista deslizamiento lateral entre el piso y las llantas.

Al analizar el modelo de un robot con la misma cinemática de un automóvil, como se muestra en la figura 2.4, por simplicidad, se supone que las dos llantas en cada eje se juntan a una sola llanta en el punto medio del eje (modelo de bicicleta). La llanta delantera se direcciona mientras que la orientación de la llanta trasera es fija. Las coordenadas son $q = (x, y, \theta, \phi)$ donde (x, y) son las coordenadas cartesianas de la llanta trasera, θ mide la orientación del cuerpo del carro con respecto al eje x , y ϕ es el ángulo de la dirección de la llanta delantera con respecto a θ .

El sistema contiene dos restricciones no-holonómicas; una para cada llanta. La ecuación de diferencias que describe la restricción en la llanta delantera se expresa como

$$\dot{x}_f \sin(\theta + \phi) - \dot{y}_f \cos(\theta + \phi) = 0 \quad (2.1)$$

mientras que para la llanta trasera es

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0 \quad (2.2)$$

donde x_f, y_f son las coordenadas de la llanta delantera. Estas ecuaciones establecen, que el movimiento de las llantas hacia los lados es cero.

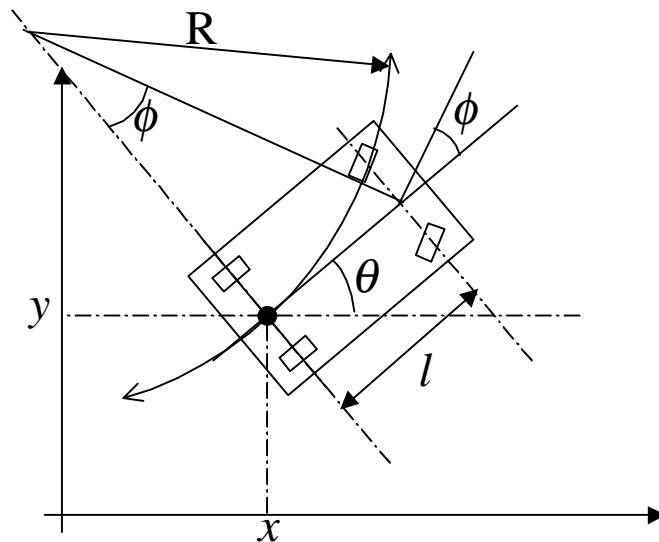


Figura 2.4: Modelo cinemático del carro común. Se localiza en (x,y) del plano con θ como la inclinación del carro y ϕ como el ángulo de direccionamiento

Usando la restricción de cuerpo rígido para calcular la posición de las llantas delanteras, (x_f, y_f) se obtienen de:

$$\begin{aligned} x_f &= x + l \cos \theta \\ y_f &= y + l \sin \theta \end{aligned} \quad (2.3)$$

donde l es la distancia entre los ejes. Reemplazando (2.3) en la primera restricción cinemática (2.1),

$$\dot{x} \sin(\theta + \phi) - \dot{y} \cos(\theta + \phi) - \dot{\theta} l \cos \phi = 0 \quad (2.4)$$

Definamos R como el radio de la curva que describe el vehículo,

$$R = \frac{l}{\tan \phi} \quad (2.5)$$

el modelo cinemático para el vehículo es el siguiente:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos \theta \\ \sin \theta \\ \frac{l}{\tan \phi} \\ 0 \end{bmatrix} v_1 + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} v_2, \quad (2.6)$$

donde v_1 y v_2 son las velocidades del vehículo y de la dirección respectivamente. Cabe mencionar que un modelo cinemático más completo debe incluir también los ángulos de rotación para cada rueda como parte de las coordenadas, para considerar la presencia de actuadores y sensores en el eje de las llantas, así como un modelo sobre las deformaciones en las llantas. No obstante, el modelo propuesto, contiene los elementos esenciales de análisis y es suficiente para propósitos de control [Laumond 1998].

2.2.2 Histéresis de la dirección

Los automóviles poseen una caja diferencial que convierte el movimiento rotacional del volante, a un movimiento lineal en las juntas de las llantas; las cuáles, a su vez, convierten este movimiento lineal en el giro de las llantas de dirección.

Existen muchas conexiones mecánicas en el sistema de dirección. Cada una crea fuente de errores, ya sea por flexión y deflexión en los elementos mecánicos, o por pérdidas de movimiento por conexiones mecánicas flojas. Estos errores impiden que el conductor conozca con exactitud la dirección de las llantas delanteras.

La eficiencia del mecanismo en el vehículo utilizado es muy importante también, ya que el sensor que se utiliza para medir el ángulo de giro, se encuentra en la flecha del motor, así que con histéresis, o “juego” en el mecanismo (ver figura 2.3), se produce una lectura equivocada de la posición real de las llantas delanteras.

Existe una prueba para medir estos errores, conocida con el nombre de “Steer-steer test” y se utiliza en el argot del automovilismo [Milliken, Milliken 1995]; con ella se mide la relación entre el ángulo en el volante y el ángulo real en las llantas delanteras.

Se realizó una prueba para cuantificar este error de histéresis en el vehículo utilizado. La prueba consiste en girar las llantas, en incrementos pequeños de ángulo; después se mide, con un transportador, el ángulo real en cada llanta y se compara con el valor ya digitalizado registrado por el sensor (potenciómetro). El método sirve para demostrar la linealidad o no linealidad de todo el mecanismo, incluyendo el sensor y el convertidor análogo digital. El barrido comienza direccionando las llantas al extremo izquierdo, y cambiando la dirección en incrementos hasta llegar al tope derecho; después, en decrementos, se regresa hasta el tope izquierdo. De esta forma se calcula la diferencia entre las medidas cuando se incrementa, contra lo que se obtiene al decrementar el ángulo. A esta diferencia, en las curvas, se le conoce como histéresis.

La prueba de histéresis se realizó con el vehículo a carga normal y en un lugar estacionario. Se colocó una varilla en las llantas para marcar el ángulo real de giro, en una hoja, posteriormente se midió con un transportador el ángulo de cada línea.

En las figuras 2.5 y 2.6 se observa que la histéresis que presenta el vehículo es grande, debido a la unión entre la flecha del motor, con el mecanismo que mueve las llantas. Esta unión es una abrazadera de plástico flexible que, al girar el motor, hace girar la pieza que sostiene las varillas que mueven las llantas. Si la fricción, entre las llantas y el piso, es mas fuerte que la deformación de esta abrazadera, primero se deforma la abrazadera permitiendo que gire la flecha del motor, aunque, sin causar un movimiento en las llantas.

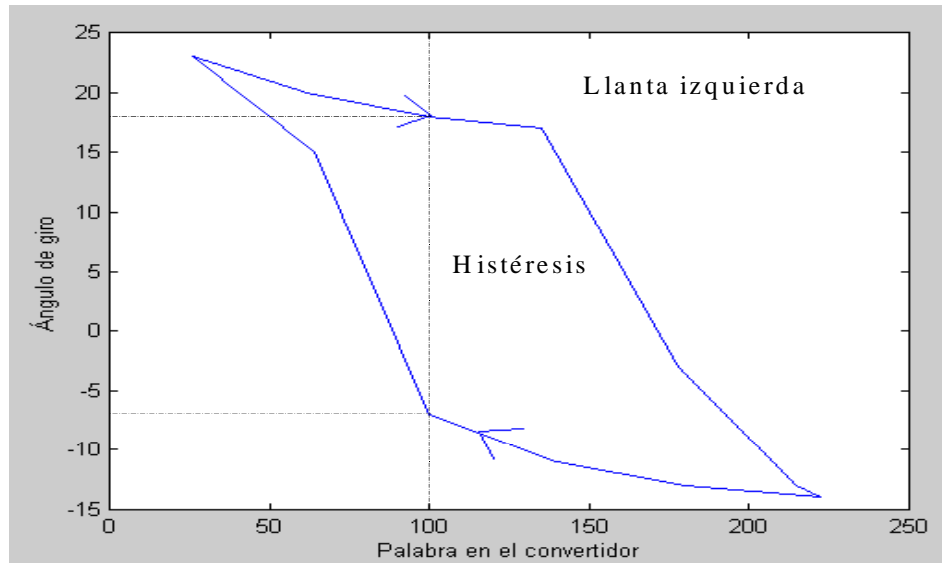


Figura 2.5: Prueba de histéresis en la dirección de la llanta izquierda. El ángulo de giro real, contra la lectura en el convertidor

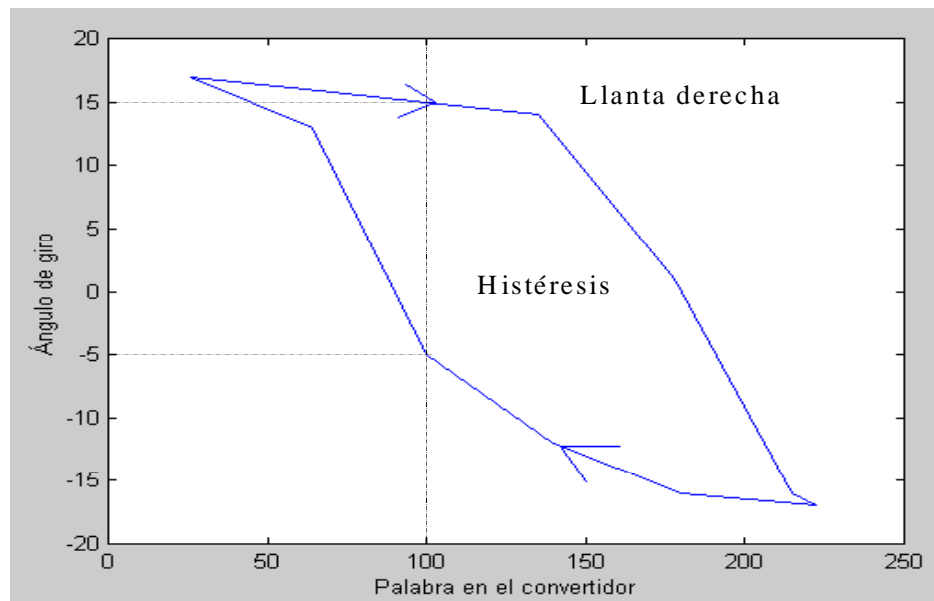


Figura 2.6: Prueba de histéresis en la dirección de la llanta derecha. El ángulo de giro real, contra la lectura en el convertidor

En la figura 2.5 se observa, por ejemplo, que ante una lectura en el sensor de 100 (en un rango de 0 a 255), la llanta izquierda está inclinada un ángulo de 18° al principio cuando el ángulo se incrementa, en cambio cuando la llanta se mueve de regreso decrecentando, se mide un ángulo de -7° ; un comportamiento similar se observa en la llanta derecha (ver figura 2.6).

2.3 Comportamiento Eléctrico

En la sección 2.1 se obtuvo el modelo de un carro común no-holonómico como el que se va utilizar, donde relaciona las variables de entrada: velocidad de desplazamiento y dirección, con las variables de salida: posición e inclinación en el plano. Para controlar el movimiento del vehículo a través de la trayectoria, es necesario modificar las variables de entrada para generar un cambio en la salida. El vehículo que se utilizó, para el desarrollo de esta tesis, cuenta con motores de corriente directa como actuadores para la tracción y dirección. El sistema de velocidad, incluye un motor de corriente directa conectado al tren de engranes que gira las llantas, mientras que el lazo de control de dirección, incluye un motor conectado al mecanismo que direcciona las llantas.

En esta sección se obtendrá el modelo del motor de corriente directa, para después obtener las ecuaciones de función de transferencia para los sistemas de velocidad y dirección.

2.3.1 Motor de corriente directa

El motor de corriente directa, es un elemento electromecánico que convierte la energía eléctrica en movimiento. Su versatilidad hace que sea muy comunmente utilizado como actuador en la robótica. La velocidad de movimiento depende directamente del voltaje que se le aplica.

Para analizar el motor de corriente directa, se representa por su circuito equivalente, como se muestra en la figura 2.7.

El principio de funcionamiento del motor es sencillo, para energizarlo se aplica un voltaje de corriente directa en las terminales. La corriente que fluye por el embobinado genera un campo magnético con cierta polaridad, que en conjunto con el imán permanente, causa un movimiento en la flecha del motor. El sentido de giro depende de la polaridad del campo magnético, generado por la corriente que fluye a través de el embobinado. Entre más corriente, más fuerte es el campo magnético y más rápido se mueve la flecha. A continuación se desarrollarán las ecuaciones del modelo para un motor eléctrico de tal tipo.

El par que desarrolla el motor se expresa por la ecuación:

$$T = K_2 i_a \quad (2.7)$$

donde K_2 es la constante de par del motor e i_a es la corriente en la armadura.

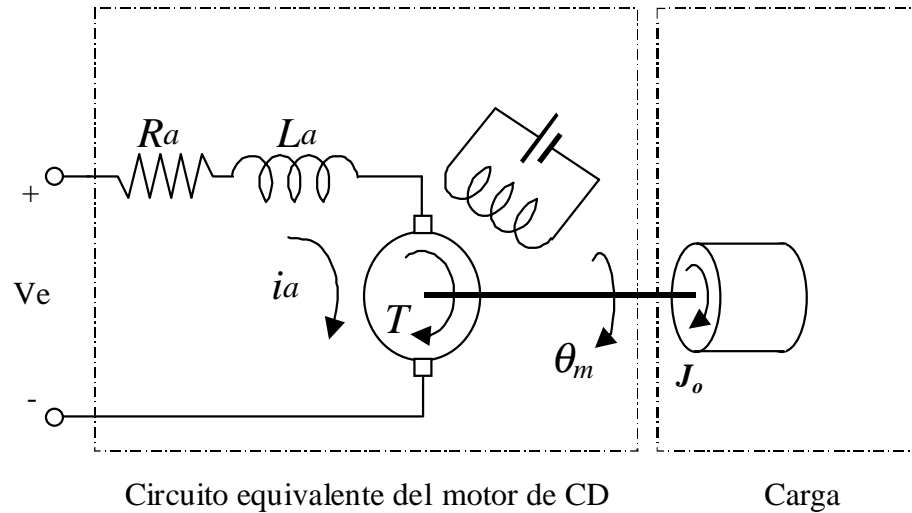


Figura 2.7: Circuito equivalente del motor de CD para obtención del modelo

Cuando la armadura gira, se induce en ella un voltaje proporcional al producto del flujo de corriente y la velocidad angular. Para un flujo de corriente constante, el voltaje inducido e_b es directamente proporcional a la velocidad angular,

$$e_b = K_3 \frac{d\theta_m}{dt} \quad (2.8)$$

donde e_b es la fuerza contraelectromotriz, K_3 es la constante de la fuerza contraelectromotriz del motor y θ_m es el desplazamiento angular en la flecha del motor.

La velocidad del motor controlado por armadura está determinada por el voltaje de la armadura V_e . La ecuación diferencial para el circuito de la armadura es:

$$L_a \frac{di_a}{dt} + R_a i_a + e_b = V_e \quad (2.9)$$

o bien, introduciendo (2.8),

$$L_a \frac{di_a}{dt} + R_a i_a + K_3 \frac{d\theta_m}{dt} = V_e \quad (2.10)$$

La ecuación para el equilibrio de pares es:

$$J_0 \frac{d^2\theta_m}{dt^2} + b_0 \frac{d\theta_m}{dt} = T = K_2 i_a \quad (2.11)$$

donde J_0 es la inercia de la combinación del motor y de la carga referida a la flecha del motor, y b_0 es el coeficiente de fricción viscosa de la combinación del motor y la carga también referida a la flecha del motor. La función de transferencia entre el desplazamiento

angular de la flecha del motor y el voltaje de entrada se obtiene de las ecuaciones (2.10) y (2.11); expresada en el dominio de Laplace [Ogata 1998],

$$\frac{\Theta_m(s)}{V_e(s)} = \frac{K_2}{s(L_a s + R_a)(J_0 s + b_0) + K_2 K_3 s} \quad (2.12)$$

dado que L_a es por lo general pequeña, se puede eliminar de la función de transferencia. Después de eliminar el término L_a y dividiendo la expresión entre R_a ,

$$\frac{\Theta_m(s)}{V_e(s)} = \frac{K_2 / R_a}{J_0 s^2 + \left(b_0 + \frac{K_2 K_3}{R_a} \right) s} \quad (2.13)$$

2.3.2 Modelo del sistema de dirección

El motor de corriente directa, en el mecanismo de dirección, mueve las llantas delanteras al girar. Por ello, la inclinación de las llantas depende directamente de la posición de la flecha del motor. El sistema se compone de una entrada y una salida: la entrada es el voltaje aplicado al motor y la salida es la inclinación de las llantas. Es fácil observar que el sistema cuenta con un integrador, ya que al aplicar una entrada escalón (un voltaje fijo), la inclinación de las llantas crece hasta llegar al tope impuesto en el mecanismo de dirección.

Debido a que este sistema tiene un integrador, no se pueden realizar pruebas escalón en lazo abierto, para su identificación. Debido a lo anterior, el método para calcular el modelo del motor-posición es teórico matemático. En esta sección se obtendrá la función de transferencia para el modelo del sistema de dirección.

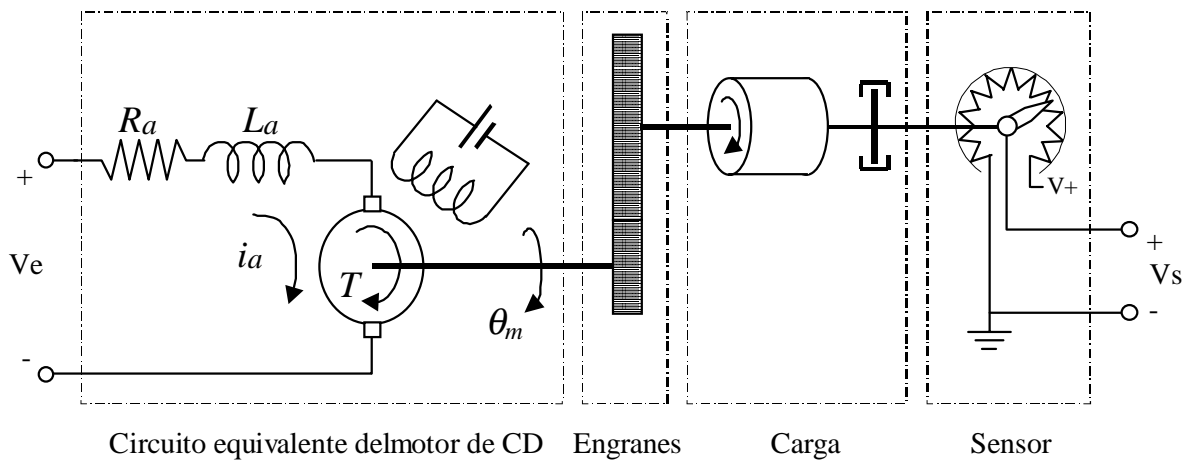


Figura 2.8: Circuito equivalente del motor de CD con la carga y sensor de posición, para el sistema de dirección.

El sistema de dirección incluye el motor de corriente directa analizado en la sección

anterior, pero se le agregó un potenciómetro para medir la posición de la flecha del motor, como se muestra en la figura 2.8 (este sensor se explica en el Capítulo 3). Siendo el voltaje de salida V_s , proporcional a la posición de la flecha del motor, se expresa en el dominio de Laplace por la relación:

$$V_s(s) = K_4 \Theta_m(s) \quad (2.14)$$

donde K_4 es una constante de proporcionalidad. Retomando la ecuación (2.13) y sustituyendo (2.14), se obtiene la función de transferencia entre voltaje de entrada y voltaje de salida,

$$G(s) = \frac{V_s(s)}{V_e(s)} = \frac{K}{s(T_m s + 1)} \quad (2.15)$$

donde,

$$K = \frac{K_2 K_4}{R_a} \quad (2.16)$$

$$T_m = \frac{R_a J_0}{b_0 R_a + K_2 K_3} \quad (2.17)$$

De la ecuación (2.15) podemos ver que la función de transferencia contiene un término $1/s$, por lo tanto este modelo posee una propiedad de integración. T_m es la constante de tiempo del sistema. En (2.17) se observa que, cuando la inercia de la carga J_0 , tiende a cero, el sistema se comporta como un integrador ideal [Ogata 1998].

2.3.3 Modelo del sistema de velocidad

En el sistema de velocidad se tiene también motores de corriente directa, sólo que en este caso se trata de dos motores: uno para las llantas del eje delantero y otro para las llantas del eje trasero. Los dos motores se conectan en paralelo, de forma que la configuración del circuito equivalente no se modifica; lo único que cambia es el valor de la resistencia y la inductancia, por un valor más pequeño. Esta configuración provoca que el consumo de corriente sea mayor. En esta sección se obtendrá la función de transferencia del sistema de velocidad.

La ecuación (2.13) expresa la función de transferencia entre el voltaje de entrada contra la posición de un motor de corriente directa. Por su parte, la relación de la velocidad angular con respecto a la posición de la flecha del motor se expresa por la relación:

$$V(s) = s \Theta_m(s) \quad (2.18)$$

donde la s de Laplace denota una derivada. Siendo así, la velocidad angular es la razón de cambio de la posición angular con respecto al tiempo.

Sustituyendo (2.18) en la ecuación (2.13),

$$\frac{V(s)}{V_e(s)} = \frac{K_2 / R_a}{J_0 s + \left(b_0 + \frac{K_2 K_3}{R_a} \right)} \quad (3.27)$$

Por lo tanto la función de transferencia entre voltaje de entrada y velocidad se simplifica,

$$G_v(s) = \frac{V(s)}{V_e(s)} = \frac{K}{T_m s + 1} \quad (3.28)$$

donde,

$$K = \frac{K_2}{R_a} \quad (3.29)$$

$$T_m = \frac{R_a J_0}{b_0 R_a + K_2 K_3} \quad (3.30)$$

Este es un sistema de primer orden con una constante de tiempo igual a T_m . Se observa que cuando el momento de inercia de la carga J_0 tiende a cero, la constante de tiempo del sistema también tiende a cero; en estas condiciones el sistema respondería inmediatamente con una ganancia igual a la entrada. Es importante observar como el cambio en el peso del vehículo cambia la constante de tiempo; puesto que esta constante se relaciona directamente con el momento de inercia.

2.4 Lista de términos

- x : posición del vehículo en el eje x
- y : posición del vehículo en el eje y
- θ : orientación del vehículo en el plano
- ϕ : dirección de las llantas delanteras del vehículo
- l : distancia entre el eje delantero y trasero del vehículo
- R : radio del círculo que describe el vehículo, al desplazarse en el plano
- R_a : resistencia de armadura, del motor de CD
- L_a : inductancia de armadura, del motor de CD

-
- V_e : voltaje en la armadura, del motor de CD
 - $V_s(s)$: voltaje de salida en el potenciómetro que mide dirección, dominio de Laplace
 - I_a : corriente en la armadura, del motor de CD
 - T : par desarrollado por el motor
 - J_o : momento de inercia debido a la carga
 - b_o : coeficiente de fricción viscosa referida a la flecha del motor
 - θ_m : desplazamiento angular en la flecha del motor
 - $\Theta_m(s)$: desplazamiento angular en la flecha del motor, en el dominio de Laplace
 - e_b : fuerza contraelectromotriz
 - K_2 : constante de par del motor
 - K_3 : constante de la fuerza contraelectromotriz
 - K_4 : constante de proporcionalidad en el voltaje que genera el potenciómetro
 - $V(s)$: Velocidad angular del motor, en el dominio de Laplace
 - s : variable en el dominio de Laplace

Capítulo 3

Arquitectura para el control del vehículo

Desde el punto de vista de un piloto humano, el control de un vehículo requiere del volante de dirección y del acelerador. Si el objetivo es sustituir al humano por un controlador; entonces, la arquitectura de control debe establecer una configuración que permita al controlador manipular la dirección y la velocidad, además de cerrar el lazo de control, retroalimentando la posición del vehículo.

En este capítulo se propone una arquitectura, la cuál establece la configuración de todos los elementos involucrados en el control del vehículo. El vehículo que se quiere controlar, cuenta con dos motores de corriente directa en paralelo para avance y retroceso, además, otro motor de corriente directa para manipular la dirección; también cuenta con dos sensores, uno para medir la orientación de las llantas delanteras, y otro para medir la posición y velocidad.

Todos los módulos eléctricos y electrónicos que componen la arquitectura se describen en este capítulo; entre ellos, los más importantes son, la tarjeta de control del motor de dirección, y la tarjeta para el control de velocidad; estas tarjetas incluyen una etapa de potencia para manipular los motores y un circuito de acondicionamiento para obtener las señales de los sensores. Además, la arquitectura se compone de la unidad de procesamiento y el módulo de comunicación inalámbrica. Los algoritmos para el control de velocidad y dirección se explican a detalle en el siguiente capítulo.

3.1 Estructura y módulos de la arquitectura

Como se mencionó al principio de este capítulo, la arquitectura se compone de módulos, según la tarea que ejecuta cada uno. En la figura 3.1 se muestra con diagramas de bloques la distribución de los módulos. La arquitectura de control propuesta separa el control del vehículo en diferentes niveles jerárquicos en cascada. En el diagrama a bloques se muestra esta jerarquía de mayor a menor, comenzando por el usuario, hasta llegar a los motores y

sensores del vehículo.

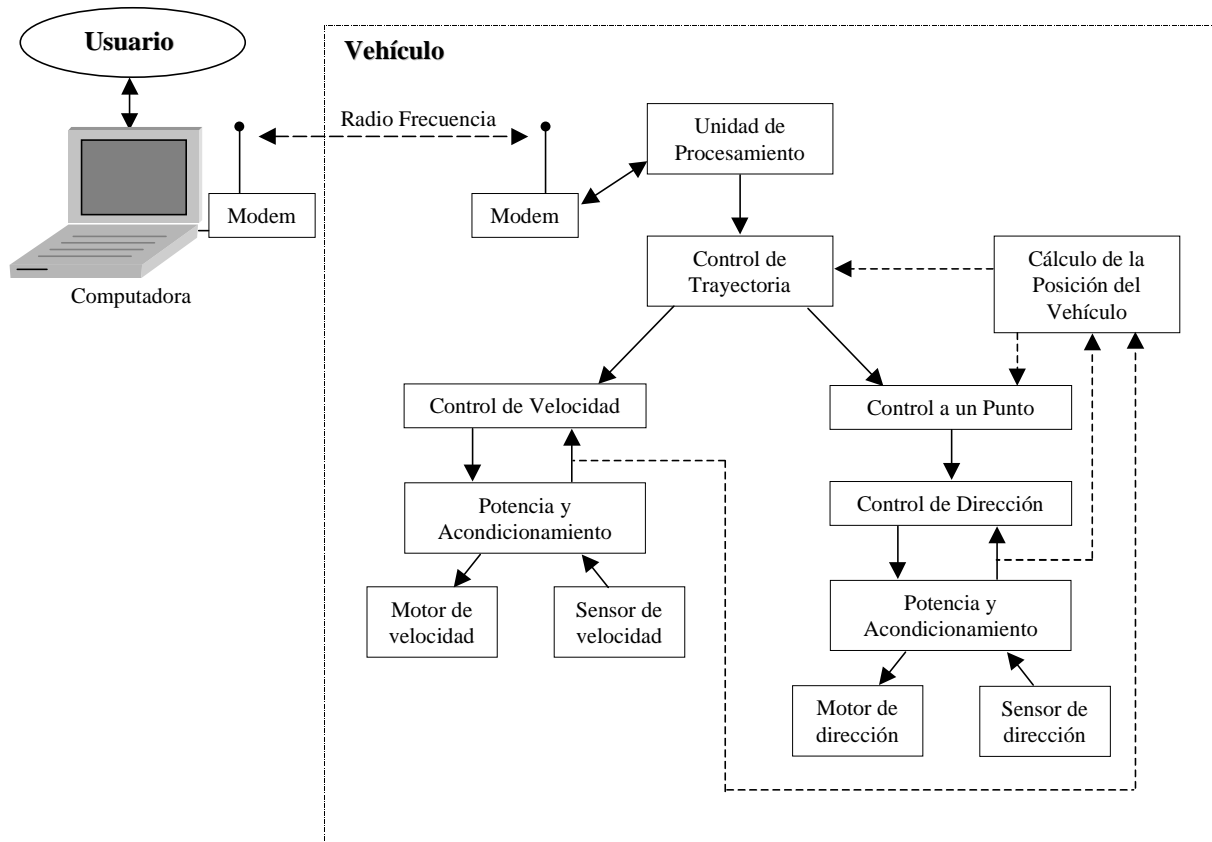


Figura 3.1: Diagrama a bloques de los módulos que componen la arquitectura de control en términos de electrónica y flujo de señales del vehículo

A continuación se hará una breve descripción de cada uno de los bloques; por ahora, sólo se describirán los bloques que se encuentran dentro del vehículo.

La unidad de procesamiento procesa los comandos que recibe del usuario, ya sea para definir la trayectoria, sintonizar los controladores o iniciar la ejecución de la trayectoria.

El controlador de trayectoria verifica la posición actual para determinar cuál es el siguiente punto a seguir de la trayectoria y lo envía como referencia al controlador a un punto. Cuando se llega al último punto, este controlador envía un comando para detener el vehículo al control de velocidad.

El controlador de velocidad se encarga de alcanzar y mantener el vehículo a la velocidad deseada y envía señales a la etapa de potencia en la tarjeta de velocidad para que, a su vez, mueva los motores de avance. Este controlador, cierra el lazo con la señal que envía el sensor de velocidad a través del circuito de acondicionamiento en la tarjeta.

El control a un punto dirige el movimiento del vehículo hacia el punto que recibe como referencia. El procedimiento consiste en calcular la amplitud de giro de las llantas

delanteras, para minimizar el ángulo de error entre el vehículo y el punto. Finalmente envía al control de dirección el ángulo de giro requerido, como referencia.

El controlador de dirección inclina las llantas de dirección al ángulo requerido; además envía señales a la etapa de potencia en la tarjeta de dirección para que, a su vez, mueva el motor de dirección. El controlador de dirección, cierra su lazo con la señal que envía el sensor de dirección a través de la tarjeta de acondicionamiento.

El cálculo de la posición del vehículo se realiza con base en las señales de los sensores de dirección y velocidad, utilizando un algoritmo que realiza una estimación de los parámetros del vehículo, basado en las ecuaciones cinemáticas. Este es uno de los módulos más importantes, ya que la posición que se calcula en este módulo es la que se utiliza para cerrar el lazo de control de trayectoria y el control a un punto. El cálculo se realiza en la unidad de procesamiento; en el Capítulo 4 se hace una descripción detallada de este módulo.

La configuración de la arquitectura propuesta, es modular y permite hacer cambios en el algoritmo de los controladores, sin alterar los demás controladores. Para que el usuario pueda modificar u observar los parámetros del vehículo, tiene que haber un módulo que se encargue de entablar una comunicación con el vehículo desde un lugar estacionario, mientras que el vehículo efectúa las tareas que se le ordenaron. Dicha comunicación debe de ser inalámbrica para que el vehículo pueda moverse libremente en su área de trabajo.

3.2 Instrumentación del vehículo

En el capítulo anterior se obtuvo el modelo del sistema de dirección y velocidad; sin embargo las ecuaciones de transferencia están en términos de un voltaje en la entrada, contra una variable física en la salida. Los controladores que se proponen en la arquitectura de control se basan en el uso de microcontroladores para su implementación; por lo tanto, las variables son únicamente datos digitales almacenados en la memoria del microcontrolador. Se requiere de algún instrumento que convierta los datos digitales calculados, en voltajes para que funcionen los motores y convierta las variables físicas de salida, en datos digitales para que los reconozca el controlador.

En esta sección se describen a detalle las tarjetas para el control de dirección y velocidad. Como se dijo anteriormente, cada tarjeta se puede dividir en dos partes principales, la etapa de potencia para actuar los motores y la de acondicionamiento de la señal del sensor. En cuanto a la etapa de potencia, tanto para la dirección como para la velocidad, se utiliza un circuito basado en transistores, llamado puente H; el microcontrolador se conecta a este circuito y con base en una señal modulada en ancho de pulso, envía la potencia al motor. En cuanto al circuito de acondicionamiento, la tarjeta para el control de dirección y velocidad son diferentes, debido a que el sensor para medir la dirección, es diferente al que mide la velocidad. En la dirección se utiliza un potenciómetro, mientras que para velocidad, se utiliza un codificador óptico. A continuación se describe cada uno de los elementos en las tarjetas de dirección y velocidad.

3.2.1 Etapa de potencia en tarjetas de control

El propósito de la etapa de potencia, es entregar suficiente energía a los motores de corriente directa con que cuenta el vehículo. Como se dijo anteriormente, la tarjeta para control de dirección y la de velocidad utilizan el mismo principio de funcionamiento, motivo por el cuál, se hace la explicación generalizando para las dos tarjetas.

Puente H, sentido de giro del motor

Para el control de los motores se utilizó la configuración *puente H*, cuyo objetivo es girar un motor de corriente directa en los dos sentidos, la inversión del giro se logra al cambiar la polarización de las bobinas del motor, para invertir el flujo de corriente a través del motor.

El circuito básico se muestra en la figura 3.2. El principio de operación indica que sólo es posible cerrar los interruptores $S1-S4$ ó $S3-S2$ a la vez. Al cerrar $S1$ y $S4$ el motor se polariza con $V+$ en el lado izquierdo y tierra del lado derecho, provocando que la corriente fluya de izquierda a derecha. Si se activan $S3-S2$, el $V+$ aparece del lado derecho y tierra del lado izquierdo del motor, por lo tanto, la corriente fluye de derecha a izquierda provocando un giro del motor en sentido contrario.

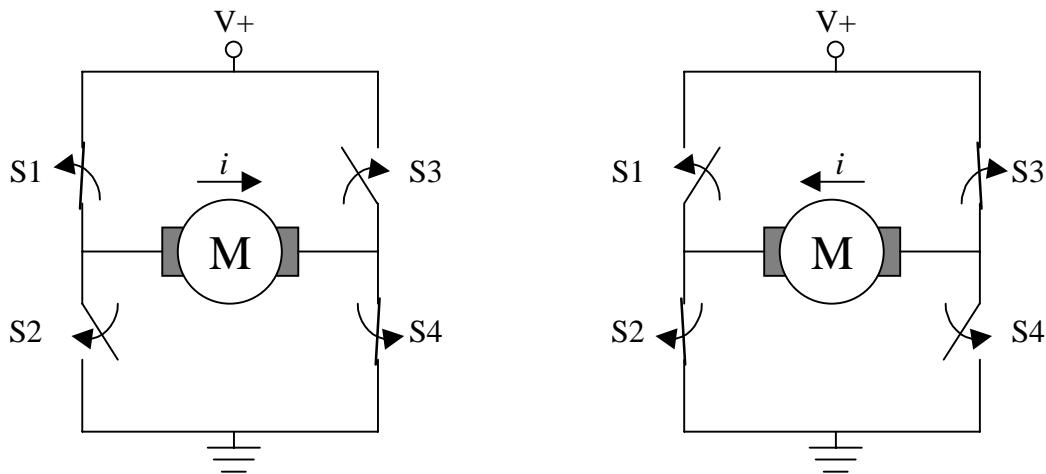


Figura 3.2: Los dos estados posibles en el *puente H*. Cuando los contactos $S1$ y $S4$ se encuentran cerrados el motor gira en un sentido; mientras que con $S2$ y $S3$ cerrados, el giro es en sentido contrario

En la práctica los interruptores se sustituyen por transistores o relevadores que realizan la conmutación. En el Anexo X se incluye el diagrama eléctrico de las tarjetas para el control de los motores de dirección y de velocidad en el vehículo de pruebas.

PWM, aplicación de potencia

El *puente H* permite cambiar el sentido de giro del motor. Ahora bien, para completar el objetivo del control, también es necesario variar la velocidad del motor; lo cuál se logra

variando la potencia que recibe el motor. Una forma de variar la velocidad, es cambiando el voltaje de alimentación, pero es necesario que el microcontrolador realice esta tarea, además un convertidor digital análogo no podría entregar la potencia necesaria. Por ello se utiliza una técnica llamada “*Modulación de Ancho de Pulso*” o “*PWM*” por sus siglas en inglés.

Las señales *PWM* son señales de voltaje con un período constante, donde se cambia el ciclo de trabajo. Así, la potencia que recibe el motor es directamente proporcional al ciclo de trabajo de la señal. De este modo se realiza un barrido de 0% a 100% de potencia entregada al motor.

El ciclo de trabajo se calcula con la siguiente ecuación:

$$\%DC = 100 \left(\frac{t_H}{T} \right) \quad (3.1)$$

donde $\%DC$ es el ciclo de trabajo, t_H es el tiempo en nivel alto, T es el período de la señal.

En la figura 3.3 se muestran ejemplos de señales *PWM* con diferentes ciclos de trabajo. Si esta señal se utiliza para conmutar los interruptores del *punte H*, se variará la velocidad del motor además del sentido de giro.

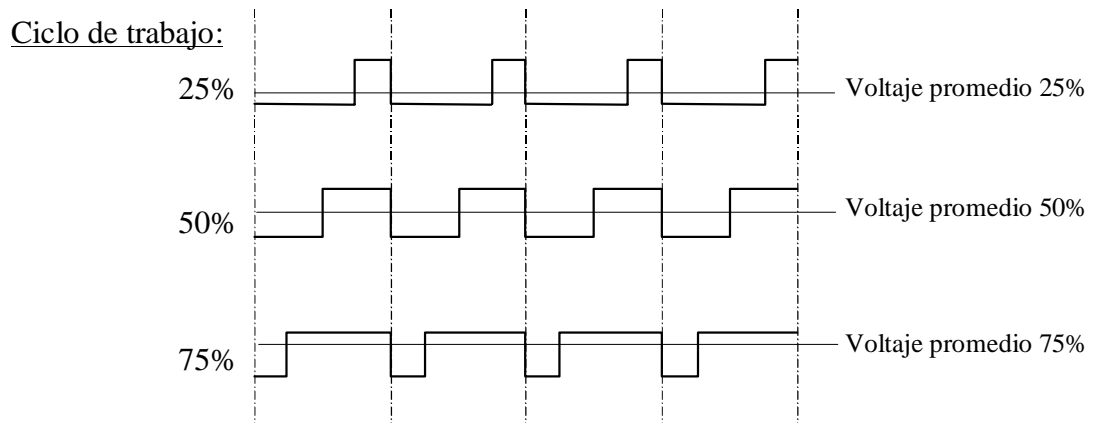


Figura 3.3. Ejemplo de señales *PWM*. La señal permanece en estado alto un 25%, 50% y 75% del tiempo, entregando ese mismo porcentaje de voltaje en promedio

En la práctica, las señales de *PWM* se utilizan con un período de tiempo pequeño, comparado con la constante de tiempo del actuador, de tal forma que no alcance a responder a los pulsos, pero sí a un voltaje promedio.

3.2.2 Sensado y acondicionamiento de dirección

Para que el controlador de dirección pueda cerrar el lazo, requiere medir la inclinación actual de las llantas. El algoritmo del controlador de dirección reside en el microcontrolador, por lo tanto, la inclinación medida debe ser un valor digital, para que pueda leerla en cada

muestreo; el acondicionamiento en la señal, tiene ese propósito.

Existen muchas técnicas para medir desplazamiento angular; entre las más utilizadas en el área de robótica están los potenciómetros y los codificadores ópticos.

Los codificadores ópticos absolutos entregan una salida digital de “n” bits, según la resolución. Se componen principalmente de un disco opaco con perforaciones que permiten o impiden el paso de la luz. Dichas perforaciones se colocan de tal forma que se pueda obtener un código conocido, ya sea binario, BCD, Gray, etc., conforme se gira el disco [Dieck 2000]. La figura 3.4 muestra un ejemplo de un codificador óptico absoluto de tres bits.

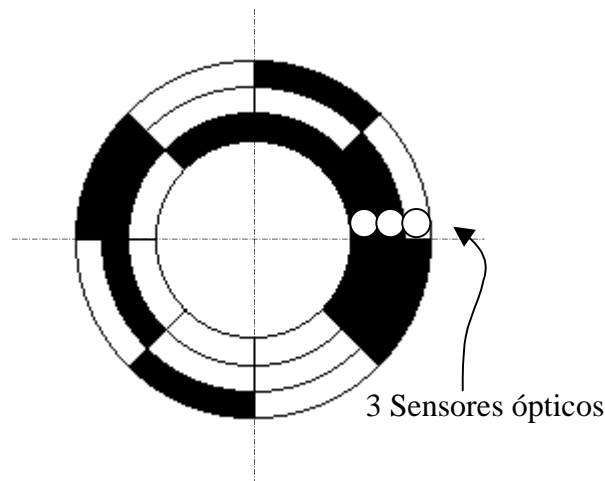


Figura 3.4: Ejemplo de codificador óptico absoluto de tres bits en código binario. El ángulo actual se obtiene con una resolución de 45 grados

El problema, con este tipo de sensores, es que si se requiere una resolución mayor, el tamaño del disco crece y su costo aumenta. Por ejemplo, para una resolución de 8 bits, se requerirían ocho sensores ópticos y el disco estaría dividido en 256 pasos, para lo cuál, se requiere de mucha precisión y su construcción es difícil de obtener con métodos sencillos.

Por otro lado, los potenciómetros son más sencillos de conseguir y se puede obtener una mayor resolución a un costo menor. El potenciómetro se compone de una resistencia con un devanado central que recorre todo el rango. La salida es una señal analógica continua y no está discretizada, por lo tanto, la resolución depende directamente del método que se utilice para digitalizar esta señal. Aquí se requiere que la tarjeta de adquisición de datos contenga un convertidor analógico-digital para discretizar la señal. En el vehículo que se utilizó, el motor de dirección ya contaba con un potenciómetro dentro del mismo encapsulado. Este potenciómetro se utilizó para medir el ángulo de la dirección. En el Anexo X se incluye el diagrama del circuito de adquisición con un convertidor analógico-digital, utilizado para convertir la señal del potenciómetro.

En la figura 3.5 se muestra el diagrama de la configuración del potenciómetro para medir el ángulo. Como se dijo anteriormente, este potenciómetro venía incluido en el mismo

encapsulado del motor que manipula la dirección, por lo que, el sensor mide la posición de la flecha del motor y no la inclinación real de las llantas delanteras. En el Capítulo 2, se hizo un análisis de la histéresis que produce el mecanismo de dirección.

Para calcular el valor del voltaje de salida se utiliza la siguiente ecuación:

$$V_{salida} = \frac{V}{R_1 + R_2} R_1 \quad (3.2)$$

En la figura se observa, que para la posición central de las llantas, esto es, cuando el vehículo se dirige en línea recta, el devanado se encuentra en el centro, por lo tanto, R_1 es igual a R_2 , provocando que el voltaje de salida sea igual a la mitad del voltaje aplicado. Si la señal de voltaje de salida se digitaliza usando un convertidor análogo-digital de 8 bits, el valor digital en la posición central, sería un 80 hexadecimal, cuando el valor equivalente del ángulo en las llantas delanteras: ϕ , sería igual a cero.

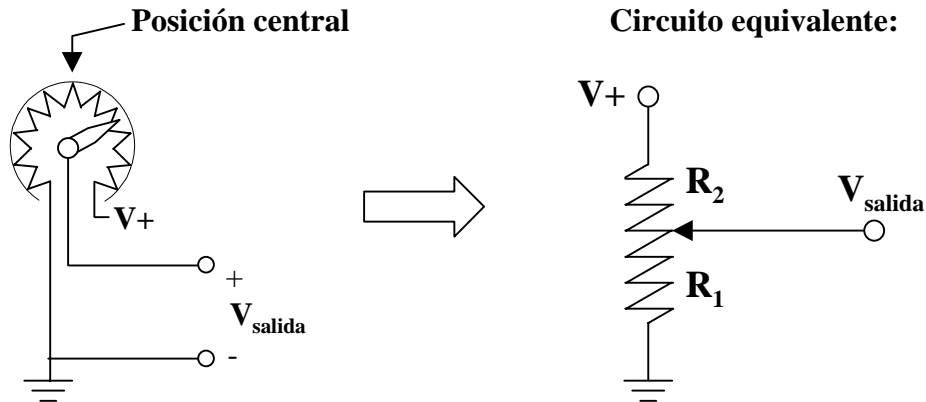


Figura 3.5: Configuración para medir ángulo con un potenciómetro. El voltaje en la salida depende directamente del ángulo que forma el devanado central.

3.2.3 Sensado y acondicionamiento de velocidad

Referente al sensado de avance y velocidad, también existen diferentes esquemas que podrían implementarse; sin embargo, los más utilizados en robótica son los codificadores incrementales, puesto que producen directamente una salida digital cuando miden desplazamientos. En este caso, no se utilizan convertidores análogo-digital, ya que la salida es inherentemente digital y puede ser acumulada por un contador o en la misma unidad de procesamiento principal. La transducción de movimiento se realiza usando tres mecanismos de codificación posible: codificadores magnéticos, codificadores ópticos, codificadores por escobillas [Dieck 2000]; sin embargo, el principio de funcionamiento es el mismo.

La salida de los codificadores incrementales es un tren de pulsos cuadrados que se generan conforme hay movimiento. El período de la señal depende directamente de la velocidad del movimiento. La diferencia de este tipo de sensores con los codificadores

absolutos es que no se tiene un código de la posición, sólo indica si hay o no movimiento y por medio de su frecuencia se determina la velocidad del movimiento.

Al contar con una segunda señal desfasada 90°, se conoce también la dirección del movimiento, al medir las transiciones en la salida de los sensores y verificando cuál señal adelanta a la otra. Un ejemplo de codificador óptico incremental con dos señales en cuadratura se presenta en la figura 3.6.

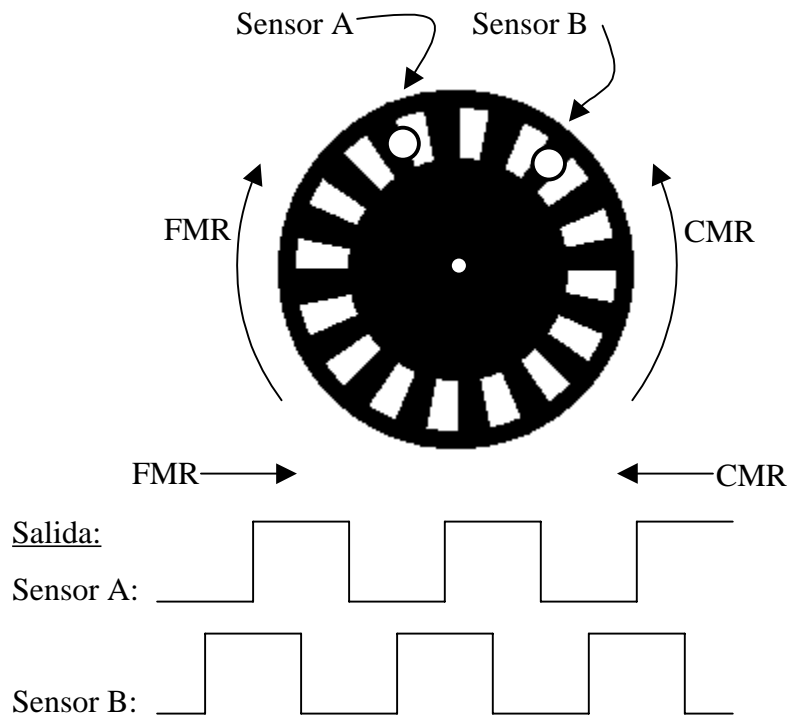


Figura 3.6: Codificador óptico incremental, con dos señales desfasadas, para obtener la dirección del giro. Durante el movimiento a favor de las manecillas del reloj, la señal del sensor B adelanta a la de A, en contra de la manecillas del reloj, la señal del sensor A adelanta a la de B

Si se mide la cantidad de pulsos generados por unidad de tiempo se obtiene la velocidad. En el vehículo se utilizó un codificador óptico incremental, con dos canales en cuadratura, para medir la distancia recorrida así como la velocidad.

El cálculo de la distancia recorrida se realiza usando el número de pulsos que se generan por vuelta de llanta, si además se conoce el perímetro de la llanta, se puede calcular la distancia recorrida por cada pulso.

$$D = \frac{P_{llanta}}{Np} \quad (3.3)$$

donde D es la distancia recorrida en cada pulso, P_{llanta} es el perímetro de la llanta y Np es el

número de pulsos por vuelta de llanta.

En el Capítulo 4, se explica cómo con el valor de distancia y posición en la dirección, se puede hacer una estimación del cambio en la posición y orientación del vehículo en el plano.

El cálculo de la velocidad, se hace usando el mismo sensor, pero con un algoritmo diferente. Se cuenta el número de pulsos detectados en el codificador, dentro de una ventana de tiempo. Si se conoce la distancia que se recorre por cada pulso y se conoce el tamaño de la ventana de tiempo, entonces se puede calcular la velocidad. La ecuación para el cálculo de la velocidad queda de la siguiente manera:

$$V = \frac{P_c}{T_v} D \quad (3.4)$$

donde V es la velocidad del vehículo, P_c es el número de pulsos contados, T_v es el tiempo que transcurre y D es la distancia recorrida por cada pulso.

Para determinar si el vehículo está avanzando o retrocediendo, se utiliza la segunda señal del codificador. Al momento en que se detecta un pulso positivo en la señal A, se mide el valor de la señal B, si se encuentra en estado alto, significa que el vehículo avanza, y si está en estado bajo, significa que el vehículo retrocede.

3.3 Unidad de procesamiento

El vehículo requiere de un dispositivo con capacidad de procesamiento para ejecutar los algoritmos de control, procesar las señales de los sensores, recibir los comandos del usuario, etc. Regularmente se utiliza una computadora para satisfacer las necesidades de procesamiento, las cuáles incluyen capacidad de memoria, equipos periféricos de entrada salida y puerto serial principalmente.

La arquitectura propuesta se basa en el uso de microcontroladores, ya que integran todas estas funciones en una sola pastilla. Se utilizan principalmente en aplicaciones específicas y dedicadas. Los más comunes cuentan con memoria integrada, puertos para entrada y salida, puerto serial, atención a interrupciones externas y contadores. Los más nuevos cuentan ya con convertidores análogo digital, salidas con PWM, comparadores análogos, etc, todo en un mismo circuito integrado.

Para este trabajo se utilizó un microcontrolador de la familia AVR de Atmel porque incluye dos salidas con PWM, además por su alta velocidad de procesamiento y su bajo costo.

Capítulo 4

Sistema jerárquico de control

En este capítulo se describe el sistema de control del vehículo. Este sistema se encarga de manipular la velocidad de desplazamiento y el volante de dirección para controlar la posición e inclinación del vehículo en el plano y de esta forma, seguir una cierta trayectoria.

Las trayectorias se definen con base en puntos en el plano, por donde debe pasar el vehículo. Existe un controlador que se encarga de llevar el vehículo a un punto, si a este controlador se le van proporcionando los puntos que componen la trayectoria, el objetivo de seguir una trayectoria se satisface. El controlador a un punto minimiza el error entre la orientación actual del vehículo y el ángulo formado por la línea recta que une la posición actual del vehículo con el punto; por lo tanto, si el error es cero, el vehículo se dirige directamente hacia el punto, suponiendo que el vehículo avanza a cierta velocidad.

El controlador a un punto utiliza la técnica de control jerárquico en cascada. El controlador de dirección se encarga de orientar las llantas a un cierto ángulo, mientras que la variable manipuladora del controlador a un punto, es el valor deseado de este ángulo.

Debido a que la trayectoria se compone de varios puntos, se desarrolló un algoritmo que determina cuando se llegó a un punto y se debe pasar al siguiente, a este algoritmo se le llamó controlador de trayectoria.

Un último controlador se ocupa de la velocidad del vehículo. La velocidad la especifica el usuario y se mantiene fija durante toda la trayectoria. Al final de la trayectoria, el controlador de trayectoria envía un comando para frenar el vehículo.

Como no se cuenta con un sensor externo que obtenga la posición y orientación actual del vehículo, se calcula por medio de un algoritmo de estimación interna de parámetros, que se basa en la inclinación actual de las llantas y la distancia recorrida en cada muestreo (dead reckoning).

El diagrama a bloques completo del sistema de control se muestra en la figura 4.1. La entrada al sistema es un arreglo de coordenadas, que definen los puntos de la trayectoria; la salida es la posición e inclinación del vehículo. En el diagrama se observa como la

retroalimentación, al controlador, es la posición e inclinación estimada y no la posición real; por lo tanto, el desempeño del controlador depende directamente de qué tan precisa sea la estimación.

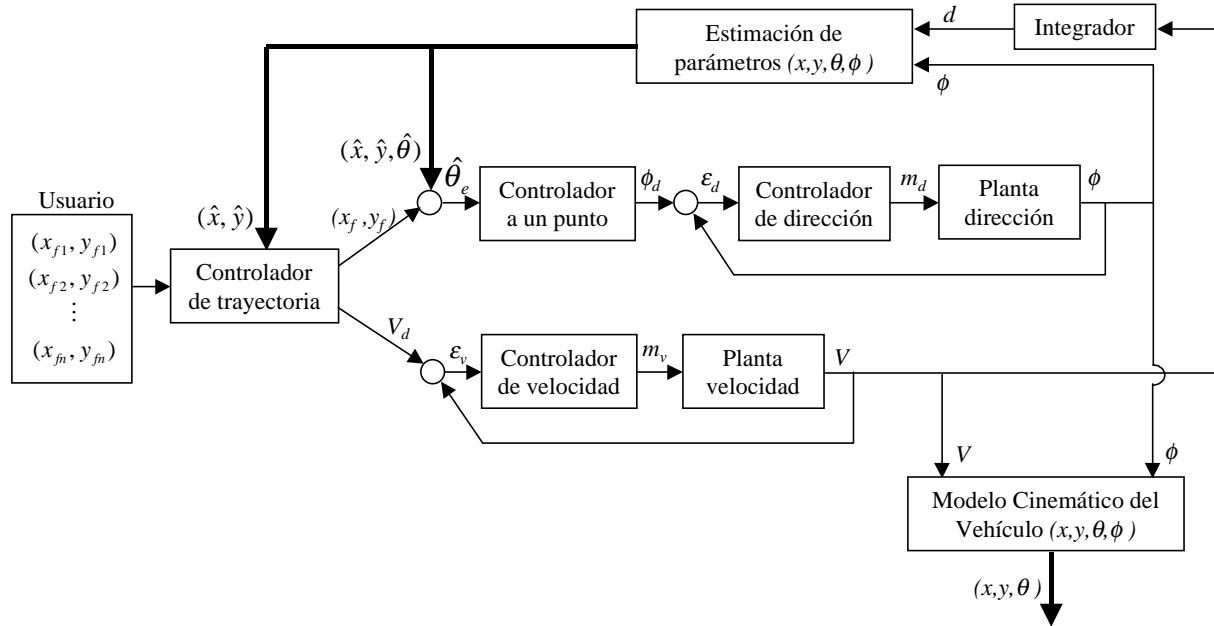


Figura 4.1: Diagrama a bloques del sistema de control propuesto

En la figura 4.1 se encuentran las siguientes variables:

- (x, y, θ) es la posición y orientación actual del vehículo teórica.
- $(\hat{x}, \hat{y}, \hat{\theta})$ es la posición y orientación actual del vehículo estimada.
- (x_f, y_f) son las coordenadas del punto a seguir.
- $\hat{\theta}_e$ es el error entre la inclinación actual del vehículo y el ángulo formado por la línea recta que une la posición actual del vehículo con el punto: $\hat{\theta}_e = \hat{\theta} - \tan^{-1}\left(\frac{y_f - \hat{y}}{x_f - \hat{x}}\right)$.
- ϕ_d es la inclinación de las llantas delanteras deseada (o salida del controlador a un punto).
- ϕ es la inclinación medida de las llantas delanteras.
- $\varepsilon_d = \phi_d - \phi$ es el error en la dirección.

- m_d es la variable manipuladora del control de dirección.
- V_d es la velocidad deseada especificada por el usuario.
- V es la velocidad medida actual.
- $\varepsilon_v = V_d - V$ es el error en la velocidad.
- m_v es la variable manipuladora del control de velocidad.

En este capítulo se describe a detalle cada uno de los bloques de control y la estimación de los parámetros.

4.1 Control de dirección

El controlador de dirección calcula las manipulaciones para el motor de la dirección, con el fin de colocar las llantas con la orientación deseada. El sistema tiene un integrador por su construcción, motivo por el cuál, al aplicar una entrada escalón, la posición aumenta y el motor no se detiene hasta no eliminar el voltaje en la entrada o hasta llegar al tope mecánico. Este comportamiento ayuda a seleccionar el tipo de controlador, puesto que debe mandar una manipulación de cero, al tener cero error.

El controlador se encarga de llevar la variable controlada en un valor deseado de referencia y mantenerla. En este caso, la variable controlada es el ángulo ϕ , de la orientación de las llantas delanteras. El control de dirección es un esclavo, por ello, la referencia la indica el controlador a un punto. El control es digital, pues como se dijo en el capítulo anterior, se implementa en un microcontrolador. El lazo de control digital completo se muestra en la figura 4.2

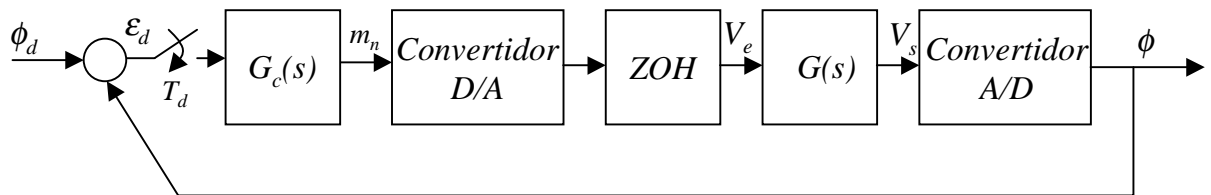


Figura 4.2: Lazo de control digital para la dirección. $G_c(s)$ es la función de transferencia del controlador y $G(s)$ es el modelo del sistema

Se propone utilizar un controlador PD digital en modo posición, cuyo cálculo de la variable manipulada m_n ,

$$m_n = K_c \left[\varepsilon_{dn} + \frac{\tau_d}{T_d} (\varepsilon_{dn} - \varepsilon_{dn-1}) \right] + m_{ss} \quad (4.1)$$

$$\varepsilon_d = \phi_d - \phi \quad (4.2)$$

donde K_c es la ganancia proporcional, τ_d es la ganancia derivativa, T_d es el tiempo de muestreo, ε_{dn} es el error actual, ε_{dn-1} es el error un tiempo de muestreo anterior y m_{ss} es la manipulación mínima que no provoca movimiento en el sistema o manipulación de estado estable.

En la ecuación de control se observa que, al tener un error igual a cero, la manipulación va a ser igual a m_{ss} , lo cuál significa que el motor se detiene. El efecto derivativo provoca un pico en la manipulación al momento de efectuar cualquier cambio en la referencia. Este pico es útil para romper la inercia inicial del motor.

4.1.1 Desempeño del controlador

Se realizaron pruebas con el fin de verificar el desempeño del controlador en lazo cerrado. Las pruebas consistieron en aplicar entradas escalón en forma ascendente y descendente haciendo un barrido completo del rango de operación.

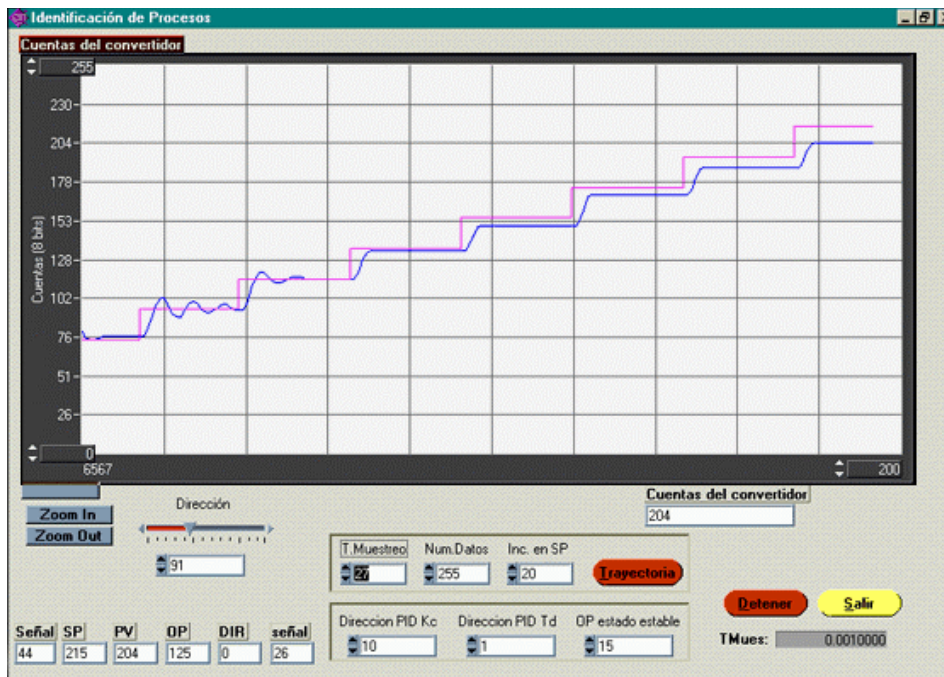


Figura 4.3: Pruebas escalón en forma ascendente al controlador PD de la dirección. En la zona de operación de abajo se observa una ganancia grande, mientras que arriba se observa muy poca ganancia

Se observa que la planta no es lineal, puesto que en una zona de operación presenta un sobreimpulso mientras que en otra no alcanza la referencia. La variación no sólo depende de la zona de operación, sino también del sentido de giro. En la figura 4.3 se muestra el giro de izquierda a derecha, en escalones ascendentes; en la zona de operación de abajo, se observa

más ganancia que arriba. En la Figura 4.4, se muestra el giro de derecha a izquierda, en escalones descendentes; en la zona de operación de arriba, se observa más ganancia que abajo.

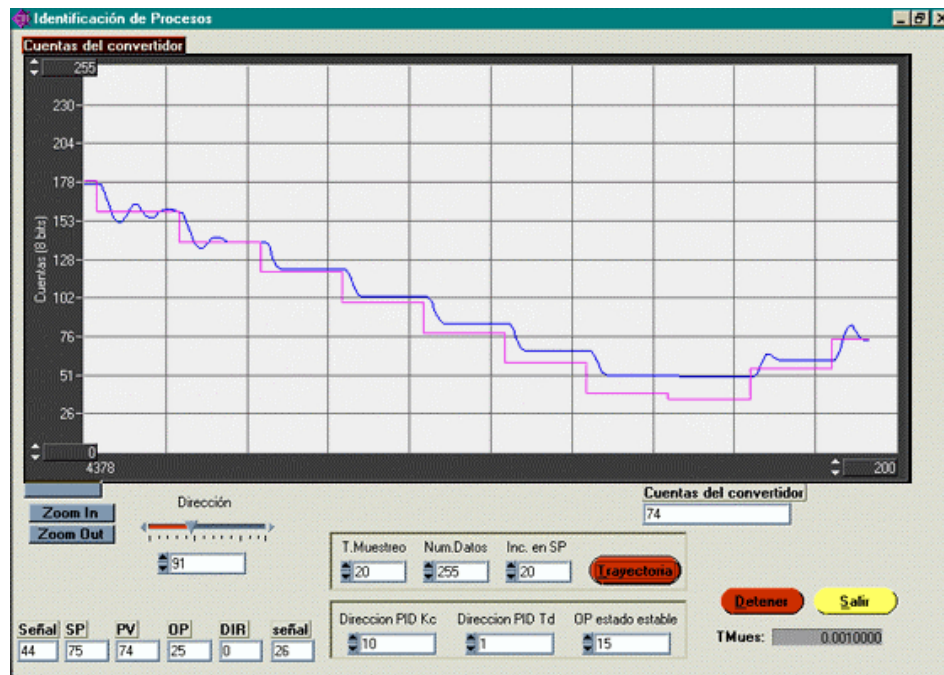


Figura 4.4: Pruebas escalón en forma descendente al controlador PD de la dirección. En la zona de operación de arriba se observa una ganancia grande, mientras que abajo se observa muy poca ganancia

El comportamiento no-lineal se debe a la pieza que une la flecha del motor con el mecanismo que mueve las llantas de dirección. La pieza funciona como un resorte en los dos extremos del movimiento.

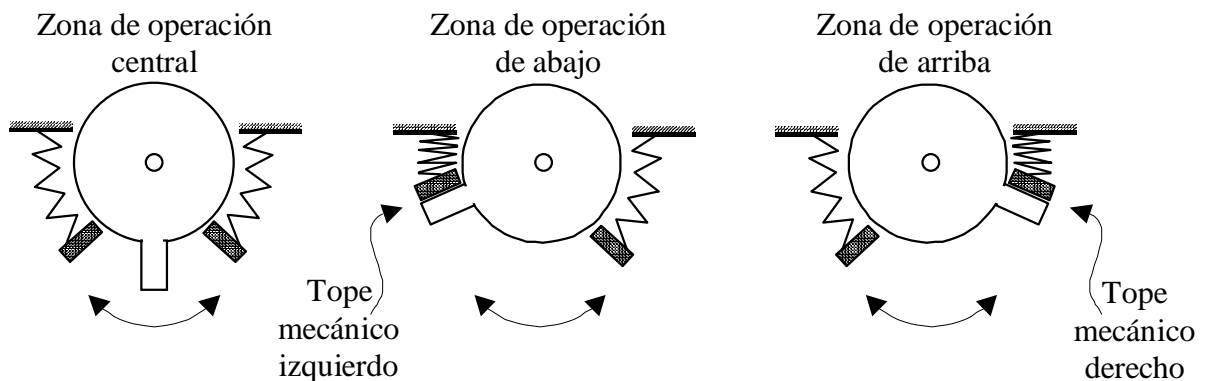


Figura 4.5: Mecanismo equivalente al mecanismo de dirección. La pieza que une la flecha del motor con la pieza que mueve las llantas, actúa como un resorte en los dos extremos de los tope mecánicos

La figura 4.5, muestra un mecanismo equivalente al de dirección; al aproximarse la flecha del motor al tope mecánico izquierdo, la fuerza del resorte actúa en contra del movimiento, y por lo tanto, la ganancia disminuye; en cambio al girar de regreso a la posición central una vez que está en el tope izquierdo, la fuerza del resorte está a favor del movimiento y por lo tanto, la ganancia aumenta. Lo mismo sucede al girar hacia el tope mecánico del lado derecho.

4.1.2 Ganancia preprogramada

Para mejorar el desempeño del controlador ante estas no-linealidades, se implementó un esquema de control adaptable conocido como “Ganancia preprogramada”. Este esquema de control divide el rango de operación en diferentes zonas; cada zona considera una sintonía diferente en el controlador. Además se dividió la operación en dos grandes regiones: cambios en referencia positivos y cambios negativos, para satisfacer la no-linealidad cuando va en sentido ascendente de forma semejante que, cuando va en sentido descendente.

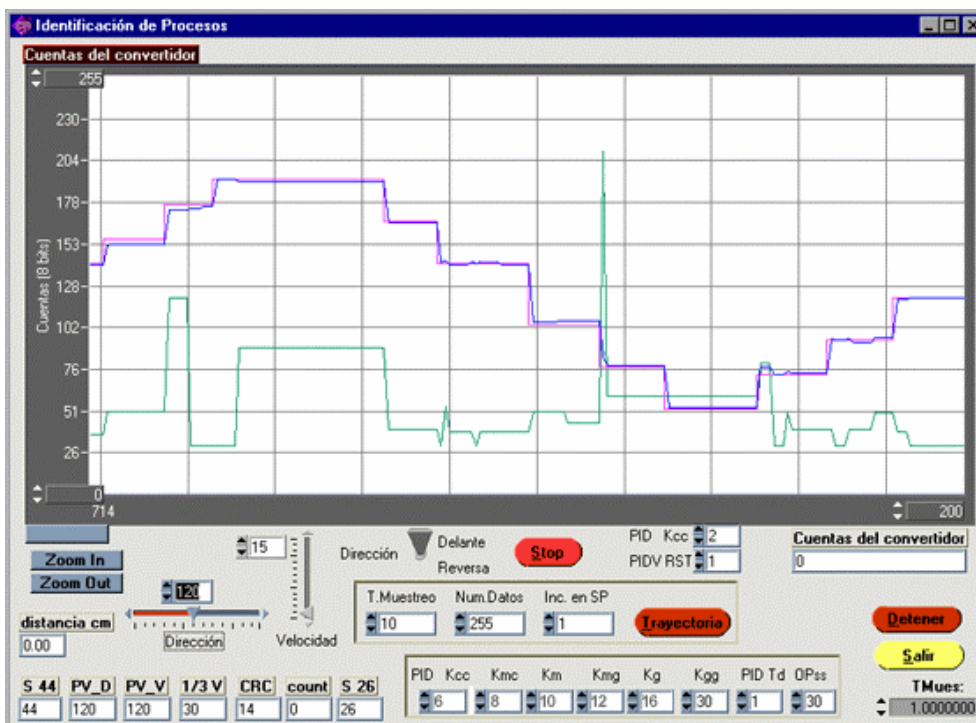


Figura 4.6: Pruebas escalón en el control de dirección utilizando un controlador PD con ganancia preprogramada. El desempeño mejora en toda la zona de operación y en los dos sentidos de giro (ascendente y descendente).

La ganancia preprogramada se divide en seis diferentes zonas de operación. La sintonía sólo cambia la ganancia proporcional K_c , la ganancia derivativa permanece igual en todas las zonas. Para verificar si el cambio es ascendente o descendente, se utiliza el valor de

referencia anterior y se compara con el actual. Las ganancias aumentan conforme a la zona de operación, desde muy chica, hasta muy grande. Por ejemplo, si la operación avanza en forma ascendente y se localiza en la zona de abajo, se utiliza la ganancia muy chica, si se localiza en la zona media, la ganancia mediana; y si se localiza en la zona de arriba, se utiliza la ganancia muy grande. Se probó este algoritmo y como resultado se obtuvo un mejor desempeño al que se tenía anteriormente. La figura 4.6 muestra una prueba en lazo cerrado, utilizando la técnica de ganancia preprogramada implementada en el microcontrolador.

La sintonía del controlador de dirección se realiza mediante un análisis de prueba y error; sin embargo su desempeño fue satisfactorio.

Se utiliza un tiempo de muestreo lo más pequeño posible, ya que la constante de tiempo del sistema de dirección es muy pequeña. El tiempo de muestreo equivale a un ciclo en el microcontrolador, debido a que se utilizó un microcontrolador muy rápido. Este tiempo de muestreo depende directamente de la velocidad con que se actualizan los datos, en el convertidor análogo digital, que lee la posición de la dirección.

4.2 Control a un punto

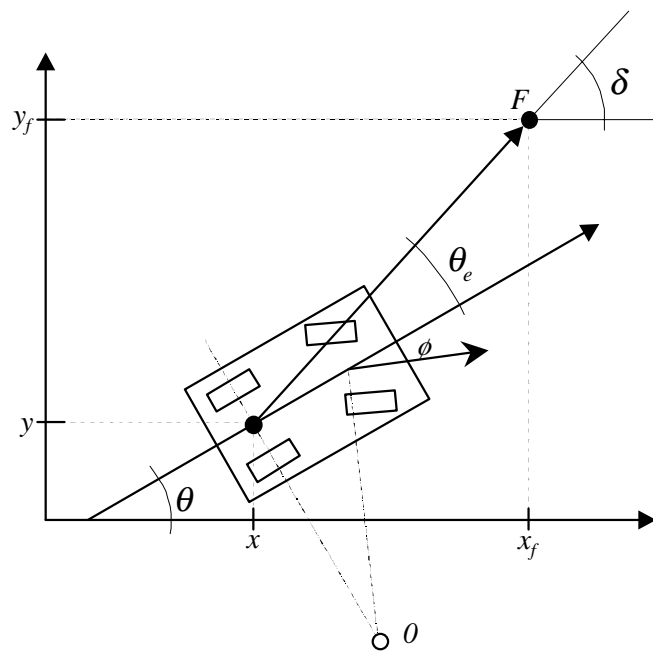


Figura 4.7: Seguimiento de un punto objetivo F. La variable controlada es θ , la variable manipulada es ϕ , y el error es θ_e . Si el error es cero, el vehículo se dirige directo al punto.

El objetivo del control para el seguimiento a un punto, es llevar el vehículo al punto $F=(x_f, y_f)$ como lo indica la figura 4.7, de forma que las coordenadas (x, y) que definen la

posición actual del vehículo sean iguales a (x_f, y_f) que definen la posición del punto objetivo. El controlador a un punto se encarga de manipular el ángulo de dirección con el fin de orientar el vehículo hacia el punto $F=(x_f, y_f)$ en el plano, conforme avanza el vehículo. La variable controlada es el ángulo θ , que representa la orientación del vehículo en el plano. La restricción cinemática no-holonómica, no permite cambiar la orientación del vehículo, sin cambiar también su posición, por lo que se requiere que el vehículo avance para satisfacer el objetivo de control.

Para implementar un sistema de control, debe de existir por lo menos una variable a controlar, un error a minimizar y una variable manipuladora que provoque un cambio en la variable a controlar. Con el sistema de control propuesto, se cuenta con todos estos elementos, aunque la relación entre estas variables es completamente no-lineal; la variable a controlar es θ , la variable manipulada es ϕ y el error es θ_e .

El error es el ángulo que se forma entre el vector en la orientación actual del vehículo θ y el vector que une el punto de la posición actual del vehículo con la posición del punto a donde se desea llegar; si el error es cero, entonces el vehículo se dirige directamente hacia el punto. En la Figura 4.7 se muestra gráficamente el cálculo del error.

El cálculo del error θ_e es la relación:

$$\theta_e = \delta - \theta \quad (4.3)$$

$$\delta = \tan^{-1} \left(\frac{y_f - y}{x_f - x} \right)$$

donde δ es el ángulo del vector que une el punto actual del vehículo y el punto objetivo F , el se calcula por métodos geométricos; motivo por lo cuál, el controlador no es lineal. Es importante aclarar que depende de la posición (x, y) , por lo cuál, esta variable sólo cambia cuando el vehículo avanza.

En el cálculo del error, se supone que las variables que definen la posición y orientación del vehículo (x, y, θ) están disponibles en todo momento, ya sea medidas físicamente o calculadas por medio de una estimación.

La utilización de estas variables, para minimizar el error, depende del algoritmo de control utilizado. En esta investigación se seleccionaron los controladores PID, para realizar esta tarea, por su fácil implementación en el microcontrolador; sin embargo, este sistema debiera funcionar con otros algoritmos de control. Los controladores PID calculan la variable manipuladora con base en una acción proporcional, integral y derivativa del error.

Como se mencionó, el controlador de seguimiento a un punto, mantiene una estructura de cascada con el controlador de dirección. El control de dirección es esclavo, puesto que recibe, como valor deseado de posición, la variable manipulada del sistema de control para seguimiento a un punto.

El cálculo de la variable de manipulación del controlador para seguimiento a un punto, se obtiene de la relación:

$$\phi_d = k_t \theta_e \quad (4.4)$$

donde ϕ_d es la variable manipuladora y k_t es el factor de la acción proporcional.

La sintonía del controlador, depende únicamente del factor de la acción proporcional; entre más pequeño sea este factor, más lentas son las correcciones, por lo que, la curva que se forma para llegar al punto, tiene oscilaciones más grandes, como se muestra en la figura 4.8.

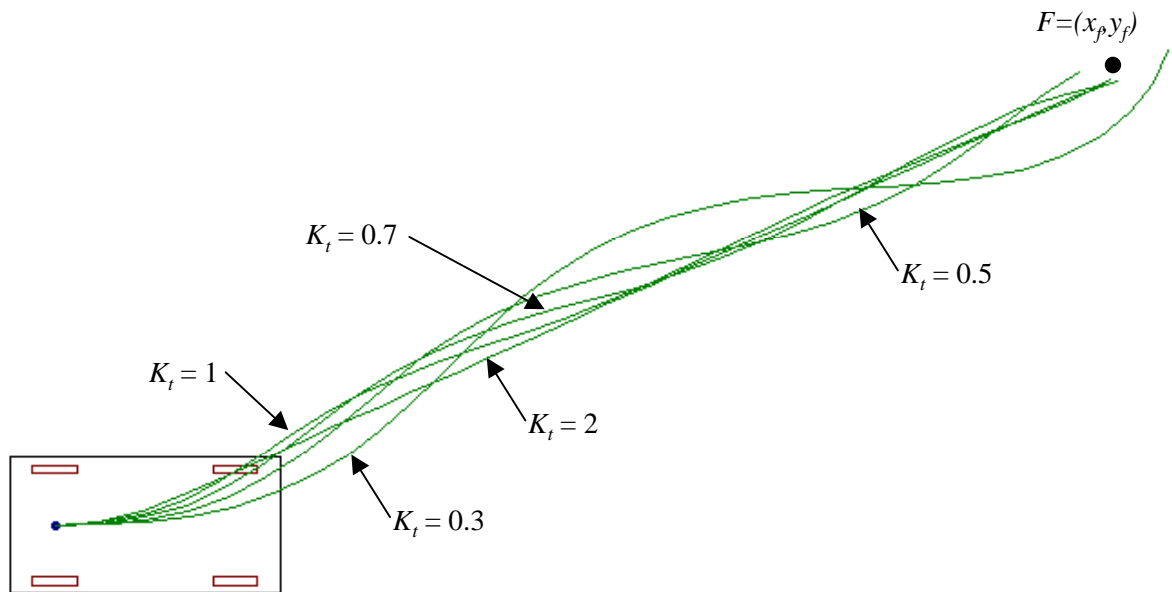


Figura 4.8: Curvas generadas con diferentes sintonías del controlador para el seguimiento a un punto

Si el factor de la acción proporcional es muy grande, el sistema se vuelve inestable, ya que con tan solo un pequeño error, la corrección sería muy grande, incluso hasta el tope, además, si el error varía entre positivo y negativo, la dirección giraría inestablemente desde el tope izquierdo hasta el tope derecho y viceversa.

El tiempo de muestreo también es muy importante, ya que la curva generada depende de la rapidez con que se corrige el error. Como el error se especifica en unidades de distancia, la variación en la curva depende de la distancia recorrida desde el muestreo anterior; si se utilizara un muestreo en unidades de tiempo, la distancia recorrida dependería de la velocidad. Si la distancia recorrida entre cada muestreo es muy grande, entonces el vehículo oscila, como se muestra en la figura 4.9.

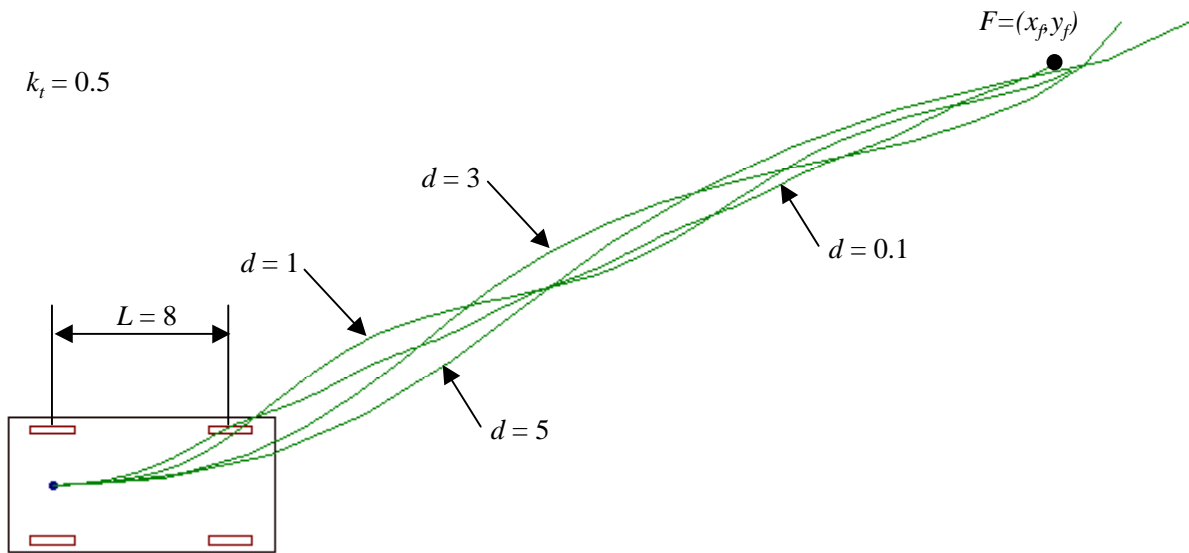


Figura 4.9: Curvas generadas con diferentes distancias de muestreo, en el controlador para el seguimiento a un punto

En la figura 4.9 se muestra el caso de un vehículo que mide 8 unidades y el controlador se sintoniza con un factor proporcional $k_t = 0.5$, el muestreo está en unidades de distancia. Se observa que, para distancias de muestreo menores al 50% de L , el comportamiento es aceptable. También se observa que, para distancias de muestreo muy pequeñas, el vehículo se dirige directo al punto, sin oscilar.

4.3 Control de velocidad

El controlador de velocidad, calcula las manipulaciones para el motor de velocidad, con el fin de llevar el vehículo a la velocidad deseada. El sistema de velocidad tiene como modelo un sistema de primer orden (ver Capítulo 2).

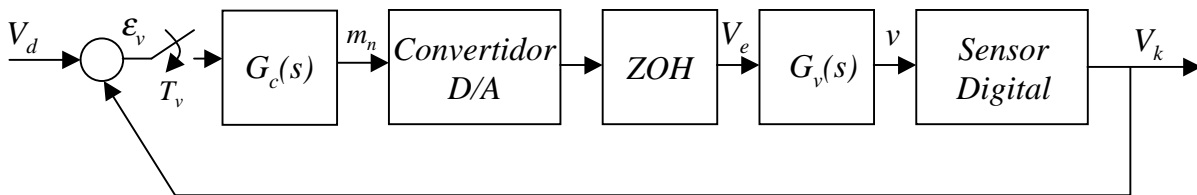


Figura 4.10: Lazo de control digital para la velocidad; $G_c(s)$ es la función de transferencia del controlador y $G_v(s)$ es el modelo del sistema

El controlador se encarga de llevar la velocidad del vehículo a un valor deseado de referencia y mantenerla. El controlador de velocidad se implementa en un microcontrolador,

como se vió en el capítulo anterior; el lazo de control digital se muestra en la figura 4.10.

Se propone utilizar un controlador PI digital en modo velocidad, con cálculo de la variable manipulada m_n ,

$$m_n = m_{n-1} + K_c \left[\varepsilon_n - \varepsilon_{n-1} + \frac{T}{\tau_i} \left(\frac{\varepsilon_n + \varepsilon_{n-1}}{2} \right) \right] \quad (4.5)$$

donde K_c es la ganancia proporcional, τ_i es la ganancia integral, T es el tiempo de muestreo, e_n es el error actual, e_{n-1} es el error un tiempo de muestreo anterior, mientras que m_{n-1} es la manipulación un tiempo de muestreo anterior.

Como se observa en la ecuación 4.5, cuando el error es igual a cero, la manipulación se mantiene en el valor anterior. La acción integral se agrega para eliminar el error de estado estable, ya que el sistema no tiene integrador.

4.4 Estimación de parámetros (x, y, θ)

La estimación es importante para el sistema de control, puesto que el desempeño general depende de una buena estimación de la posición y orientación del vehículo. Si no se conoce la posición y orientación actual, por medio de sensores externos, es entonces necesario estimarlos por medio de los sensores disponibles.

Los únicos sensores disponibles son el potenciómetro, que mide el ángulo actual de las llantas, y el codificador óptico, que mide la distancia recorrida. Si se conoce el perímetro de las llantas, se calcula la distancia que recorre en cada vuelta. Así mismo, si conocemos el número de pasos por vuelta de llanta, se calcula la distancia que recorre por cada paso en el codificador.

Considerando su modelo cinemático, cuando conocemos la distancia que recorrió el vehículo, con cierta inclinación de llantas ϕ , se calcula la inclinación del vehículo ($\Delta\theta$), el avance en el eje x (Δx) y el avance en y (Δy). Si estos cambios se acumulan, se obtiene la posición y orientación del vehículo en todo momento. A este método de estimación se le conoce como conteo interno o “dead-reckoning”.

Estos cálculos se realizan con funciones trigonométricas, gracias al análisis de la geometría del carro. Recordando el modelo cinemático de la figura 2.1 y tomando un tiempo de muestreo, se analizan los cambios del vehículo en su posición y orientación. La figura 4.11 ilustra el diagrama geométrico para el cálculo de estos cambios.

Se observa que los cambios en x , y y θ dependen directamente de la posición actual de las llantas delanteras y de la longitud del vehículo. A mayor longitud, el radio de giro es más grande y las vueltas resultan más abiertas. Para el cálculo de los cambios en las variables, se supone que entre los dos instantes de muestreo, el giro es constante.

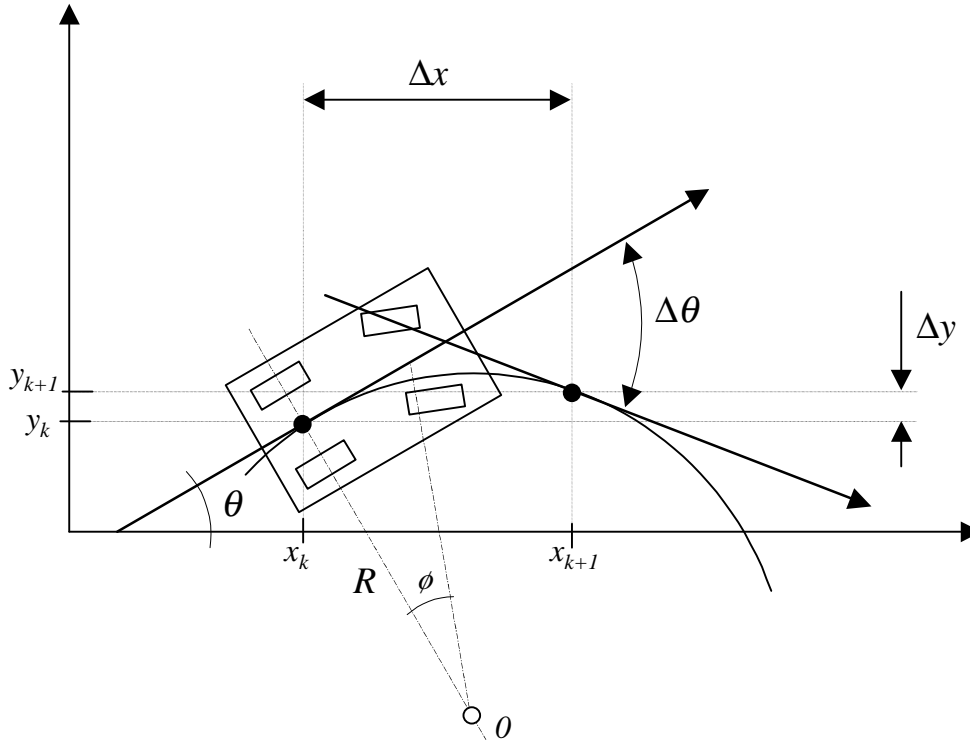


Figura 4.11: Diagrama geométrico para el cálculo de Δx , Δy , $\Delta\theta$. Se ilustran dos instantes del muestreo: el instante k y el instante $k+1$; se supone que entre esos dos muestreos, el giro es constante, por ello no hubo cambio en ϕ

Para calcular $\Delta\theta$, Δx y Δy , primero se calcula el cambio en la inclinación del cuerpo del vehículo θ ,

$$\Delta\theta = \frac{d}{R} = \frac{d \tan \phi}{L} \quad (4.6)$$

donde d es la distancia lineal recorrida del vehículo, L es la longitud entre los ejes, R es el radio del círculo que describe el movimiento del vehículo, mientras que ϕ es el ángulo actual, que adquiere un valor positivo cuando el vehículo se inclina hacia la izquierda y negativo cuando se inclina hacia la derecha.

Una vez realizado el cambio en la variable θ , se calculan los cambios en la posición,

$$\begin{aligned} \Delta x &= R[\sin(\theta + \Delta\theta) - \sin(\theta)] \\ \Delta y &= R[-\cos(\theta + \Delta\theta) + \cos(\theta)] \end{aligned} \quad (4.7)$$

Así, se obtiene el incremento en la posición y orientación del vehículo en cada instante de muestreo. Si el muestreo se define en unidades de tiempo, entonces la distancia lineal recorrida depende de la velocidad. Si el muestreo, en cambio, se realiza en unidades de distancia, se simplifica el cálculo de la ecuación 4.6, ya que la distancia es constante.

4.4.1 Aproximaciones para su implementación

La estimación de los parámetros es la parte más crítica en la implementación, según la arquitectura de control propuesta en esta investigación. Si se tiene una estimación precisa de la posición real, el recorrido real del vehículo será igual al deseado.

Para implementar un algoritmo que estime la posición utilizando las ecuaciones 4.6 y 4.7, se requieren funciones trigonométricas, además de cálculos matemáticos con precisión de punto flotante.

Para simplificar los cálculos en un microcontrolador, la curva que describe el vehículo se aproxima por medio de pequeñas líneas rectas. A continuación se analizan tres métodos posibles para simplificar los cálculos.

Método 1: Avance en dirección anterior

Este método utiliza un vector que se dirige un ángulo θ con origen en la posición anterior. El método se basa en la suposición, de que si se toma un trozo suficientemente pequeño del arco que describe el vehículo, se aproxima a una línea recta.

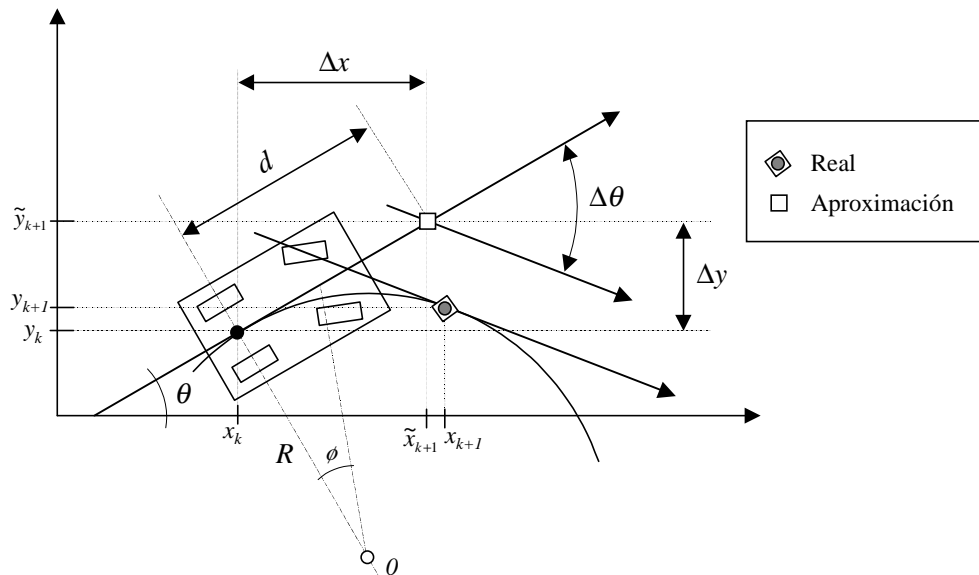


Figura 4.12: Aproximación por el Método 1. Se utiliza un vector con inclinación θ para el cálculo de Δx y Δy

Para el cálculo del cambio en la posición con este método, se sustituyen las ecuaciones (4.6), (4.7) por las siguientes ecuaciones:

$$\Delta\theta = \frac{d}{R} = \frac{d \tan \phi}{L} \quad (4.8)$$

$$\begin{aligned}\Delta x &= d[\cos(\theta)] \\ \Delta y &= d[\sin(\theta)]\end{aligned}\quad (4.9)$$

Con esta aproximación se traza una línea recta a una inclinación θ , y se avanza, sobre esa recta, una distancia d . En la figura 4.12 se hace una comparación gráficamente de esta aproximación, con respecto al movimiento real. Nótese que entre más pequeña sea la distancia d , la posición calculada por la aproximación, es más cercana a la realidad.

En la figura 4.12 se exagera la curvatura, además, se toma una distancia grande, con el fin ilustrativo de mostrar los dos puntos: el real y el aproximado.

Método 2: Avance en dirección posterior

Este método utiliza un vector con una inclinación $\theta + \Delta\theta$ con origen en la posición anterior del vehículo. Este método se basa en la misma suposición de aproximar un arco con una línea recta. Para el cálculo de la posición, se sustituyen las ecuaciones (4.6) y (4.7) por las siguientes ecuaciones:

$$\Delta\theta = \frac{d}{R} = \frac{d \tan \phi}{L} \quad (4.10)$$

$$\begin{aligned}\Delta x &= d[\cos(\theta + \Delta\theta)] \\ \Delta y &= d[\sin(\theta + \Delta\theta)]\end{aligned}\quad (4.11)$$

Con esta aproximación se traza una línea recta a una inclinación $\theta + \Delta\theta$, y se avanza sobre esa recta una distancia d .

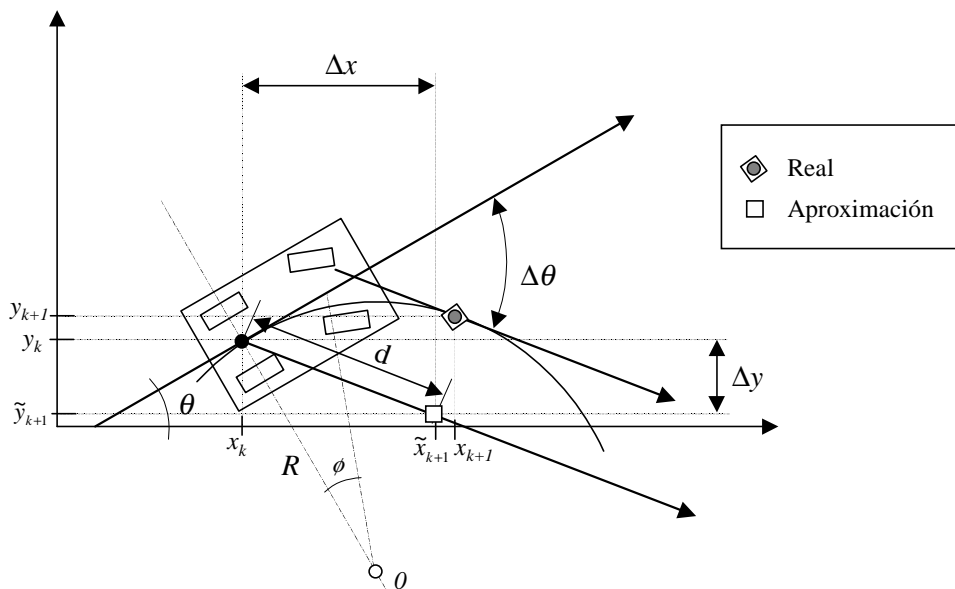


Figura 4.13: Aproximación por el Método 2. Se utiliza un vector con inclinación $\theta + \Delta\theta$ para el cálculo de Δx y Δy

En la figura 4.13 se comparan gráficamente la posición calculada con esta aproximación, respecto al movimiento real. Nótese también que, entre más pequeña sea la distancia d , la aproximación es más cercana a la realidad. También, se exagera la curvatura, además, se toma una distancia grande, con el fin ilustrativo de mostrar los dos puntos, el real y el aproximado.

Método 3: Avance en dirección promedio

Este método traza un vector con una inclinación $\theta + (\Delta\theta/2)$, con origen en la posición anterior. La longitud del vector es la distancia d recorrida por el vehículo. Para el cálculo de la posición, se sustituyen las ecuaciones (4.6) y (4.7) por las siguientes ecuaciones:

$$\Delta\theta = \frac{d}{R} = \frac{d \tan \phi}{L} \quad (4.12)$$

$$\begin{aligned} \Delta x &= d \left[\cos \left(\theta + \frac{\Delta\theta}{2} \right) \right] \\ \Delta y &= d \left[\sin \left(\theta + \frac{\Delta\theta}{2} \right) \right] \end{aligned} \quad (4.13)$$

En la figura 4.14 se ilustra gráficamente esta aproximación, respecto al movimiento real. También, entre más pequeña sea d , la aproximación es más cercana a la realidad.

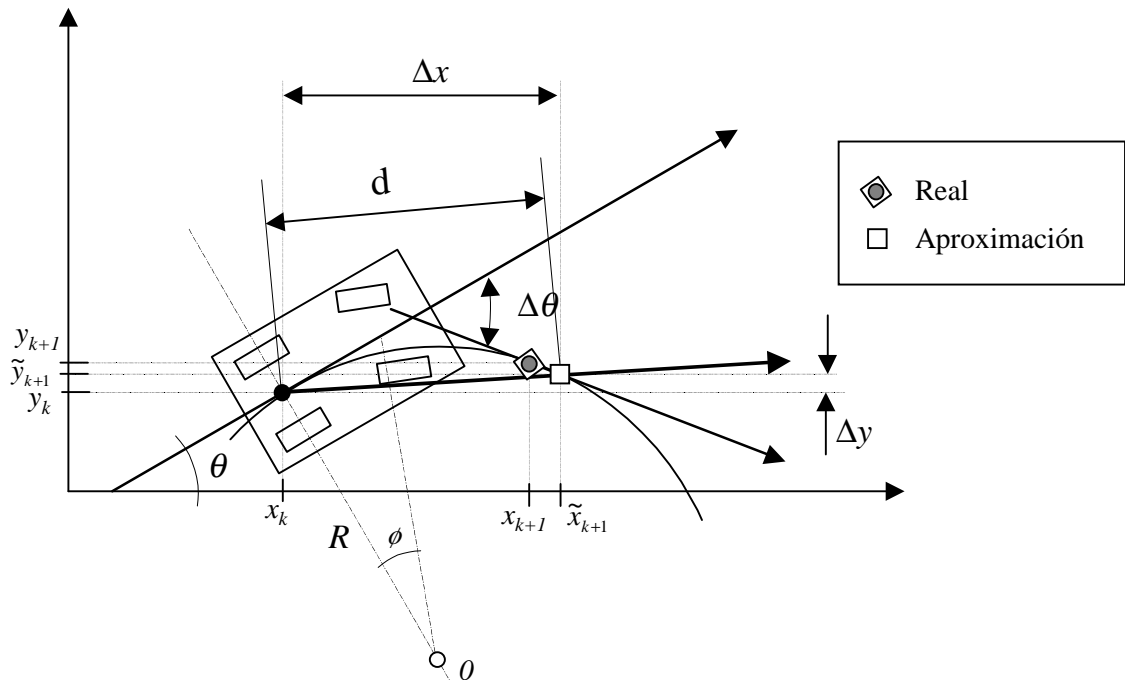


Figura 4.14: Aproximación por el método 3. Se utiliza un vector con una inclinación $\theta + \Delta\theta/2$ para el cálculo de Δx y Δy . En la gráfica se muestran la posición real y la aproximación

Nuevamente, en la figura 4.14 se exagera en el error con fines ilustrativos. Entre más pequeña sea la distancia d y menos pronunciada sea la curva, el error es mucho más pequeño.

4.4.2 Comparación de los métodos de aproximación

Con el fin de hacer una comparación y de ilustrar cómo la aproximación mejora entre más pequeña sea d , se probaron estos métodos en un simulador. Se introdujeron las medidas reales del vehículo utilizado en las pruebas. Se realizó primero, una prueba utilizando una distancia d equivalente a una vuelta completa de llanta. Los resultados se muestran en la figura 4.15, donde solo se muestra el resultado utilizando las aproximaciones por el método 2 y el método 3. El método 2 se grafica con una equis: “X”, mientras que el método 3, con un cuadro, y la posición real con un punto: “•”. La prueba se realizó avanzando el vehículo con una inclinación en las llantas de 17° (máxima inclinación en el vehículo utilizado para las pruebas).

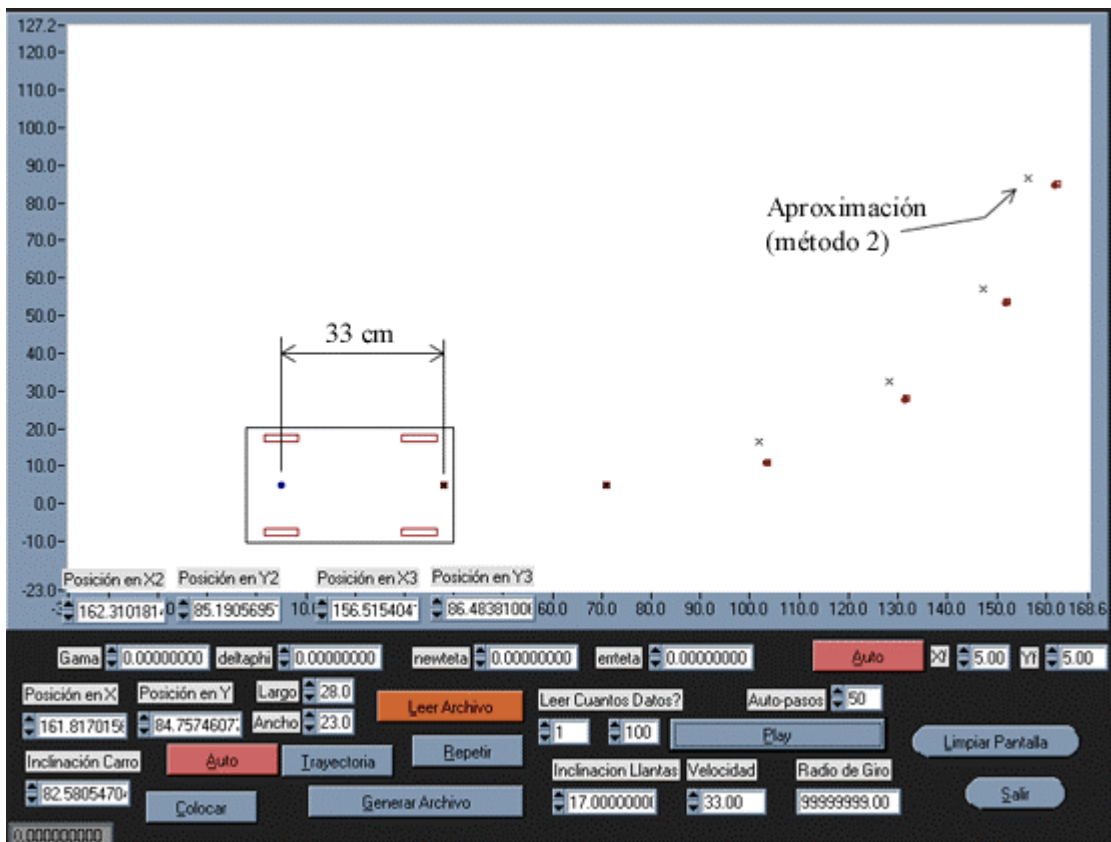


Figura 4.15: Prueba en el simulador con el Método 2 y Método 3 de estimación de la posición; utilizando la distancia d de una vuelta de llanta

Después se redujo d al equivalente de $1/3$ de vuelta de llanta. Los resultados se muestran en la figura 4.16; al igual que en la figura 4.15, solo se muestra el resultado de los cálculos utilizando el Método 2 y el Método 3. Finalmente, se puede observar claramente como el

error se reduce considerablemente al reducir d .

Utilizando las medidas reales del vehículo, se obtiene un error máximo de 4 cm, por cada 2 metros, avanzando a máxima curvatura y midiendo la distancia cada vuelta de llanta. Por otro lado, usando los mismos parámetros pero midiendo la distancia cada 1/3 de vuelta de llanta, se obtiene un error de 0.5 cm, cada 2 metros. Este error, es el que se introduce únicamente por la aproximación, los cuáles fueron calculados en simulación. Para obtener el error real, a este error se le agregan todos los factores que no se consideran en la simulación, como son, errores en la lectura de los sensores, partes mecánicas flojas, etc.

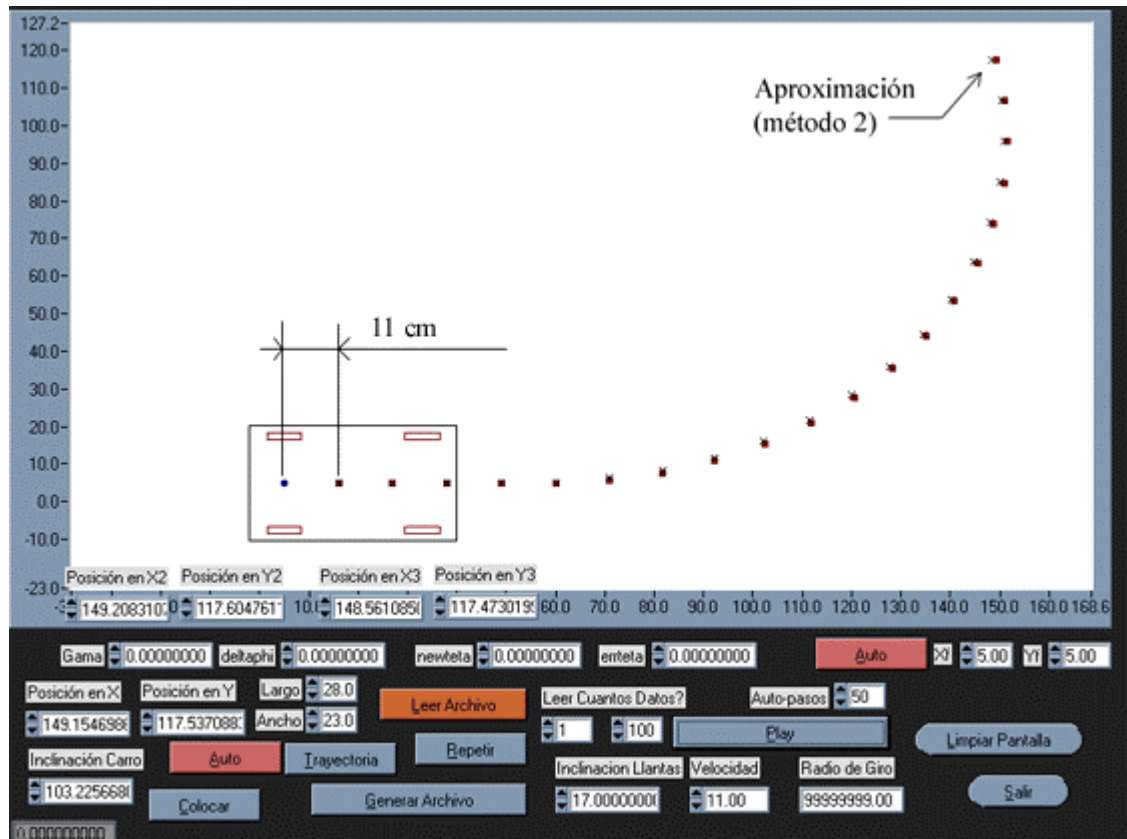


Figura 4.16: Prueba en el simulador con el Método 2 y el Método 3 de estimación de la posición; utilizando la distancia d de 1/3 de vuelta de llanta

4.5 Control de trayectoria

Para seguir una trayectoria, que se define por medio de puntos, es necesario un algoritmo que reconozca cuándo el carro llegó a un punto y ordene, al controlador para el seguimiento de un punto, avanzar al siguiente punto en la trayectoria. Este algoritmo verifica la distancia desde la posición actual del vehículo al punto; si esa distancia es cero, significa que el vehículo alcanzó la posición del punto.

Para el cálculo de la distancia entre el vehículo y el punto actual, se utiliza la siguiente ecuación:

$$d(P, P_f) = \sqrt{(x_f - x)^2 + (y_f - y)^2} \quad (4.14)$$

donde d es la distancia, $P_f=(x_f, y_f)$ son las coordenadas del punto y $P=(x, y)$ son las coordenadas de la posición actual del vehículo. La condición para avanzar al siguiente punto, es que la distancia sea igual o cercana a cero.

4.5.1 Puntos objetivo inalcanzables

Existe una zona, a la cuál, el controlador para el seguimiento de un punto, no puede llevar al vehículo. Esta zona se localiza en la parte izquierda y en la parte derecha del vehículo, dentro de dos círculos con valor del radio mínimo de la curvatura alcanzable; dicho radio mínimo se establece debido a los topes mecánicos de la dirección. La figura 4.17 muestra gráficamente la localización de esta zona.

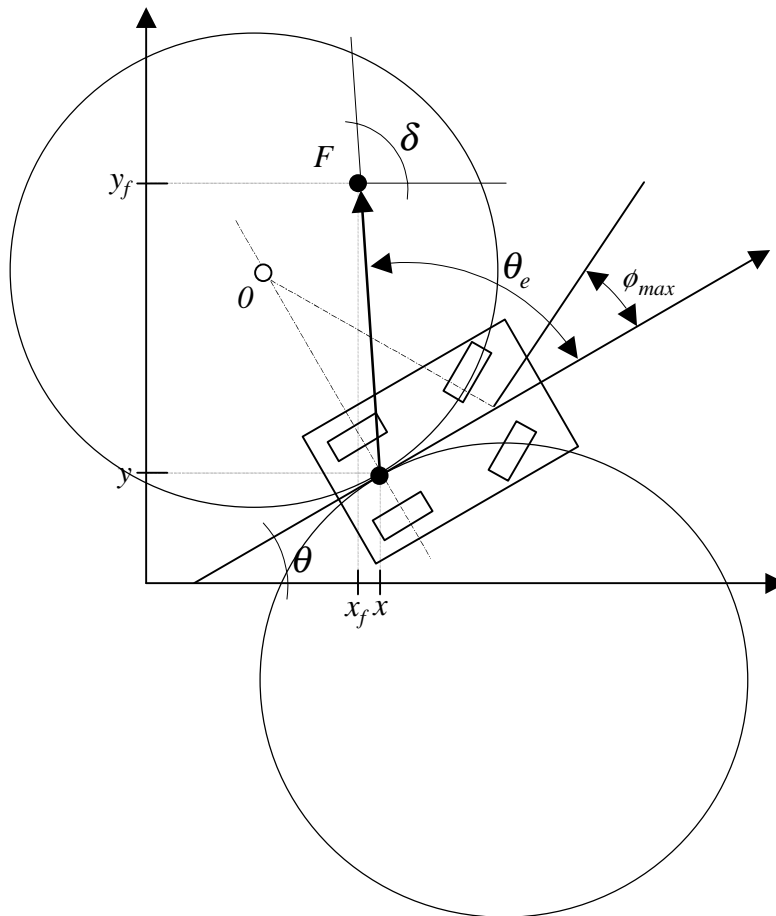


Figura 4.17: El vehículo no llega con el algoritmo de control propuesto, a un punto objetivo dentro de la zona circular. El radio del círculo está dado por R_{min} que se presenta al tener ϕ_{max}

Cualquier punto que caiga dentro del círculo acotado por el radio mínimo, provoca que el vehículo permanezca dando vueltas alrededor del punto, ya que la distancia entre la posición actual del vehículo al punto, nunca será cero, según se describe en la figura 4.17.

Para verificar si un punto cae dentro de esta zona, primero se calcula el radio de giro mínimo, cuyo valor es constante y depende del lugar donde se coloque el tope mecánico de la dirección,

$$R_{\min} = \frac{L}{\tan \phi_{\max}} \quad (4.15)$$

donde L es la longitud que existe entre los ejes delantero y trasero del vehículo, ϕ_{\max} es la inclinación máxima en las llantas, ya sea hacia la izquierda o hacia la derecha.

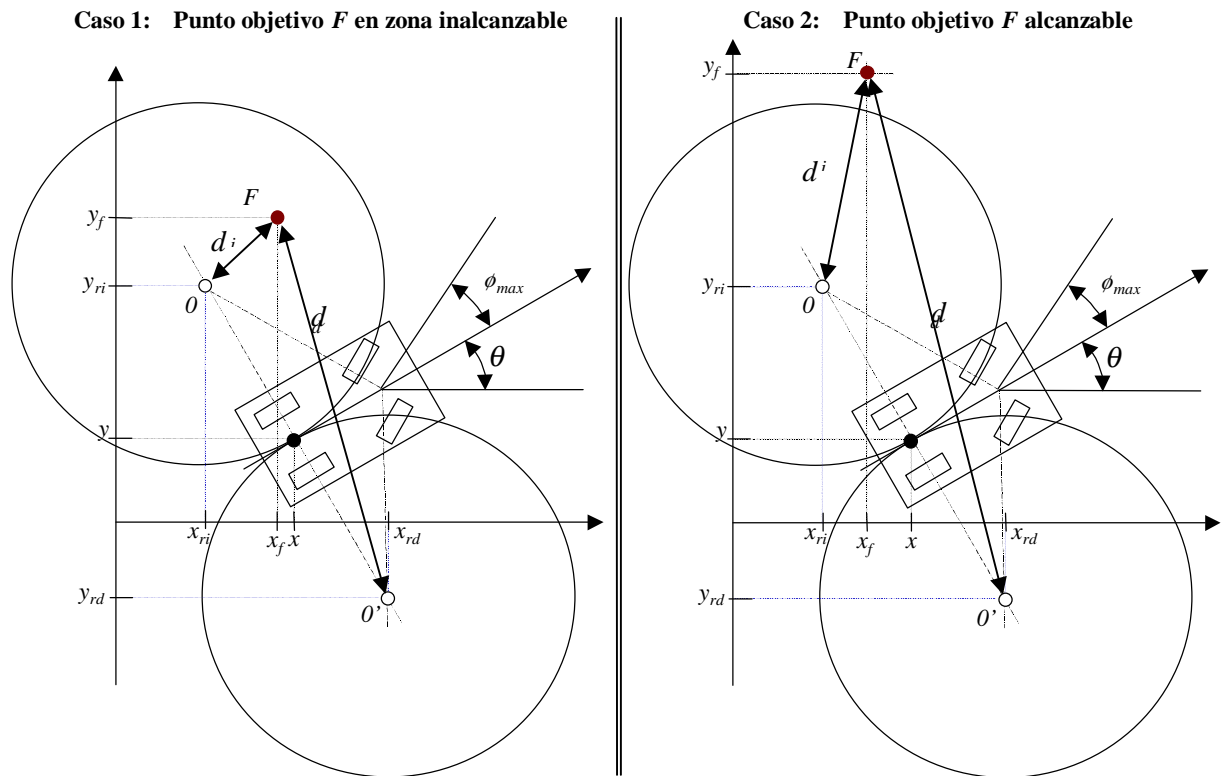


Figura 4.18: Caso 1, el cálculo de la distancia, determina que el punto está en la zona inalcanzable. Caso 2, el cálculo de la distancia, determina que el punto es alcanzable

Después se calculan las coordenadas del centro del círculo a distancia R_{min} de la posición del carro. Como hay dos círculos, uno a la izquierda y otro a la derecha, se calculan los dos centros; para el círculo izquierdo:

$$\begin{aligned}x_{ri} &= x + R_{min} \cos(\theta + 90^\circ) \\y_{ri} &= y + R_{min} \sin(\theta + 90^\circ)\end{aligned}\tag{4.16}$$

y para el círculo derecho:

$$\begin{aligned}x_{rd} &= x + R_{min} \cos(\theta - 90^\circ) \\y_{rd} &= y + R_{min} \sin(\theta - 90^\circ)\end{aligned}\tag{4.17}$$

donde (x,y) es la posición actual del vehículo, (x_{ri},y_{ri}) son las coordenadas del centro del círculo izquierdo y (x_{rd},y_{rd}) son las coordenadas del centro del círculo derecho.

Enseguida, se calcula la distancia que existe desde cada centro hasta el punto objetivo, si una de estas distancias es menor a R_{min} , se concluye que el punto objetivo está dentro de la zona inalcanzable.

$$\begin{aligned}d_i &= d(P_f, P_{ri}) \\d_d &= d(P_f, P_{rd})\end{aligned}\tag{4.18}$$

donde $P_f=(x_f,y_f)$ son las coordenadas del punto objetivo, $P_{ri}=(x_{ri},y_{ri})$ son las coordenadas del centro del círculo izquierdo, $P_{rd}=(x_{rd},y_{rd})$ son las coordenadas del centro del círculo derecho, d_i es la distancia del centro del círculo izquierdo al punto objetivo, mientras que d_d es la distancia del centro del círculo derecho al punto objetivo.

La figura 4.18 presenta dos casos; para el Caso 1, el punto objetivo dentro de la zona inalcanzable, mientras que para el Caso 2, el punto objetivo fuera de la zona inalcanzable. Se observa como en el Caso 1, la distancia d_i es menor al radio del círculo R_{min} , mientras que en el caso 2, ninguna de las distancias es menor.

Si en la trayectoria dada por el usuario, se presenta el Caso 1, entonces, la distancia al punto objetivo nunca sería cero o cercana a cero, con el actual algoritmo de control de trayectoria. Para evitar que el vehículo permanezca girando alrededor del punto, se requiere mejorar el algoritmo de control de trayectoria.

4.5.2 Mejoras al algoritmo de control de trayectoria

En esta sección se presentan dos estrategias para mejorar el algoritmo de control de trayectoria, con el fin de evitar que el vehículo permanezca girando alrededor de un punto objetivo inalcanzable. La primer estrategia consiste en alejar el vehículo hacia el lado contrario, hasta sacar el punto de la zona inalcanzable; recordando que la zona inalcanzable se mueve junto con el vehículo. La segunda estrategia, mide la distancia que falta para llegar al punto, al momento en que esa distancia, en lugar de disminuir, aumenta, se determina que el vehículo ya llegó al punto y pasa al siguiente. A continuación se hace un análisis detallado

de cada una de estas estrategias.

La primer estrategia, requiere determinar si el punto objetivo se encuentra dentro de la zona inalcanzable; en caso afirmativo, invierte el signo de la variable manipuladora del controlador a un punto ϕ_d , la cuál, a su vez, es el valor deseado en el controlador de dirección. Al invertir el signo, el algoritmo provoca que el vehículo se aleje en sentido contrario a donde se encuentra el punto, hasta que el punto ya no sea inalcanzable. La condición, en el algoritmo de control de trayectoria sería modificada por la regla:

$$\begin{aligned} \text{Si } d_i < R_{\min} &\Rightarrow \phi_d = -\phi_{\max} \\ \text{Si } d_d < R_{\min} &\Rightarrow \phi_d = \phi_{\max} \end{aligned} \quad (4.19)$$

donde d_i y d_d son las distancias que definen la zona inalcanzable izquierda y derecha respectivamente (ver ecuación 4.18), R_{\min} es el radio de giro mínimo (ver ecuación 4.15), ϕ_d es el “set point” en el controlador de dirección.

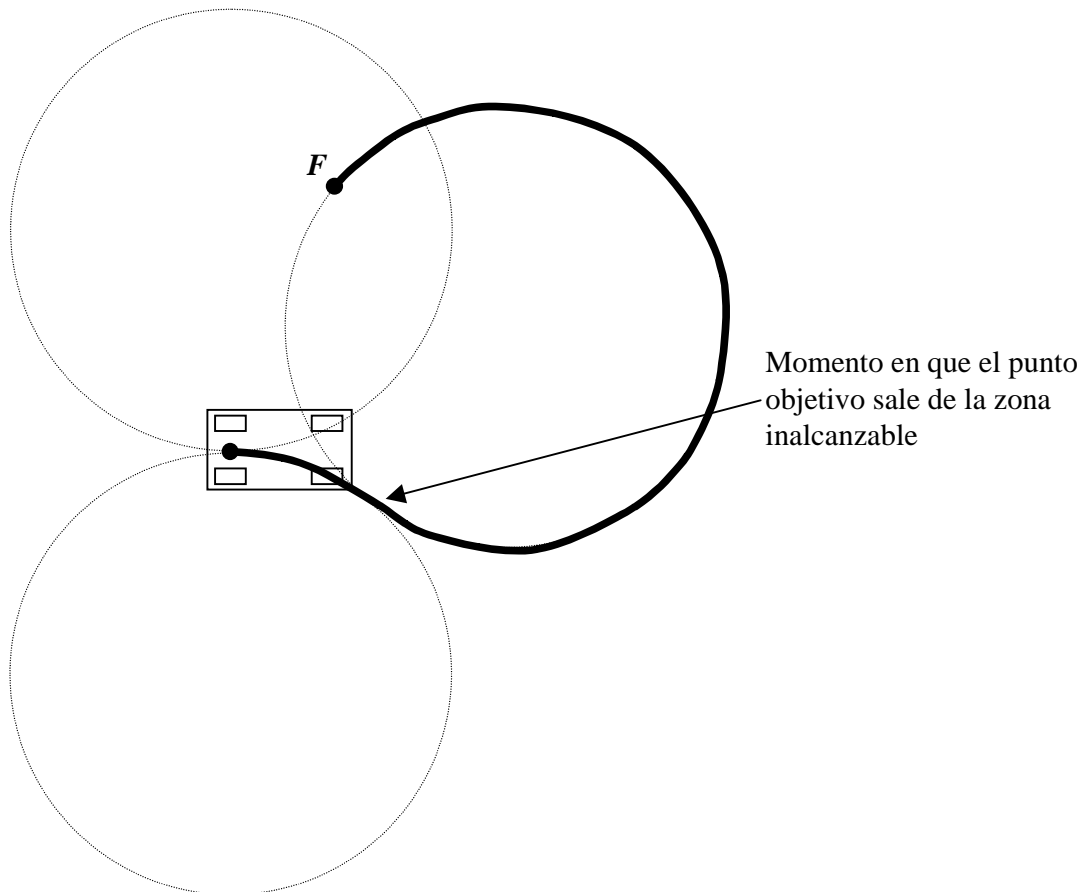


Figura 4.19: Ejemplo de seguimiento a un punto objetivo inalcanzable usando la primer estrategia. Al inicio el vehículo se aleja del punto, hasta el momento en que el punto sale de la zona inalcanzable, definida por los círculos

En la figura 4.19 se muestra el recorrido que haría el vehículo utilizando este algoritmo. Nótese que, en un principio, el punto objetivo se encuentra en la zona inalcanzable de lado izquierdo. El algoritmo comienza avanzando el vehículo hacia el lado contrario del punto, hasta sacar el punto de la zona inalcanzable, para finalmente, llegar al objetivo.

La segunda estrategia, verifica que la distancia del vehículo al punto vaya disminuyendo; al momento en que dicha distancia aumenta, el algoritmo supone que el vehículo alcanzó el punto y pasa al siguiente. Sin embargo, se requiere verificar esta condición, cuando el vehículo se encuentra a una distancia menor del radio de giro mínimo, con el fin de considerar la transición del punto anterior al actual. Además, el vehículo se encuentra orientado en dirección contraria al punto actual. Con esta estrategia, lo único que se requiere es cambiar la condición original: *si la distancia es igual o cercana a cero, por: si la distancia actual es menor a R_{min} y además mayor o igual a la distancia anterior*. Esta estrategia no requiere analizar si el punto objetivo cae dentro de la zona inalcanzable; sin embargo, funciona bien para todos los casos.

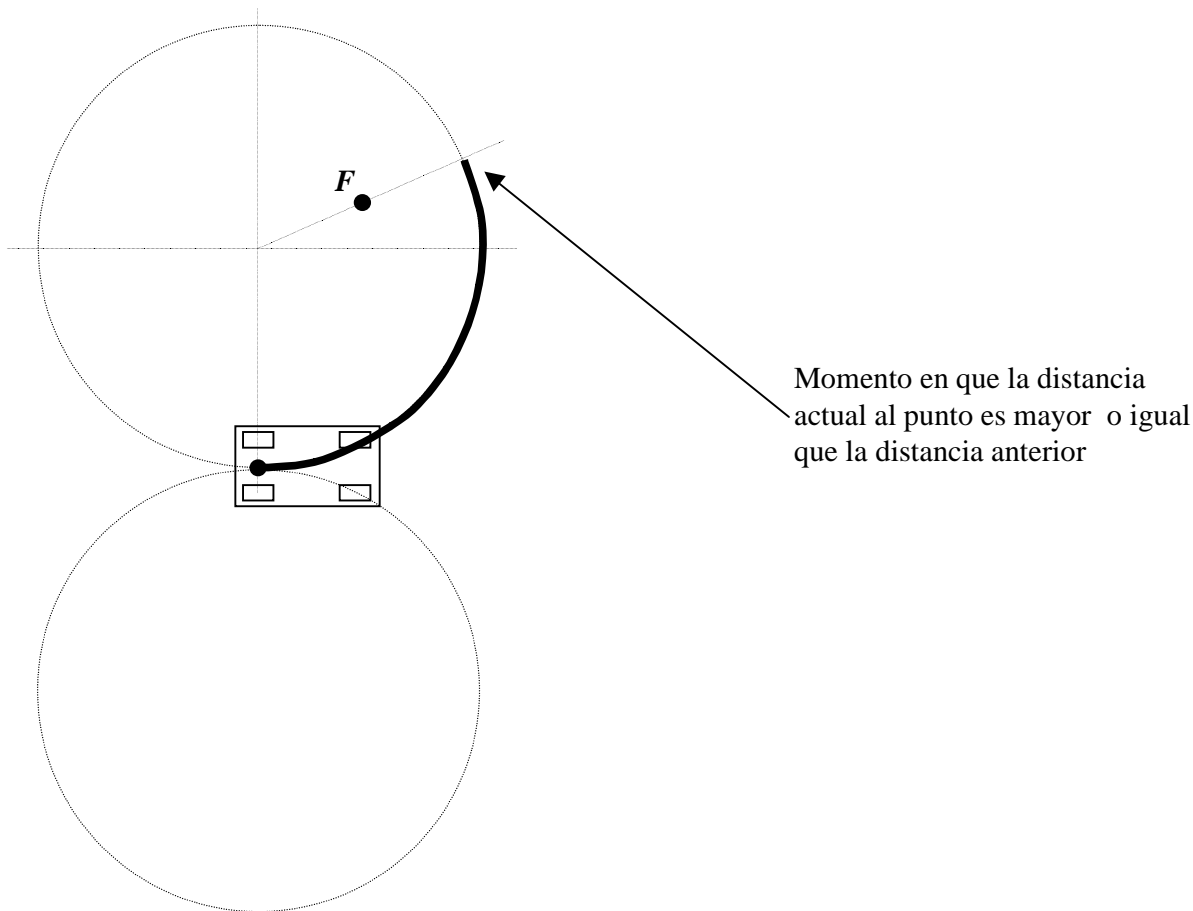


Figura 4.20: Ejemplo de seguimiento a un punto objetivo inalcanzable usando la segunda estrategia

La figura 4.20 muestra el recorrido que haría el vehículo utilizando la segunda estrategia. Nótese que en un principio, el punto objetivo se encuentra en la zona inalcanzable; sin

embargo, de todos modos el algoritmo intenta llegar al punto; al momento en que la distancia al punto deja de disminuir, el algoritmo supone que el vehículo ya llegó al punto.

La primer estrategia asegura que el vehículo pase por todos los puntos, mientras que con la segunda estrategia el vehículo no pasa por los puntos que se encuentran en la zona inalcanzable. Sin embargo, la forma de la curva que sigue el vehículo con la primer estrategia requiere más maniobras; mientras que en la segunda, la curva es más suave. En la figura 4.21, se compara el seguimiento de una trayectoria, usando las dos estrategias. La trayectoria se define por medio de tres puntos (F_1 , F_2 y F_3). El punto F_2 se encuentra en la zona inalcanzable; nótese cómo con la primer estrategia, el vehículo se aleja primero del punto para después alcanzarlo; mientras que con la segunda estrategia, el vehículo se aproxima lo más que puede al segundo punto, y al momento en que la distancia al punto deja de disminuir, avanza al tercer punto.

Las diferentes curvas que se generan por medio de estas dos estrategias, son completamente distintas, aún y cuando se tienen los mismos puntos.

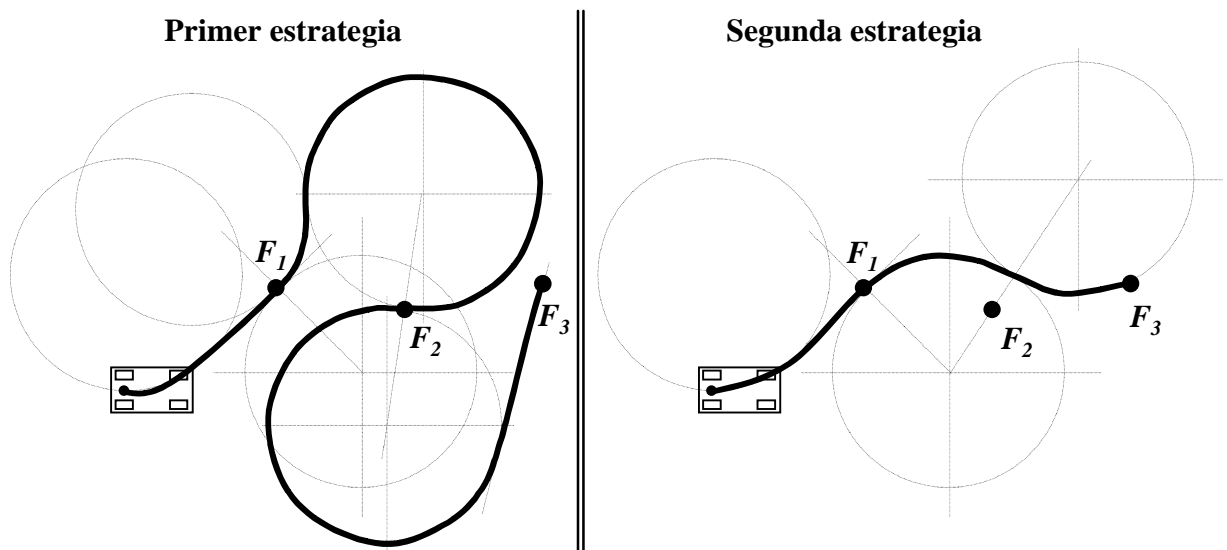


Figura 4.21: Comparación del seguimiento de una trayectoria usando las dos estrategias. Primer estrategia: El vehículo pasa exactamente por los puntos, sin importar la curva que se genera. Segunda estrategia: La curva es más suave, aunque no pasa exactamente por los puntos

En la implementación se utilizó la segunda estrategia, ya que, como se observa en la figura 4.21, las curvas son más suaves; además, es más sencilla de implementar y requiere menos cálculos numéricos.

Capítulo 5

Implementación y experimentos

Este capítulo describe la implementación del lazo de control propuesto descrito en los capítulos 2, 3 y 4. La implementación abarca desde la caracterización mecánica y eléctrica del vehículo, la construcción de los módulos que componen la arquitectura para el control, hasta la programación de los algoritmos de control.

Este capítulo se divide en tres secciones, donde sólo se presenta la implementación y los resultados obtenidos en los experimentos.

5.1 Modelo del sistema

El vehículo utilizado, se adquirió ya contruido, en su versión original el vehículo es de control remoto. Se escogió este vehículo, por su construcción mecánica, la cuál, es muy parecida a un carro común. En la figura 5.1 se muestra una fotografía del vehículo utilizado.



Figura 5.1: Fotografía del vehículo que se utilizó para hacer las pruebas

El mecanismo de la dirección, cumple con la geometría de Ackerman, como se vió en el Capítulo 2. El tamaño del vehículo, es suficiente para colocar todos los instrumentos electrónicos para el control; las medidas de este vehículo se muestran en la Figura 5.2.

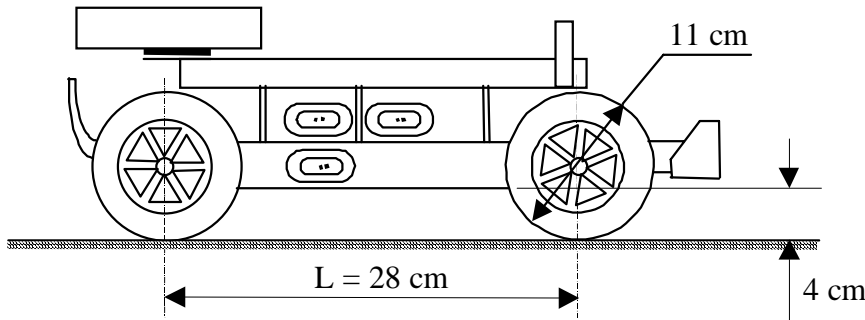


Figura 5.2: Esquema del vehículo con las medidas más importantes. Se muestra la distancia entre ejes de las llantas L , como se definió en el modelo cinemático

Este vehículo cuenta con tres motores de corriente directa: dos para el avance y uno para la dirección. La caracterización de estos motores en cuanto a voltajes de alimentación y consumo de corriente se muestra en la siguiente tabla:

	Motor de dirección	Motores de avance
Consumo de corriente	100 mA máximo	1.5 A máximo - cada uno
Voltaje de alimentación	7.2 V nom.	7.2 V nom.

Tabla 5.1: Caracterización de los motores de corriente directa en el vehículo

El motor de dirección cuenta con un potenciómetro unido a la flecha para sensar la posición. La caracterización del potenciómetro arrojó los siguientes resultados:

Valor nominal:	4.8 k Ω	
Rango de operación:	600 Ω	Todo a la derecha ($\phi = -17^\circ$)
	4.2 k Ω	Todo a la izquierda ($\phi = 17^\circ$)

Tabla 5.2: Características estáticas del sensor de dirección

En la tabla 5.2 se observa que el potenciómetro no trabaja en su rango completo pues el mecanismo de dirección llega a los topes mecánicos antes de que se llegue al tope en el potenciómetro. Los topes mecánicos están colocados de tal manera que las llantas únicamente pueden girar de -17° , tomando ϕ negativo cuando se gira hacia la derecha, hasta 17° , cuando el mecanismo gira hasta el tope del lado izquierdo.

La constante de tiempo, en la función de transferencia del sistema de dirección, no se obtuvo experimentalmente; ya que las pruebas se hicieron con las llantas al aire, obteniéndose una respuesta muy parecida a un integrador ideal. Para realizar las pruebas de la dirección, con el vehículo en movimiento, se requiere de una comunicación inalámbrica que soporte alta velocidad, la cuál no se tenía. La ganancia está dada en unidades de velocidad, ya que al cambiar el voltaje de entrada, se varía la velocidad del movimiento en la flecha del motor, debido al integrador en la función de transferencia. Tampoco se obtuvo el valor de esta ganancia, ya que el motor está unido al mecanismo de dirección y el movimiento se ve limitado por los topes.

La función de transferencia, del sistema de velocidad, tampoco se pudo obtener experimentalmente, dadas las limitaciones en la comunicación, para observar la respuesta, con el vehículo en el piso. Las pruebas realizadas fueron con el vehículo al aire, donde la respuesta es casi inmediata; obteniendo una constante de tiempo muy pequeña, en el orden de milisegundos.

5.2 Arquitectura para el control

La arquitectura para el control incluye cada uno de los elementos físicos encargados de controlar el vehículo. En esta sección se explicará la implementación de la arquitectura propuesta en el Capítulo 3.

5.2.1 Tarjeta para el control de dirección

Como se explicó en el Capítulo 3, cada tarjeta cuenta con una etapa de potencia y una etapa de acondicionamiento. En cuanto a la etapa de potencia para la dirección, se cuenta con un puente H; sin embargo, es necesario cambiar los interruptores de la figura 3.2 por algún elemento electrónico que funcione como interruptor, pero cuya activación sea eléctrica; en este caso, tenemos la posibilidad de utilizar transistores trabajando como interruptores o relevadores. El controlador de la dirección, requiere cambiar constantemente la dirección de giro del motor; por lo tanto, se requiere un elemento de conmutación rápida; por esta razón, además del bajo requerimiento de corriente, se utilizaron transistores, en el puente H para el control de dirección.

La lógica para la conmutación en el puente H, debe incluir señales de control para seleccionar el sentido de giro. El diseño de la tarjeta para el control de dirección controla el motor con tres señales. Una señal es la de “PWM”, la cuál, dependiendo del ciclo de trabajo, indica la velocidad del giro del motor. Al activarse la segunda señal, indica que el giro del motor es a favor de las manecillas del reloj, lo cuál provoca un cambio en ϕ positivo, para futuras referencias, a esta señal se le llamó “IZQ”. La tercera señal que requiere el circuito, se le llamó “DER” y, al activarse, indica un giro en contra de las manecillas del reloj del motor, que provoca un cambio en ϕ negativo. Si las señales “IZQ” y “DER” se activan simultáneamente, el motor no se mueve; así mismo, si ninguna de las dos se activa, tampoco se mueve el motor. Estas dos señales de control sirven para conmutar los transistores.

En cuanto a la etapa de acondicionamiento del sensor de dirección, se conectó un extremo del potenciómetro a +5V y el otro a tierra, obteniendo en el devanado central un barrido de 0 a 5 volts. Para digitalizar esta señal, se utilizó un convertidor análogo-digital de aproximaciones sucesivas con un rango en la entrada de 0 a 5V y una resolución de 8 bits. Dado a que el potenciómetro no trabaja en todo el rango completo, como se mostró en la tabla 5.2, el rango a la salida del convertidor va desde 32d (20h) para una lectura cuando la dirección está en el extremo izquierdo, hasta 224d (E0h) cuando se gira al extremo derecho. Además, el convertidor entrega un 128d (80h) cuando las llantas apuntan en línea recta. Como se observa en la tabla 5.3.

El diagrama eléctrico del circuito implementado para el control de dirección se incluye en el Anexo A.

Posición	Valor resistencia	Voltaje salida	Lectura digitalizada
17 °	600 Ω	0.625 V	32d (20h)
0°	2.8 kΩ	2.5	128d (80h)
-17 °	4.2 kΩ	4.375 V	224d (E0h)

Tabla 5.3: Rango de operación del convertidor de 8 bits para lectura de la dirección

5.2.2 Tarjeta para el control de velocidad

En cuanto a la tarjeta para el control de avance y retroceso del vehículo se implementa también utilizando un puente H. Sin embargo, dado que la inversión en el giro del motor es muy ocasional y los requerimientos de corriente son altos, se utilizaron relevadores. De manera similar, el control cuenta con una señal de PWM y dos señales para la conmutación de los relevadores del puente H. En este caso, las señales de conmutación son “D” y “R”, para seleccionar el movimiento hacia adelante o de reversa.

El sensor de velocidad se compone de un codificador óptico incremental que gira a la misma razón que la llanta trasera izquierda. Conforme gira el motor se generan pulsos cada cierta distancia; si se cuentan el número de pulsos cada cierto tiempo, se obtiene la velocidad. Si se cuentan los pulsos directamente, se obtiene la distancia lineal que avanzó la llanta izquierda.

El codificador óptico utilizado genera treinta pulsos por cada vuelta. Se puede obtener la resolución de este sistema, ya que se conoce el radio de las llantas; por lo tanto se puede calcular el perímetro. El perímetro de la llanta, es la distancia lineal que recorre el carro en cada vuelta; si esta distancia se divide entre los treinta pulsos del codificador, se obtiene la distancia que representa cada pulso. Sustituyendo estos valores y haciendo pruebas en campo, se obtuvo que cada 10 pasos en el codificador, el vehículo avanza 11.1 cm., lo cuál representa una resolución de 1.11 cm.

Se conectó la señal del codificador a una entrada de interrupción del microcontrolador;

de esta forma, cada que se genera un pulso se ejecuta una subrutina que acumula el número de pulsos. Para medir la velocidad se genera una interrupción a intervalos regulares de tiempo, la cantidad de pulsos que se han acumulado durante ese intervalo de tiempo representa la velocidad. De esta forma, la velocidad es directamente el número de pulsos (distancia), por unidad de tiempo.

5.2.3 Unidad de procesamiento

La unidad de procesamiento central se encarga de integrar el control de todos los dispositivos de entrada y salida; además, se encarga del procesamiento matemático para implementar el lazo de control para el seguimiento de la trayectoria. Por ello, se requiere un dispositivo con alta capacidad de procesamiento de señales y con facilidad para la programación de los algoritmos de control. Además requiere de la capacidad de transmitir los datos serialmente a una computadora remota. De acuerdo a como se propuso en la arquitectura de control, la unidad de procesamiento central va instalada en el vehículo; por lo tanto, debe tratarse de un dispositivo ligero, y con bajo consumo de potencia. Por todas estas razones el dispositivo idóneo para esta función es un microcontrolador.

El microcontrolador utilizado cuenta con dos contadores, los cuáles se utilizan para generar las señales en PWM. Cuando el contador se configura en modo PWM, trabaja como un contador hacia arriba y hacia abajo, pues va contando hacia arriba y al llegar al valor máximo comienza a decrementar, la frecuencia se puede configurar con un registro que divide la frecuencia del reloj principal entre 8, 64, 256 y 1024, para obtener frecuencias más lentas al reloj principal. Se añade también un registro de comparación al valor del registro del contador. En la figura 5.3 se puede observar como trabaja en este modo de operación. Al pasar el contador a un valor mayor al registro de comparación, se desactiva una señal de salida, y al pasar de regreso abajo de este valor, la señal de salida se activa nuevamente. De esta forma se genera una salida con un ciclo de trabajo con el porcentaje que representa, el valor del registro de comparación, con el valor máximo en el contador.

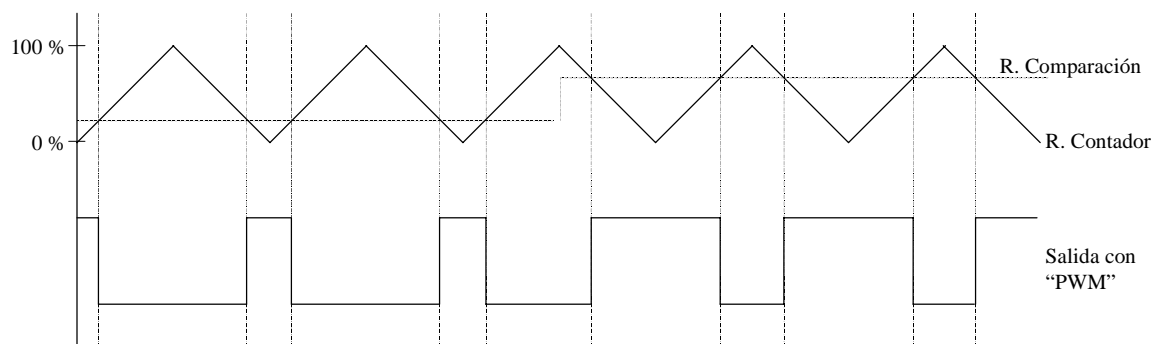


Figura 5.3: Ejemplo de señal PWM utilizando un contador y un registro de comparación

En la implementación se utilizó el microcontrolador AT90S8515 de la familia AVR RISC de Atmel. El microcontrolador cuenta con una arquitectura RISC con 120 instrucciones, 8k bytes de memoria Flash de programa, 512 bytes memoria SRAM interna,

32 x 8 registros de propósito general, 32 líneas programables de Entrada-Salida, puerto serial UART, el tiempo de ciclo por instrucción: $25\mu\text{s}$ @ 4 MHz, un reloj/contador de 8 bits con divisor de frecuencia independiente, un reloj/contador de 16 bits con divisor de frecuencia independiente y con diferentes modos de operación, PWM dual y dos interrupciones externas.

Las características más importantes que lo hacen adecuado para esta aplicación son: la posibilidad de configurar los contadores para generar automáticamente dos salidas de PWM (dos registros de comparación); su alta velocidad de procesamiento de instrucciones y la capacidad de memoria de programa suficiente para introducir todos los algoritmos de control.

En la implementación de un lazo de control con el nivel de complejidad como el propuesto, es muy importante organizar el programa en el microcontrolador de tal forma que se pueda ir agregando nuevo código, sin necesidad de modificar el ya existente. En este proyecto se utilizó un solo microcontrolador para programar todas las tareas, desde la lectura de los sensores, la generación de las señales PWM para manipular los actuadores, hasta la implementación de los algoritmos de todos los controladores en el lazo de control.

Para obtener una organización ordenada, se requiere clasificar las subrutinas de tal forma que se puedan modificar bloques funcionales sin necesidad de modificar otros bloques. Por ejemplo, modificar el algoritmo de control a un punto, sin la necesidad de modificar los demás algoritmos de control; o bien, cambiar el algoritmo del controlador de dirección, sin modificar las rutinas de lectura del sensor y de movimiento del motor. Esta modularidad se logra clasificando las subrutinas; para identificar los módulos necesarios, se toma el diagrama, a bloques, del lazo de control y se programan las subrutinas para cada uno de los bloques.

En la figura 5.4 se muestra un diagrama de flujo del programa principal, que incluye cada una de las subrutinas que se implementaron para el seguimiento de la trayectoria. La lectura de los sensores se produce por interrupciones; por lo tanto, las subrutinas de lectura de posición y velocidad pueden activarse en cualquier momento, dentro del programa principal, en cuanto se genere la interrupción; por lo que las variables de posición y velocidad están disponibles en todo momento. La subrutina de estimación de parámetros (x, y, θ, ϕ) también se ejecuta por interrupción, cada vez que se avanza una distancia fija. La recepción de datos en la comunicación serial, de igual forma, se maneja por interrupciones; cada vez que se tiene un dato listo en el puerto serial, se genera una interrupción que llama una subrutina que procesa el dato recibido.

5.2.5 Comunicación serial

Se diseñó un protocolo de comunicación entre la computadora con el vehículo. La arquitectura de comunicación es “Half-Duplex”, ya que el módem que se utilizó en la transmisión inalámbrica tenía esta limitante. La velocidad de transmisión es de 1200 bps. Para sincronizar la comunicación y hacer un buen uso de la línea, se implementó un protocolo de tipo maestro-esclavo. La computadora es el maestro; cada vez que requiere los datos del vehículo, envía un comando de petición de datos. El vehículo es el esclavo; no

envía datos hasta que no se los pidan. Todos los comandos que envía el maestro siguen un mismo formato. Primero se envía un “byte” de comando, y después otro “byte” con el valor. Además se agregaron otros dos “bytes” como banderas de inicio y fin, sólo para darle sincronía a la comunicación. Con esto nos da la flexibilidad de tener hasta 254 comandos diferentes (256 menos 2 para no repetir banderas de inicio y fin). El vehículo envía sólo los datos más importantes para el usuario, los cuáles corresponden a la estimación de (x,y,θ) .

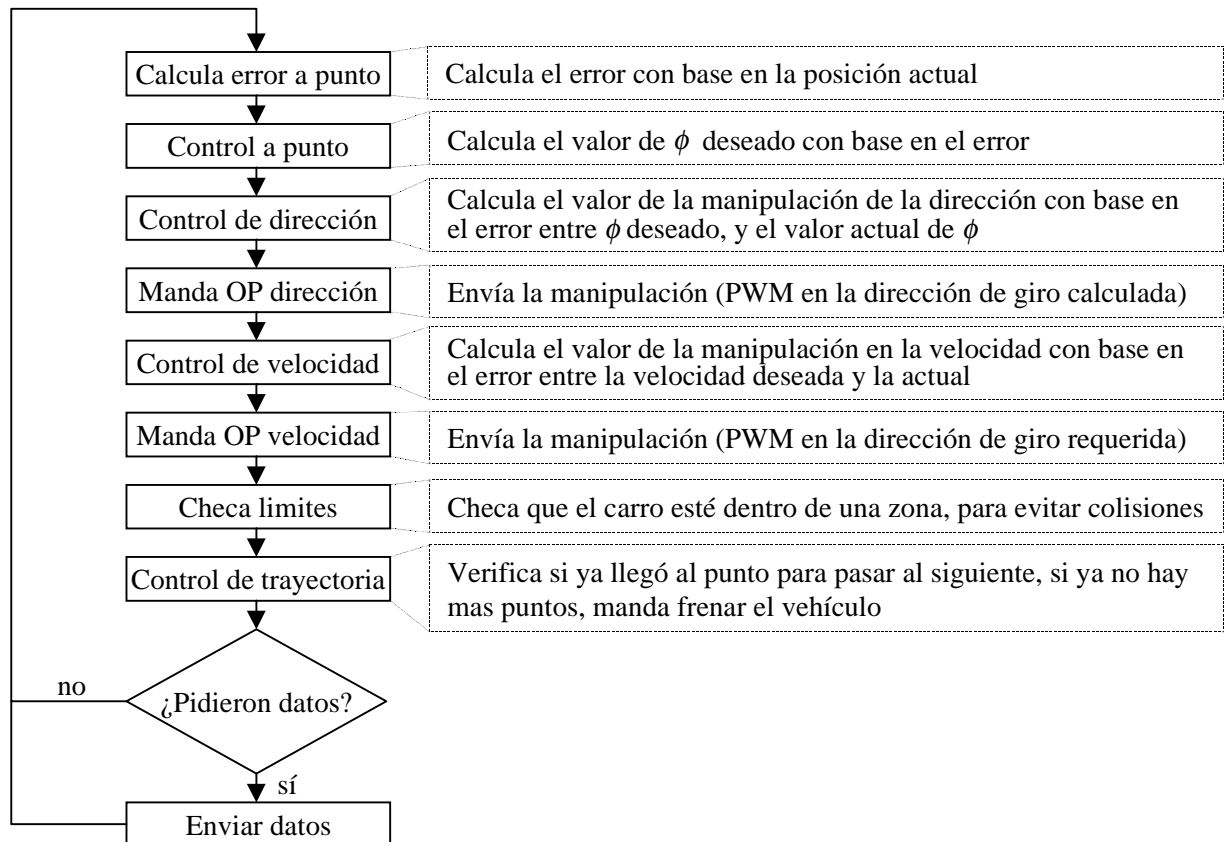


Figura 5.4: Diagrama de flujo del programa principal implementado en el microcontrolador

5.3 Controladores

Para implementar el lazo de control completo, se comenzó desde el nivel inferior relacionado directamente con señales a los dispositivos y se fue ascendiendo en jerarquía hasta llegar al control de la trayectoria. A continuación se muestra la implementación de cada uno de los controladores.

5.3.1 Lazo de control de dirección

Al implementar el cálculo de la manipulación, en el microcontrolador, la ecuación del cálculo de la manipulación sufre un pequeño cambio con fines de facilitar su programación;

la ecuación resultante es:

$$m_n = K_c [\varepsilon_n + K_d (\varepsilon_n - \varepsilon_{n-1})] + m_{ss} \quad (5.1)$$

donde K_d incluye a τ_d y T en una sola constante, para evitar implementar la división entre un valor decimal. De esta forma, todas las variables son enteras y se implementan en variables de 8 *bits*. La manipulación resultante va de 0 a 255, la cuál es directamente el valor del registro de comparación del PWM (0% a 100% respectivamente), el signo del error actual, se utiliza para indicar la dirección de giro del motor. La estrategia de ganancia preprogramada, se implementó de tal forma que se separa la zona de operación en seis diferentes zonas y además, verifica si el movimiento es hacia la izquierda o hacia la derecha, para ajustar la ganancia del controlador, según la zona y dirección de giro.

La subrutina “control de dirección” es la encargada de realizar el cálculo de la manipulación y la subrutina “manda OP dirección” se encarga de poner esta manipulación en el registro de comparación del PWM, y de poner la dirección de giro del motor (señales “IZQ” y “DER” en la tarjeta del puente H). El control se dividió en dos subrutinas, una que calcula la manipulación basada en una ecuación de control de tipo PD y otra que envía la manipulación. Con esta división de funciones el programa se vuelve más modular, ya que si se desea cambiar de controlador, la única subrutina que hay que cambiar, es la que calcula la manipulación, mientras que la que envía la manipulación al motor, no cambia.

La sintonía del controlador de dirección se hizo a prueba y error; sin embargo, el desempeño es satisfactorio, como se observó en el Capítulo 4.

5.3.2 Lazo de control de velocidad

En la implementación, el cálculo de la manipulación en el microcontrolador se cambió por la siguiente ecuación:

$$m_n = m_{n-1} + K_c (\varepsilon_n - \varepsilon_{n-1}) + RST \left(\frac{\varepsilon_n + \varepsilon_{n-1}}{2} \right) \quad (5.2)$$

donde RST es el inverso de la ganancia integral multiplicado por el tiempo de muestreo, multiplicado, a su vez, por la ganancia proporcional. Este cambio de variables se hizo para evitar operaciones de punto flotante. El tiempo de muestreo es de 327.68 milisegundos; sin embargo, para facilitar los cálculos en el microcontrolador, solo se manejan números enteros, por lo que la variable RST ya incluye la división del tiempo de muestreo entre la ganancia integral y la multiplicación por la ganancia proporcional; además, RST sólo puede tomar valores enteros. Finalmente para reducir el efecto integral, porque en la implementación sintonizando con: $RST=1$ aún era grande el efecto y se volvía inestable, se dividió todo el término integral entre cuatro, se escogió este número a prueba y error, además, para implementarlo sólo hay que hacer dos corrimientos hacia la derecha en los *bits* de la variable, quedando finalmente la siguiente ecuación:

$$m_n = m_{n-1} + K_c (\varepsilon_n - \varepsilon_{n-1}) + \frac{RST \left(\frac{\varepsilon_n + \varepsilon_{n-1}}{2} \right)}{4} \quad (5.3)$$

La división entre 4 se realiza hasta después de la multiplicación por RST en el microcontrolador, para trabajar sólo con números enteros; la implementación se hizo con variables de 8 bits. La manipulación resultante va de 0 a 255, y es directamente el valor del registro de comparación del PWM (0% a 100% respectivamente). La subrutina “control de velocidad” es la encargada de realizar el cálculo de la manipulación, y la subrutina “manda OP velocidad” se encarga de poner esta manipulación en el registro de comparación del PWM, y de poner la dirección de giro del motor (señales “D” y R” en la tarjeta del puente H), dependiendo del valor que haya dado el usuario en “vel_dir”; de esta forma, se controla el avance hacia delante o de reversa. Separando el control en dos subrutinas, una que calcula la manipulación con base en una ecuación de control de tipo PI, y otra que envía la manipulación, hace que el programa sea modular.

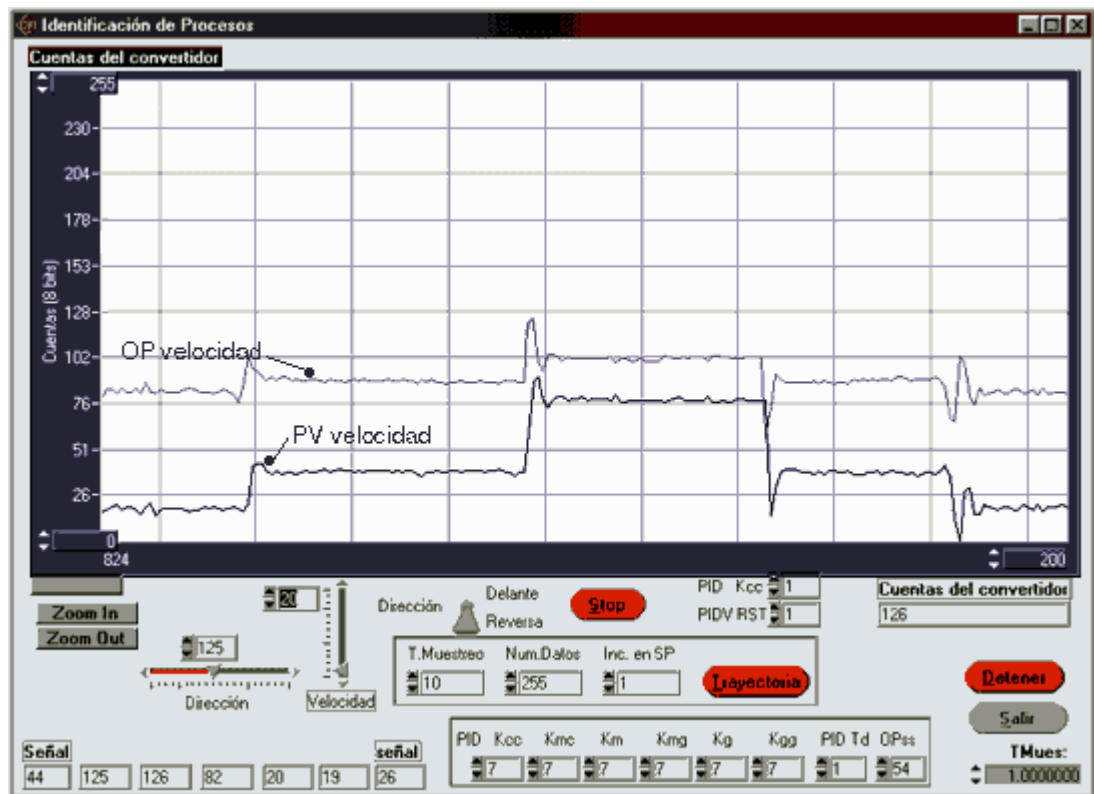


Figura 5.5: Gráfica de desempeño del controlador de velocidad. La gráfica se obtuvo con el vehículo estacionado y con las llantas al aire.

La velocidad se define al principio de la trayectoria y se mantiene fija durante todo el recorrido. Las variaciones que haya en la velocidad, se deben a las perturbaciones que se presentan al momento de estar en el piso ejecutando la trayectoria. Las pruebas que se realizaron, para verificar el desempeño, se hicieron con el vehículo estacionado y con las

llantas al aire para que giraran libremente. El realizar las pruebas con el vehículo estacionado permitió conectar un cable directo a la computadora, para la transmisión serial, y así aumentar la velocidad de transmisión. No se pudo medir el desempeño en el piso, ya que la velocidad de transmisión del módem es muy lenta, y la gráfica que se obtendría sería muy pobre en contenido.

La gráfica de desempeño del controlador de velocidad se muestra en la figura 5.5. Se observa que hay un sobretiro aproximadamente de 40%, sin embargo, se espera que al ponerlo en el piso, este sobretiro desaparezca, debido a que la ganancia disminuye por la inercia que tiene que vencer al acelerar y la inercia que trae a favor al reducir la velocidad.

5.3.3 Estimación de parámetros (x, y, θ)

La estimación de los parámetros se realiza calculando el cambio que hay en cada una de las variables a estimar.

En la programación del microcontrolador, se utilizó el Método 2. Como se mencionó en el Capítulo 4, este método aproxima la curva de desplazamiento, a una línea recta con dirección $\theta + \Delta\theta$. Se seleccionó este método, porque la programación es más sencilla, reduciendo así, el tiempo que dedica el microcontrolador a realizar los cálculos. El muestreo se fijó, para no depender del tiempo, sino de la distancia, de esta forma la distancia d , siempre es la misma, y las ecuaciones se pueden reducir aún más, escalando todo por d , así las variables “Distx” y “Disty” representan incrementos de 11.1 centímetros. Las ecuaciones para la estimación de los parámetros quedan finalmente de la siguiente forma:

$$\Delta\theta = \frac{1}{R} = \frac{\tan\phi}{L} \quad (5.4)$$

$$\Delta x = \cos(\theta + \Delta\theta) \quad (5.5)$$

$$\Delta y = \sin(\theta + \Delta\theta) \quad (5.6)$$

Las funciones trigonométricas seno y coseno, se implementaron utilizando una tabla. La tabla relaciona el valor de θ , con un valor de 8 *bits*, que representa los decimales de una variable de 16 bits; por ello, el máximo valor que puede tomar es 0.99999999; lo cuál concuerda con las funciones seno y coseno, que sólo pueden tomar valores entre -1 y 1 . Se implementó un algoritmo para calcular el seno y el coseno con una misma tabla de datos, la cuál sólo representa los valores de θ de 0° a 90° , en el Anexo D se explica este algoritmo.

En cuanto al cálculo de la ecuación 5.4, se formó también una tabla que relaciona directamente ϕ , con $\Delta\theta$. El cálculo de esta tabla se hizo en forma experimental, ya que la relación de la lectura del sensor de dirección no es lineal respecto a la inclinación real en el ángulo ϕ ; además la distancia L puede cambiar un poco y la distancia d , que se está midiendo no es directamente la distancia del centro del eje trasero, sino la distancia que recorre la llanta trasera izquierda detectada por el codificador óptico. Otro factor que también afecta el cálculo de esta relación teóricamente contra hacerlo de manera

experimental, es la histéresis en el mecanismo de la dirección.

Los experimentos que se realizaron para llenar la tabla consistieron de lo siguiente: primero se colocaba el ángulo ϕ en una posición y se avanzaba el vehículo una cierta distancia. Se contaba cuantos “pasos” había detectado el codificador óptico (cada paso equivale a 11.1 centímetros). Después se medía la inclinación final que había alcanzado el vehículo con un transportador y una regla adherida al vehículo. Como el vehículo describe un círculo cuando se tiene una inclinación fija en ϕ , se puede calcular el $\Delta\theta$ en cada paso,

$$\Delta\theta_{\phi_k} = \frac{\theta_{tot}}{Np} \quad (5.7)$$

donde $\Delta\theta_{\phi_k}$ es el $\Delta\theta$ para el ϕ utilizado, θ_{tot} es la inclinación total, medida con el transportador, y Np es el número de pasos detectados por el codificador óptico.

Las pruebas se hicieron tres veces por cada ϕ_k , para calcular un promedio de $\Delta\theta$ para cada ϕ , cubriendo todo el rango de operación de ϕ .

La tabla 5.4 muestra los datos obtenidos experimentalmente.

Phi	Pasos1	Angulo1	DeltaTeta1	Pasos2	Angulo2	DeltaTeta2	Pasos3	Angulo3	DeltaTeta3	Prom
62	46	360	7.826086957	17	125	7.35294118	13	95	7.307692308	7.495573
66	22	138	6.272727273	15	118	7.866666667	15	118	7.866666667	7.335354
70	14	98	7	19	129	6.78947368	23	163	7.086956522	6.95881
74	16	104	6.5	17	118	6.94117647	19	130	6.842105263	6.761094
80	30	182	6.066666667	23	150	6.52173913	23	145	6.304347826	6.297585
86	18	110	6.111111111	17	108	6.35294118	19	120	6.315789474	6.259947
90	13	77	5.923076923	26	150	5.76923077	16	95	5.9375	5.876603
96	22	110	5	21	105	5	23	101	4.391304348	4.797101
100	22	98	4.454545455	21	95	4.52380952	20	93	4.65	4.542785
106	24	84	3.5	74	227	3.06756757	22	80	3.636363636	3.40131
110	29	87	3	33	84	2.54545455	29	86	2.965517241	2.836991
116	24	40	1.666666667	46	48	1.04347826	37	41	1.108108108	1.272751
120	53	40	0.754716981	33	26	0.78787879	39	31	0.794871795	0.779156
126	37	2	0.054054054	37	2	0.05405405	37	2	0.054054054	0.054054
128	31	2	0.064516129	43	3	0.06976744	43	2	0.046511628	0.060265
130	49	7	0.142857143	35	3	0.08571429	43	4	0.093023256	0.107198
140	44	86	1.954545455	48	97	2.02083333	47	76	1.617021277	1.864133
150	46	145	3.152173913	34	114	3.35294118	37	131	3.540540541	3.348552
160	33	157	4.757575758	28	127	4.53571429	35	150	4.285714286	4.526335
170	29	145	5	35	184	5.25714286	24	127	5.291666667	5.182937
180	28	152	5.428571429	32	182	5.6875	31	176	5.677419355	5.59783
190	25	145	5.8	33	184	5.57575758	21	124	5.904761905	5.760173

Tabla 5.4: Pruebas en campo con el vehículo, para obtener la relación entre ϕ y $\Delta\theta$.

La figura 5.6 muestra la relación entre ϕ y $\Delta\theta$; los valores de θ están en una escala de 0 a 255 para denotar 0° a 360° respectivamente, la cuál se codificó utilizando una variable de 16 bits, para darle más resolución y utilizar decimales.

En la figura 5.6 se observa que al medir la distancia en la llanta izquierda trasera afecta los resultados; cuando el vehículo gira hacia la izquierda, la llanta trasera izquierda avanza menos que la de lado derecho. Por lo tanto, una misma distancia de avance en la llanta trasera izquierda, representa un $\Delta\theta$ mayor cuando gira hacia la izquierda, que cuando gira

hacia la derecha.

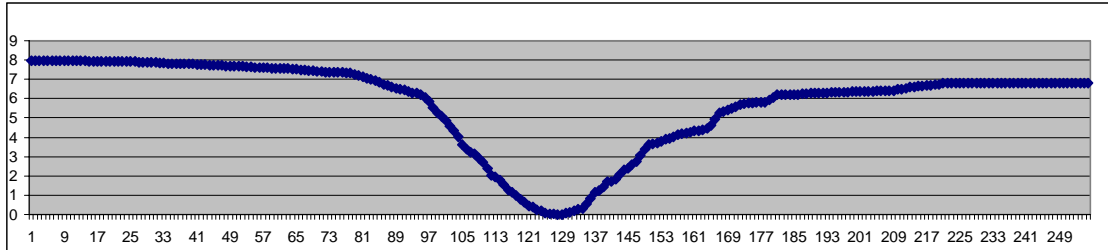


Figura 5.6: Gráfica que relaciona ϕ con $\Delta\theta$, los datos fueron tomados de forma experimental.

Aunque se haya calculado la tabla experimentalmente existen errores, ya que la histéresis en el mecanismo de la dirección permite que haya “juego” en la dirección de las llantas, sin que sea detectado por el sensor; además, las llantas presentan una pequeña deformación que varía con la velocidad, la cuál tampoco se modeló. La deformación se debe a la aceleración generada al avanzar en una curva, la cuál provoca, una fuerza lateral que deforma la llanta, cuya superficie es suave.

5.3.4 Lazo de control a un punto

La implementación del control a un punto, se realizó en el microcontrolador, utilizando tablas para el cálculo de la tangente inversa para calcular el ángulo formado entre la orientación actual del vehículo y la línea que une la posición final con la actual. La división, en el término de la tangente, se implementó utilizando variables de 16 *bits*, de tal modo que el resultado tuviera decimales. La tangente inversa se implementó utilizando cuatro tablas, para proporcionarle mayor exactitud. Si la parte alta del resultado (parte entera) es mayor a 3, se busca la parte alta de la tangente inversa en una tabla. Si la parte alta del resultado de la división (parte entera) vale 0, 1 ó 2, se busca en otra tabla (una tabla para cada caso) que desgloza la primera parte de la tangente inversa de 0 a 1, de 1 a 2, o de 2 a 3 respectivamente. Los resultados de la tabla se obtienen en una escala de 0 a 255 para denotar 0° a 360° , respectivamente, en la variable θ_e (variable de 8 *bits*).

La ecuación de control, se implementó en el microcontrolador con una $k_f=1$, debido a que la dirección se mide con un convertidor que maneja una escala diferente (0 a 255 para denotar -17° a 17°), la ganancia se incluye de forma implícita en este cambio de escalas. El cálculo recorre el error θ_e de tal forma que, cuando vale cero, ϕ_d se codifica como 128d, que es el valor donde las llantas apuntan en línea recta. La implementación consiste en una simple resta; al tratarse de variables de 8 *bits*, un número negativo (si se desborda la resta), equivale a un valor positivo mayor de 128d, ya que no se manejan número negativos.

4.3.5 Lazo de control de trayectoria

El control de la trayectoria que se implementó en el microcontrolador, consta de una

subrutina, la cuál verifica la distancia que falta para llegar al punto; cuando esta distancia es cercana al punto y comienza a crecer, en lugar de disminuir, da por hecho que se llegó a ese punto y se pasa al siguiente. El único cambio en la implementación, es que al momento de calcular la distancia que falta para llegar al punto, la cuál

$$d = \sqrt{(x_f - x)^2 + (y_f - y)^2} \quad (5.8)$$

en realidad se calcula el cuadrado de la distancia, ya que el cálculo de la raíz cuadrada no se implementó en el microcontrolador,

$$d^2 = (x_f - x)(x_f - x) + (y_f - y)(y_f - y) \quad (5.9)$$

de esta forma, el cálculo se facilita, ya que se realiza únicamente con sumas, restas y multiplicaciones.

Esta distancia al cuadrado se almacena en una variable en cada iteración para comparar el nuevo valor con el anterior y de la comparación se toma una decisión, de continuar hacia el mismo punto o pasar al siguiente.

El máximo número que soporta el programa es de 32 puntos, sin embargo, fácilmente se puede aumentar esta capacidad, asignando más memoria al arreglo de datos. Aunque el cálculo de la posición actual del vehículo se implementó usando variables de 16 bits, el usuario define los puntos únicamente con la parte entera de la posición, y se define en la misma escala del cálculo de la posición; por lo tanto un entero representa 11.1 centímetros.

El vector de puntos se alimenta como coordenadas desde la computadora; por lo tanto, se guardan en el espacio de memoria del vector de punto en el eje x , y otro vector de puntos en el eje y . Para el acceso a los puntos se cuenta con un apuntador. Aunque la capacidad de los vectores es de 32 puntos, se pueden definir menos puntos, ya que existe otra variable que indica el número de puntos que dió el usuario.

Se realizaron varios experimentos para probar el funcionamiento del lazo de control de trayectoria completo. Recordando que este controlador funciona como maestro, y tiene como esclavos el control a un punto y, a su vez, el control de dirección. Además este controlador tiene como función frenar el vehículo al llegar al punto final.

El primer experimento que se realizó consistió en llegar a un solo punto; se definió el punto en las coordenadas (27,27) como punto objetivo. Estas coordenadas equivalen a la posición (299.7,299.7) en centímetros (27x11.1=299.7). El vehículo se colocó en la posición (0,0); se definió una cierta velocidad y el vehículo llegó al punto. El máximo error que se observó en la ejecución de la misma prueba diez de veces, fue de 30 centímetros, generando un círculo con radio de 30 centímetros alrededor del punto. Lo cuál representa un 10% de error. Este error se atribuye principalmente a la histéresis, por el “juego” que presenta el vehículo en el mecanismo de dirección, como se explicó en el Capítulo 2.

El segundo experimento consistió en la definición de 2 puntos. Cuyas coordenadas fueron: $q_1=(27,27)$ y $q_2=(0,0)$. Lo que implicaba para el vehículo, ir al punto (299.7,299.7)

centímetros y regresar a la posición inicial (0,0). Aquí se observó mas claramente como el error dependía de la velocidad, ya que al ir el vehículo más rápido, las vueltas las hacía más abiertas y terminaba en una posición, atrás del punto (0,0), y al ir más despacio, las vueltas eran más cerradas y terminaba en una posición adelante del punto (0,0). En la figura 5.7 se muestra gráficamente este comportamiento.

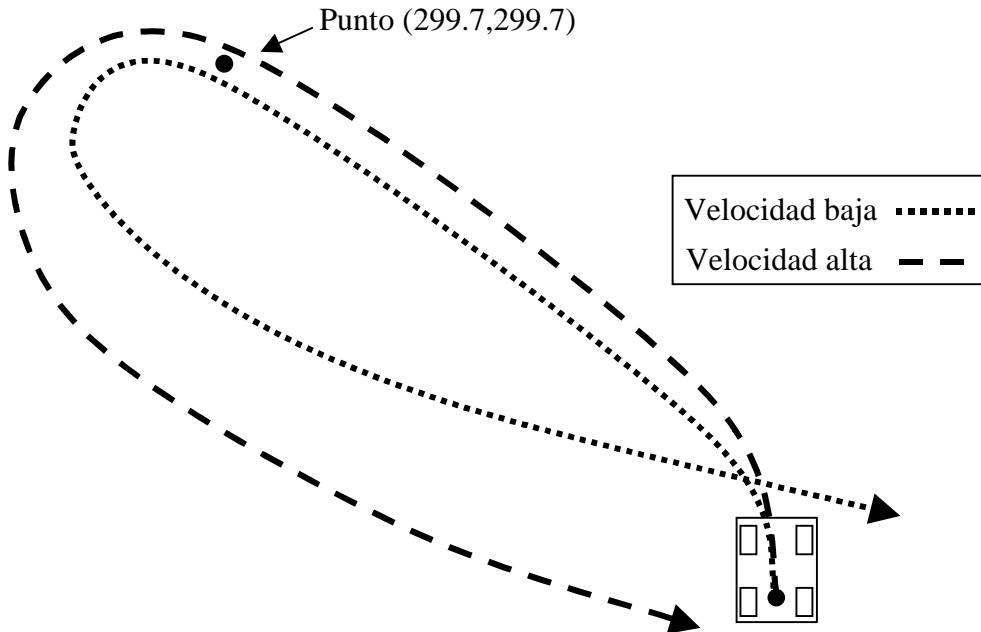


Figura 5.7: Seguimiento de trayectoria definida por dos puntos. Al ir el vehículo a baja velocidad las vueltas son más cerradas, al ir a alta velocidad las vueltas son más abiertas.

Nuevamente, este comportamiento se atribuye al juego que presenta el mecanismo de la dirección, que no puede ser detectado por el potenciómetro en la flecha del motor, afectando directamente la estimación de los parámetros.

La medición del error al recorrer la trayectoria, se realizó observando la trayectoria que recorría el vehículo, y midiendo qué tan alejado pasaba de los puntos. Para evaluar el desempeño, se grabaron en video las pruebas. Cabe mencionar, que debido a la forma en que se estiman los parámetros de posición, el error es acumulativo; conforme más alejados estén los puntos, el error es más grande.

La primer prueba grabada en video consistió en llevar el vehículo a un punto objetivo localizado en la posición (54,0), la posición inicial del vehículo fue de (0,0). Este movimiento representa 6 metros en dirección del eje X.

La figura 5.8 muestra el recorrido. Se colocó una marca de papel en el punto objetivo. Se observa como el vehículo avanza en línea recta hacia el punto. Al final, el centro del eje trasero del vehículo queda separado del punto objetivo una distancia de 25 cm.

El error que se obtiene es de 25 cm, definiendo el error, como el radio del círculo, alrededor del punto objetivo, en donde llega el vehículo.

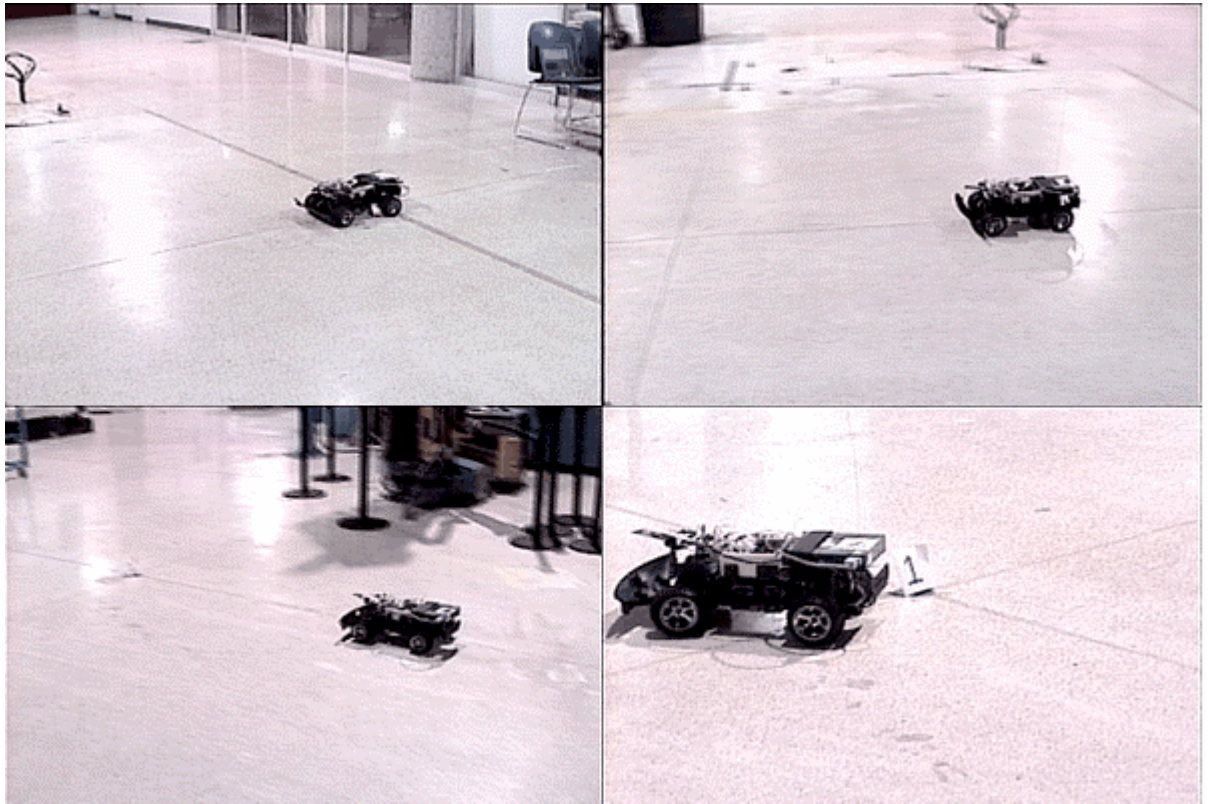


Figura 5.8: Recorrido del vehículo a un punto. Se observa como el vehículo llega al punto, con un error de 25 cm hacia la izquierda

La siguiente prueba consistió en formar una trayectoria de dos puntos. La posición inicial del vehículo es (27,135). El primer punto objetivo se localiza en (81,135), y el segundo en (121,76). La posición inicial, no fue cero para mover el vehículo en dirección negativa, sin que los parámetros de posición fueran negativos. Representando el movimiento en metros y definiendo la posición inicial en (0,0), los dos puntos objetivo se definen en (6,0) y (10.5,-6.5).

La figura 5.9 muestra el recorrido obtenido del vehículo, al ejecutar la trayectoria definida por dos puntos objetivo. El error que se obtiene en el primer punto es mínimo, menor a 5 cm., mientras que en el segundo punto se obtiene un error de 40 cm. Aquí se observa como al aumentar la distancia, el error es más grande, debido a la forma en que se estiman los parámetros de posición.

También se probó el controlador, con una trayectoria de tres puntos. Los primeros dos puntos son los mismos que la prueba anterior, mientras que se agrega un punto más, en la posición (90,54). Representando el movimiento en metros y definiendo la posición inicial en (0,0), los tres puntos objetivo se definen en (6,0), (10.5,-6.5) y (7,-9).

La figura 5.10 muestra el recorrido obtenido del vehículo, al ejecutar la trayectoria definida por tres puntos objetivo. En este caso, el error en el primer punto objetivo, es también pequeño, menor de 10 cm. En el segundo punto objetivo, existe un poco más de error, aproximadamente 35 cm., mientras que en el tercer punto se obtuvo un error aproximadamente de 75 cm.

Cabe mencionar, que el posicionamiento inicial del carrito es muy importante, si desde el principio, el carro avanza desviado, toda la trayectoria se ejecuta con error.

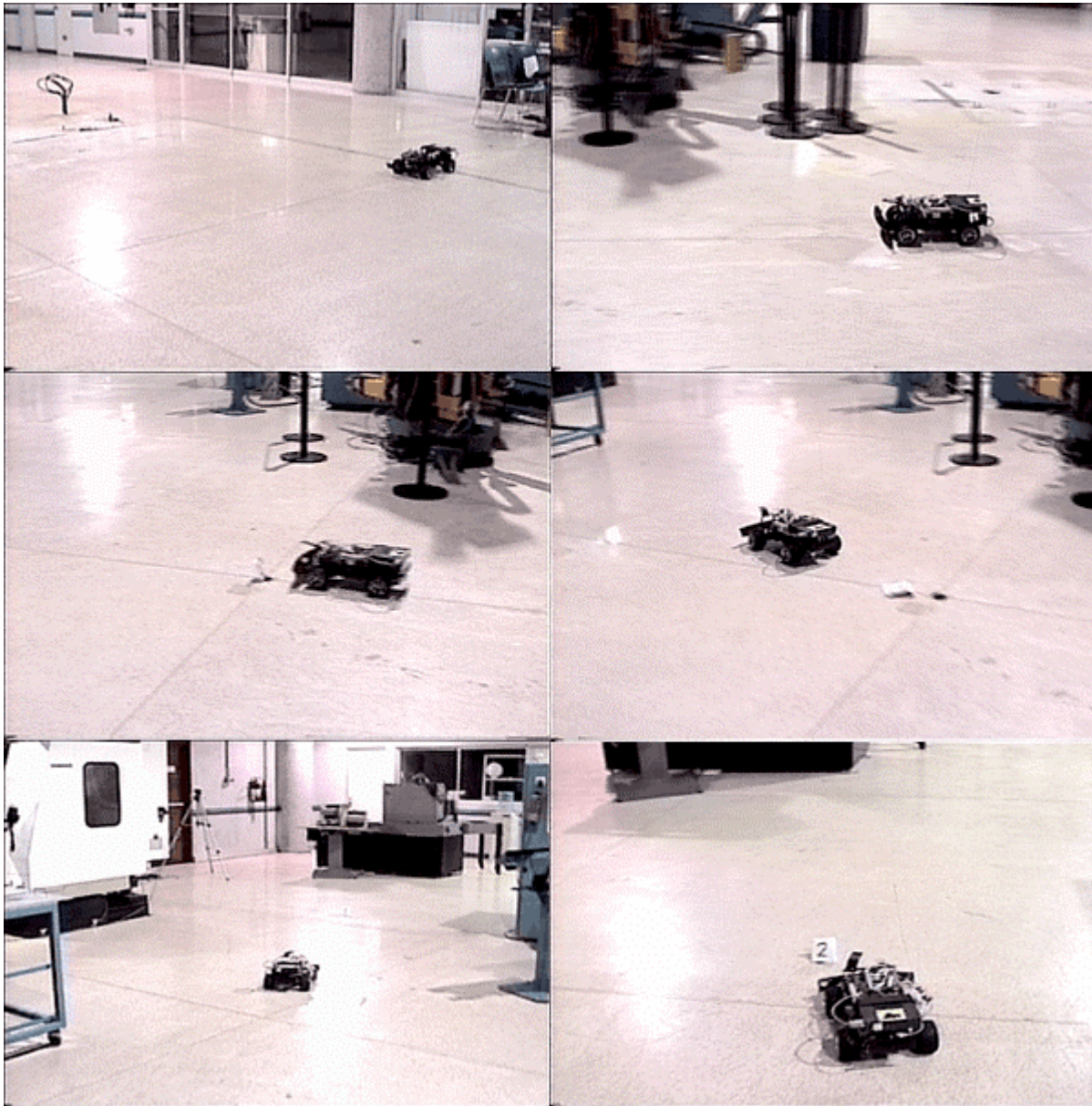


Figura 5.9: Recorrido del vehículo con una trayectoria de dos puntos. Se observa como pasa por el primer punto con un error de 5 cm, mientras que queda a 40 cm del segundo punto

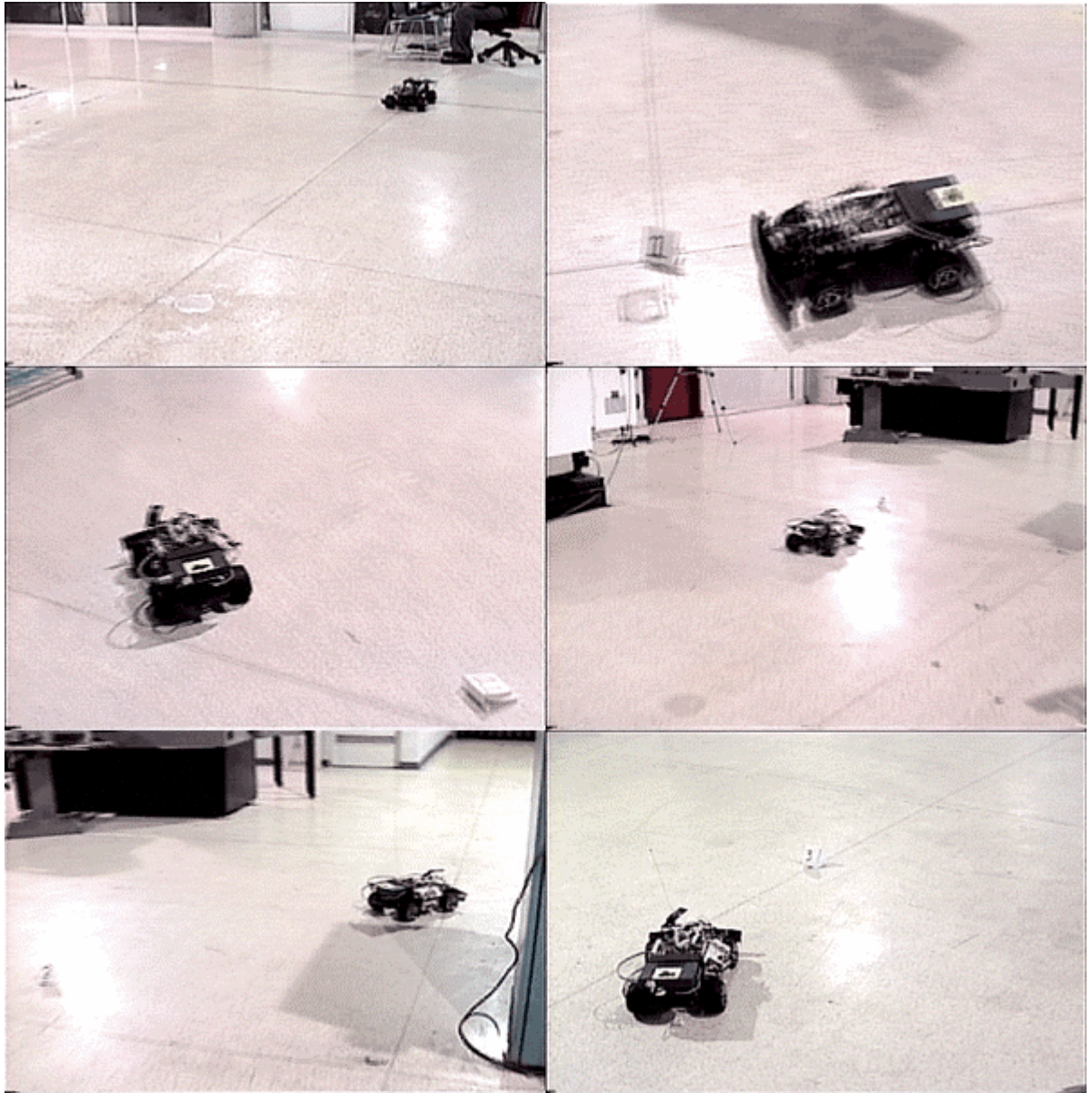


Figura 5.10: Recorrido del vehículo con una trayectoria de tres puntos. Se observa como pasa por el primer punto con un mínimo error, mientras que por el segundo con un error de aproximadamente 35 cm. El tercer punto es el que tiene más error, ya que el vehículo queda a 75 cm del punto

Capítulo 6

Conclusiones

La presente tesis propone el desarrollo de un vehículo que sigue una trayectoria dada, controlando su dirección y su velocidad. El vehículo se clasifica como semiautónomo, ya que el cálculo de la trayectoria la realiza el usuario. El controlador que aquí se desarrolla se encarga de convertir esa trayectoria, en movimientos reales del vehículo.

La cinemática del vehículo es igual a la de un carro común; se tienen dos ejes, en el eje delantero están las llantas que giran para direccionar el vehículo, mientras que la dirección de las llantas en el eje trasero está fija. La trayectoria se define por medio de puntos en el plano, en pares de coordenadas. La parte central del eje trasero del vehículo es el punto que debe de pasar por los puntos para formar la trayectoria. El control se realiza con un controlador en cascada, cuyo controlador maestro se encarga de controlar la posición del vehículo en el plano, verificando que el vehículo pase por los puntos. Los controladores esclavos controlan la dirección de las llantas delanteras y la velocidad lineal del vehículo. El control maestro va cambiando el valor de referencia en el controlador esclavo de dirección, de tal forma que el vehículo se dirija hacia el siguiente punto, al llegar al último punto de la trayectoria frena el vehículo.

La implementación se hizo en un vehículo comercial de control remoto que ya contaba con motores de corriente directa para la dirección y la tracción. El vehículo se instrumentó con los circuitos necesarios para controlar los motores y para leer las señales de los sensores desde un microcontrolador. El control de los motores de corriente directa utilizó puentes H y la señal de control se envía mediante modulación de ancho de pulso (PWM), para regular la potencia entregada al motor. El sensor de dirección es un potenciómetro unido a la flecha del motor que mueve la dirección, mientras que el sensor de avance es un codificador óptico incremental conectado físicamente con la llanta trasera izquierda.

Para el cálculo de la posición del vehículo en el plano, se estima la posición y la inclinación del cuerpo del vehículo. La estimación se realizó utilizando únicamente los sensores de dirección y de avance; por lo cuál el lazo de control de la trayectoria se cierra interiormente al vehículo y no al exterior.

La tarjeta desarrollada para el manejo de potencia del motor de la dirección, funcionó muy bien desde el principio, entregando la potencia adecuada al motor, sin calentar los circuitos. La interfase entre la tarjeta de control con el microcontrolador fue sencilla de programar, permitiendo hacer pruebas de movimientos sencillos en menos de una semana. Los comandos del microcontrolador AT90S8515 de Atmel, permiten efectuar tareas complejas de una forma sencilla y con pocas líneas de código; además, la velocidad de operación es mucho más rápida que los microcontroladores de la familia 8051 convencionales.

El lazo de control de dirección, en un principio presentó problemas, dado a la no-linealidad del actuador; sin embargo, el uso de la estrategia de ganancia preprogramada, permitió que el desempeño mejorara y se obtuviera finalmente un tiempo de respuesta en fracciones de segundo, con un mínimo error.

Al implementar la tarjeta de control de velocidad se tuvieron problemas, debido al consumo de potencia de los motores. El problema se resolvió, utilizando transistores con una mayor capacidad de corriente, y colocando disipadores de calor en dichos transistores.

El ruido eléctrico que se genera al usar los motores de corriente directa para el avance, fue el que más problemas causó, porque afectaba al microcontrolador en la ejecución del programa. El microcontrolador se reiniciaba aleatoriamente, y los valores de las variables se perdían. La solución a este problema, fue aterrizar el chasis en donde se montaron todas las tarjetas; de esta forma se aisló el ruido eléctrico del microcontrolador.

El ruido eléctrico, también afectó la señal del codificador óptico que mide la velocidad y avance, provocando una lectura errónea del sensor. Este problema se trató de solucionar, usando una fuente diferente a la de los motores, sin embargo, el problema persistió y la solución, fue por medio de *software*, donde, se ignoraban los pulsos que fueran muy pequeños (“*glitch*”).

El lazo de control completo propuesto cumple con los requerimientos para el seguimiento de la trayectoria, gracias a su implementación, el vehículo se mueve en el plano pasando por los puntos especificados con un mínimo error. El control aunque es muy sencillo (únicamente se usaron PID's), controla el vehículo a través de la trayectoria.

El controlador de trayectoria, aunque en este desarrollo, solamente envía el punto siguiente al controlador a un punto, se puede desarrollar en trabajos futuros, de tal forma que, manipule diferentes puntos al controlador a un punto, para llevar al vehículo por trayectorias más complicadas, que en lugar de estar definidas por puntos, se definan por curvas *spline*, o por algún otro método; recordando que lo que se busca en la arquitectura de control propuesta, es tener un esquema modular y que sea posible extenderse y utilizarse en diferentes aplicaciones.

El error en las trayectorias, se debe a la histéresis que tenía el vehículo debido al “juego” en el mecanismo de la dirección. Hay que recordar, que el vehículo en su versión original es un juguete, y los mecanismos no están diseñados para trabajar con mucha exactitud.

El esquema de control desarrollado, puede ser utilizado en cualquier vehículo semejante

a un carro común, independientemente de la tarea que vaya a desarrollar. Además, la modularidad de la arquitectura de control, permite que sólo se sustituyan los bloques necesarios para adecuarlo a otro tipo de vehículo.

La arquitectura de control jerárquica, permite descomponer un problema complejo, en subproblemas más sencillos de resolver.

Las limitantes se encuentran en la forma en que se obtiene la posición del vehículo en el plano. La estimación de los parámetros tiene errores que se deben al juego en el mecanismo que mueve la dirección del vehículo. No es posible sensar la posición real de las llantas delanteras, ya que el sensor se encuentra en la flecha del motor y no en las llantas, y el juego en el mecanismo permite una diferencia entre la posición de la flecha, respecto a la posición real de las llantas.

Se propone utilizar algún sensor que detecte la posición real en las llantas delanteras o corregir el problema en el mecanismo de dirección para que no haya juego. Otra mejora que se puede hacer a este proyecto, es cerrar el lazo de control al exterior, ya sea con brújulas electrónicas, GPS o con cámaras de video, de forma que se elimine la parte de estimación de parámetros y que se tenga la posición real del vehículo en todo momento.

El algoritmo de control propuesto no toma en cuenta la posición del punto siguiente al punto al cual se dirige, para suavizar la trayectoria recorrida, sino que sólo se dirige al punto sin tomar en cuenta la inclinación de llegada. Como trabajo futuro se propone agregar la inclinación en cada punto como una variable a controlar, de tal forma que se obtengan trayectorias más suaves.

El estudio de los robots móviles en general, se puede descomponer en tres subproblemas: sensado del espacio, razonamiento del espacio y actuación en el espacio. Esta tesis sólo se enfoca a la parte de actuación. Para hacer un vehículo completamente autónomo, es necesario tener estos tres subproblemas resueltos, por lo tanto, el trabajo presente puede servir para unirse con otros trabajos relacionados al razonamiento del espacio y el sensado, de tal forma que se pueda construir un vehículo completamente autónomo.

Apéndice A

Programa de simulación

Se desarrolló un programa para simular el movimiento del vehículo. Las ecuaciones que rigen el movimiento son las obtenidas del modelo cinemático. Este programa sirvió para probar los algoritmos del controlador de trayectoria y el controlador a un punto. El programa se desarrolló en un paquete de programación llamado LabWindows CVI. Se escogió este paquete por la facilidad de desarrollar las interfases con el usuario y el manejo de gráficos.

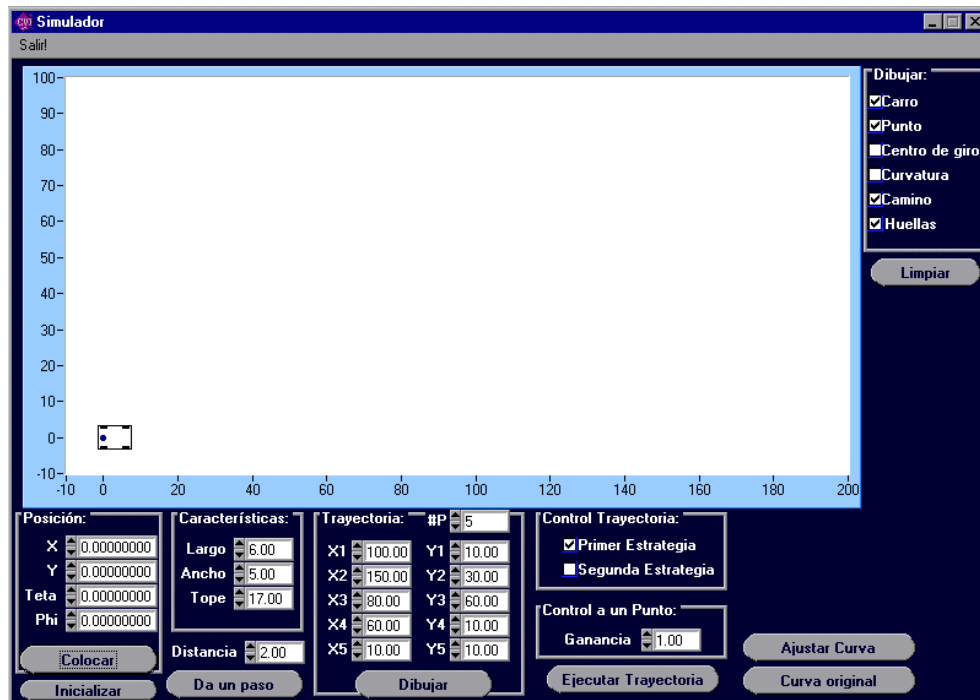


Figura A.1: Pantalla principal del simulador. Presenta el plano donde se mueve el vehículo y todos los controles para hacer las simulaciones

En la figura A.1 se muestra la pantalla principal del simulador. Sobre esta pantalla se

realizan las simulaciones; la mayor parte la ocupa el plano en donde se mueve el vehículo. Los controles que se encuentran en el resto de la pantalla, permiten mover diferentes parámetros del vehículo o del controlador, así como también las opciones de despliegue del vehículo. Todos los controles se separan en diferentes espacios, delimitados por un marco. Por ejemplo, el lugar en donde se especifican las “Características” del vehículo: largo, ancho y la posición del tope mecánico en la dirección.

En el espacio de “Posición”, se tienen los parámetros que definen la posición (“X”, “Y”) e inclinación (“ θ ”) actual del vehículo en el plano, así como también la inclinación de las llantas delanteras (“ ϕ ”).

Dentro del espacio de “Trayectoria”, el simulador permite especificar una trayectoria de hasta cinco puntos. Las coordenadas de los puntos se especifican por su posición en el *eje x* (X1, X2... X5) y en el *eje y* (Y1, Y2... Y5). En la parte superior, se puede especificar el número de puntos que se desean ejecutar (“#P”, máx. 5). Finalmente el botón de “Dibujar” permite trazar la trayectoria definida en el plano, uniendo con líneas rectas desde la posición actual del vehículo, cada uno de los puntos por donde se desea pasar.

En el espacio de “Control de Trayectoria”, se especifica la estrategia para el controlador de trayectoria, como se define en el Capítulo 4.

En el espacio de “Control a un Punto” se especifica la ganancia del controlador a un punto, como se explicó en el Capítulo 4.

A continuación se presentan varios ejemplos de lo que se puede hacer con el simulador.

A.1 Comparar las dos estrategias de control de trayectoria

Paso 1: Especificar la trayectoria y dibujarla. Ver Figura A.2.

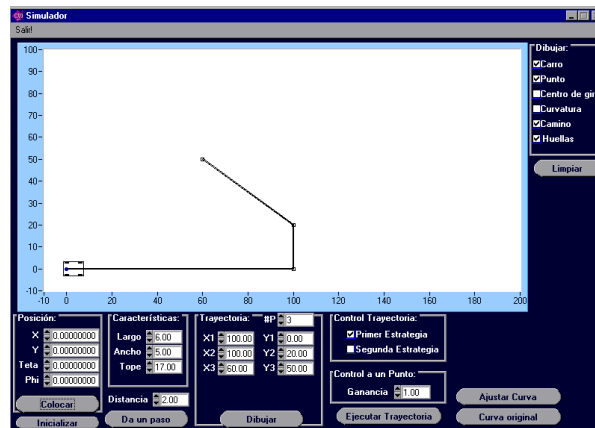


Figura A.2: Dibujar trayectoria en el plano

Paso 2: Seleccionar “Primer Estrategia” y después “Ejecutar Trayectoria”. Después “Inicializar” la posición del vehículo, y seleccionar “Segunda Estrategia” y después “Ejecutar Trayectoria”. Ver Figura A.3.

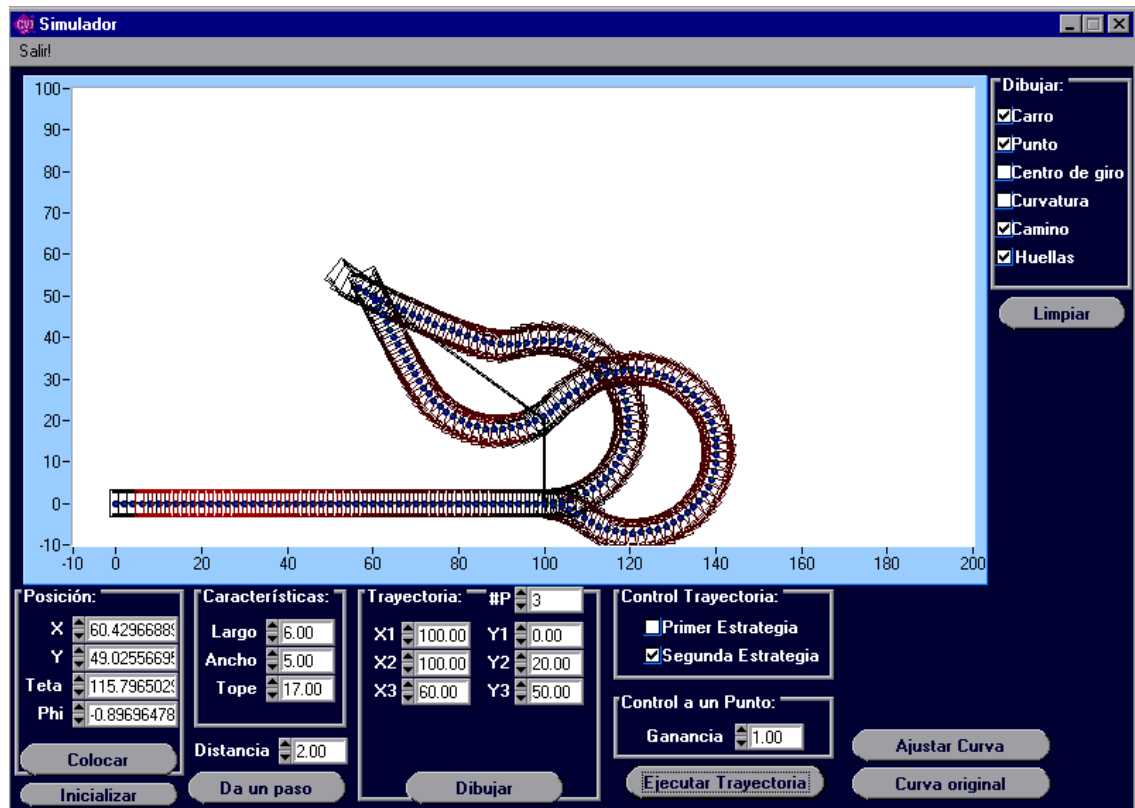


Figura A.3: Ejecutar la trayectoria usando cada una de las dos estrategias

En la figura se observa como en uno de los recorridos (Primer Estrategia), no pasa exactamente por los puntos, mientras que en el segundo recorrido, si pasa por los puntos. La distancia que avanza el vehículo en cada paso se puede cambiar en el control donde dice “Distancia”.

A.2 Cambiar ganancia en el controlador a un punto

Paso 1: Especificar una trayectoria y Ejecutarla con una ganancia de 0.6. Dibujar el centro de giro, y deshabilitar la opción de dibujar Carro. Utilizar la “Primer Estrategia”.

En la figura A.4, se observa como los cambios en el centro de giro, que dependen directamente del cambio en phi, son pequeños, lo cual forma un patrón gradual, ya que la ganancia es pequeña.

Paso 2: Inicializar la posición del carro y seleccionar una ganancia de 3.0, para finalmente “Ejecutar Trayectoria”.

En la figura A.5, se observa como los cambios en el centro de giro, o el cambio en el volante de dirección son más bruscos entre muestreos, y se genera un patrón casi discretizado, entre volante recto y volante torcido hasta el tope mecánico.

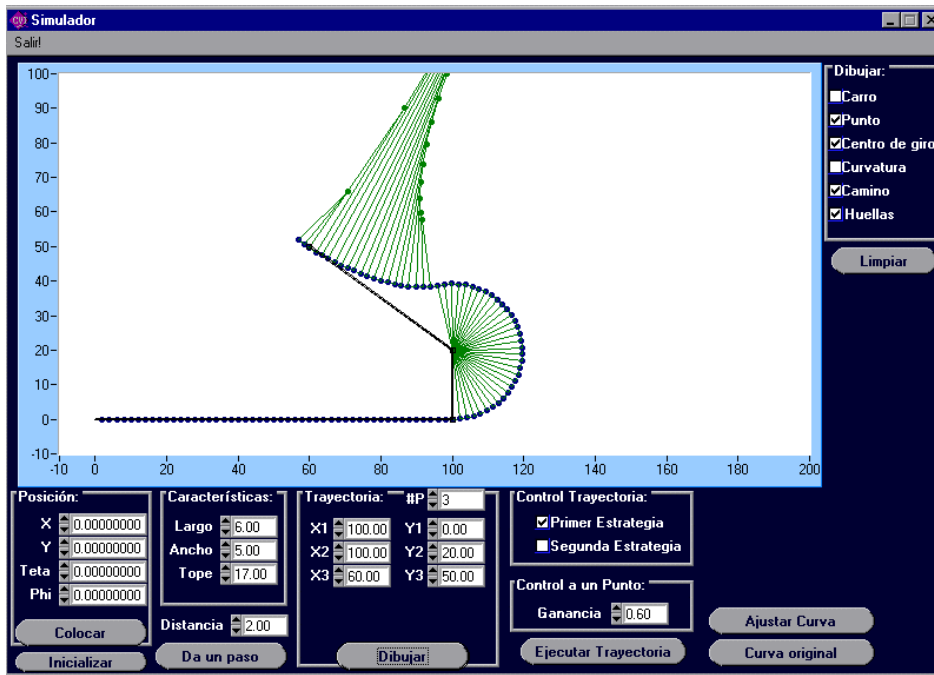


Figura A.4: Ejecución de trayectoria con una ganancia de 0.60 en el controlador

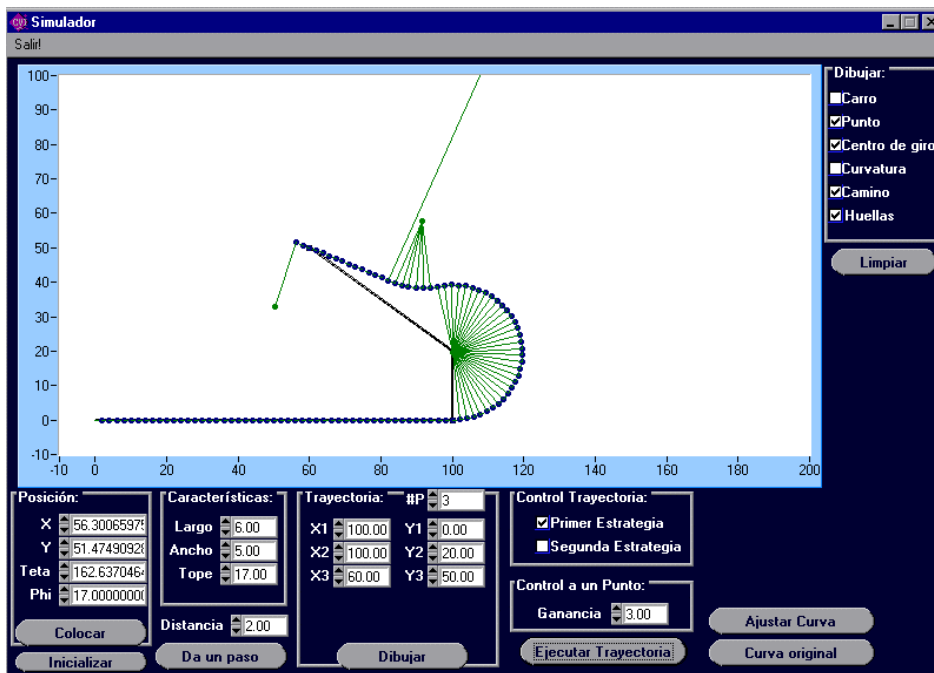


Figura A.5: Ejecución de la trayectoria con una ganancia de 3.0 en el controlador

A.3 Efecto de cambiar la longitud del vehículo

Ejecutar una trayectoria con un vehículo que mida diez unidades de largo. Después ejecutar la misma trayectoria pero con un vehículo que mida únicamente cinco unidades.

En la figura A.6 se muestra el resultado obtenido. El vehículo pequeño puede dar las vueltas más cerradas, que el vehículo de mayor longitud.

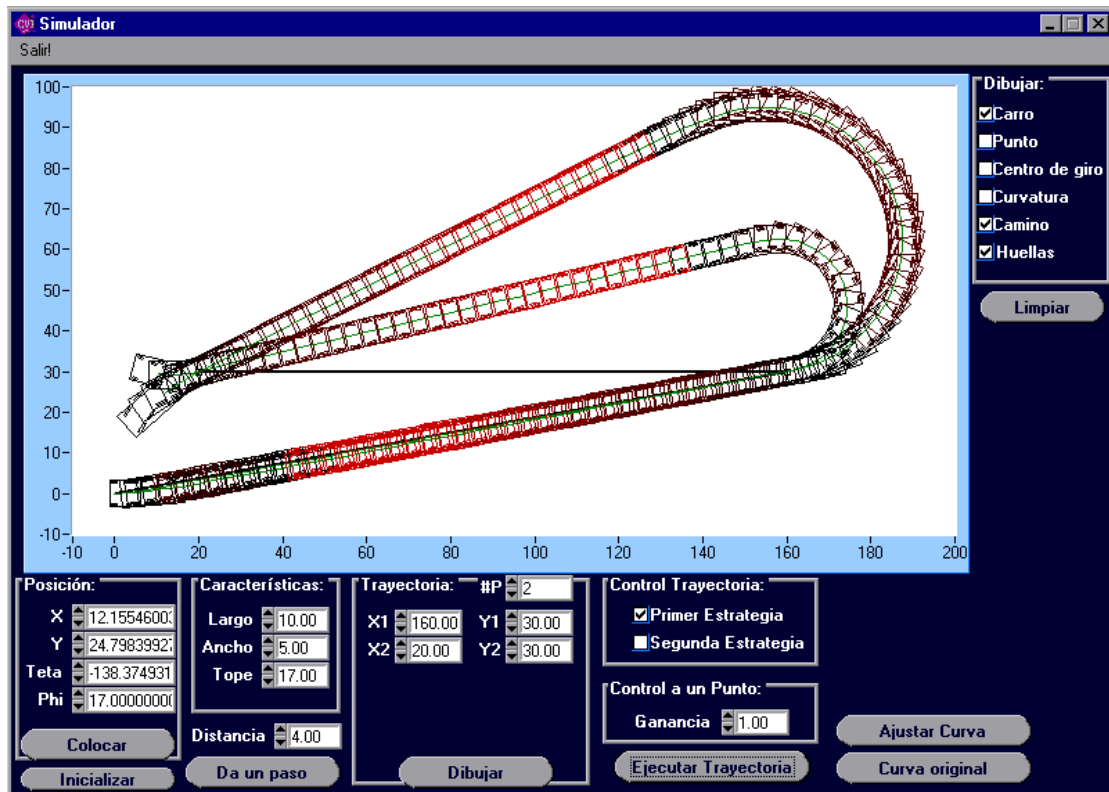


Figura A.6: Se observa el efecto de cambiar la longitud del vehículo. Un vehículo pequeño puede dar vueltas más cerradas que uno más largo. El radio de giro depende de la longitud del vehículo, como se explicó en el Capítulo 2.

Apéndice B

Programa del microcontrolador AVR

El programa en el microcontrolador, contiene todos los módulos que componen el lazo de control. Cada módulo está contenido en una subrutina, de forma que el programa principal, hace una llamada a cada subrutina y se repite indefinidamente. Además, el microcontrolador se encarga de manejar la comunicación serial para recibir y enviar comandos, entre el vehículo y una computadora, de forma remota utilizando un módem. El microcontrolador también manipula los motores a través de la etapa de potencia y recibe la señal de los sensores por medio de las etapas de acondicionamiento, en las tarjetas de control.

Tanto para la comunicación con las tarjetas de control, como para la comunicación serial con la computadora, se utilizan interrupciones. El funcionamiento de las interrupciones, está basado en eventos de tiempo real; cada vez que se tiene un dato listo, se interrumpe al programa principal, para ejecutar una subrutina que atiende la interrupción, de forma que el programa continúa mientras no haya algún evento que atender.

B.1 Manejo de variables en el microcontrolador

El manejo de la información es clave para la modularización, las variables se deben almacenar en espacios de memoria donde todas las subrutinas tengan acceso; por ello, todas las variables son globales. Dado que se trata de un microcontrolador de 8 *bits*, las variables se manejan en espacios de memoria de un tamaño múltiplo de 8 *bits*, dependiendo de la resolución que se requiera de cada variable. Por ejemplo, si solamente se utilizaran 8 *bits* para representar el valor de θ , la resolución máxima alcanzada sería de 1.406° ($360^\circ/2^8$), pero si se utilizan 16 *bits*, se podría alcanzar una resolución de 0.005493° ($360^\circ/2^{16}$). Definiendo como resolución el cambio mínimo en una variable que produce un cambio en el *bit* menos significativo de su representación binaria.

Para facilitar los cálculos en el microcontrolador, se utilizó una escala distinta en las variables que normalmente toman valores de porcentajes. Por ejemplo, las variables manipuladoras toman valores de 0 a 255, para denotar 0 a 100% respectivamente

(obteniendo una resolución de 0.390625%). Cuando se requiere mayor resolución en una variable se aumenta otro *byte* para formar variables de 16 *bits*. Por ejemplo, θ se implementó en una variable de 16 *bits*, donde los 8 *bits* más significativos son los enteros (0 a 100%), y los 8 *bits* menos significativos son los decimales. La representación de los números con decimales se muestra en la siguiente figura:

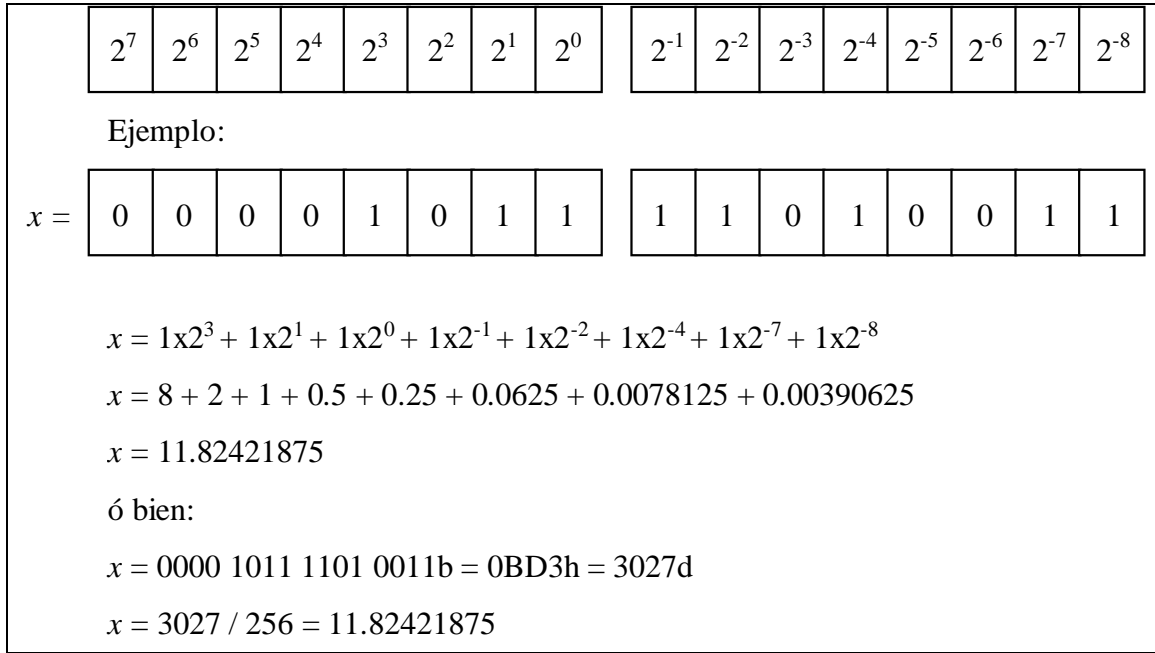


Figura B.1: Representación de números de 16 bits. El byte más significativo representa el valor entero, y el menos significativo los decimales

Las variables de posición (x,y) también se representaron en valores de 16 bits, para dar más resolución, donde 1 representa un macropaso, y cada macropaso es de 10 pasos del codificador óptico, que representan 11.1 cm. Por lo tanto la máxima distancia que puede detectar es de $11.1\text{cm} \times 256 = 2841.6$ cm. Es decir, el área de trabajo máxima de este vehículo es de 28.41 metros de largo, por 28.41 metros de ancho. Dado que las ecuaciones en la estimación de parámetros involucran funciones trigonométricas, los valores de x,y no siempre son números enteros; el avance lineal del vehículo tiene una resolución de 11.1 cm, pero la resolución de las variables (x,y) es de 0.043359 cm ($11.1\text{ cm} / 256$). En la tabla B.1 se enlistan las variables más importantes utilizadas en el programa junto con la resolución de cada una y una descripción de lo que representan.

Variable	Mem.	Valores y resolución	Descripción
var_pvd	8 bits	Representación No-lineal	Es la lectura actual del convertidor análogo digital
var_spd	8 bits	Representación	Es el "Set point" de la dirección, puede tomar valores de 0 a 255 en la misma escala que la

		No-lineal	lectura del convertidor análogo-digital
var_opd	8 bits	0 a 100% resolución: 0.390625%	Manipulación del motor de dirección. Este valor representa directamente el ciclo de trabajo de la señal PWM.
st_dir	8 bits	0 ó 1	La dirección de giro del motor de dirección. Donde un 0 representa movimiento a la izquierda, un 1 representa movimiento a la derecha.
var_pvv	8 bits	0m/s a 8.638m/s resolución: 0.0337 m/s	Es el número de pasos en el codificador óptico cada 327.68 milisegundos. Cada paso equivale a 1.11 cm. La máxima velocidad que puede detectar son 255 pasos / 327.68 milisegundos, que corresponde a 2.8305 m cada 0.32768 seg, o bien, 8.638 m/s
var_spv	8 bits	0 a 8.638 m/s	Es el “Set point” de la velocidad, puede tomar valores de 0 a 255 en la misma escala que var_pvv.
var_opv	8 bits	0 a 100% resolución: 0.390625%	Manipulación del motor de velocidad. Este valor representa directamente el ciclo de trabajo de la señal PWM.
vel_dir	8 bits	0, 1 ó 2	La dirección de avance de velocidad: 0 reversa, 1 delante y 2 alto.
Distx	16 bits	0 a 28.41 m resolución: 0.043359 cm	La posición estimada en el eje x, representado en número con decimales, donde un 1 entero representa 11.1 cm.
Disty	16 bits	0 a 28.41 m resolución: 0.043359 cm	La posición estimada en el eje y, representado en número con decimales, donde un 1 entero representa 11.1 cm.
Teta	16 bits	0 a 360° resolución: 0.005493°	La inclinación del cuerpo del vehículo θ , estimada. Representación en número con decimales, donde un 1 entero representa 1.406°

Tabla B.1: Lista de variables más importantes utilizadas en el programa, el espacio asignado de memoria, los valores posibles junto con la resolución y la descripción

Apéndice C

Diagramas eléctricos

La siguiente figura muestra el diagrama del *Puente H* para el motor de dirección:

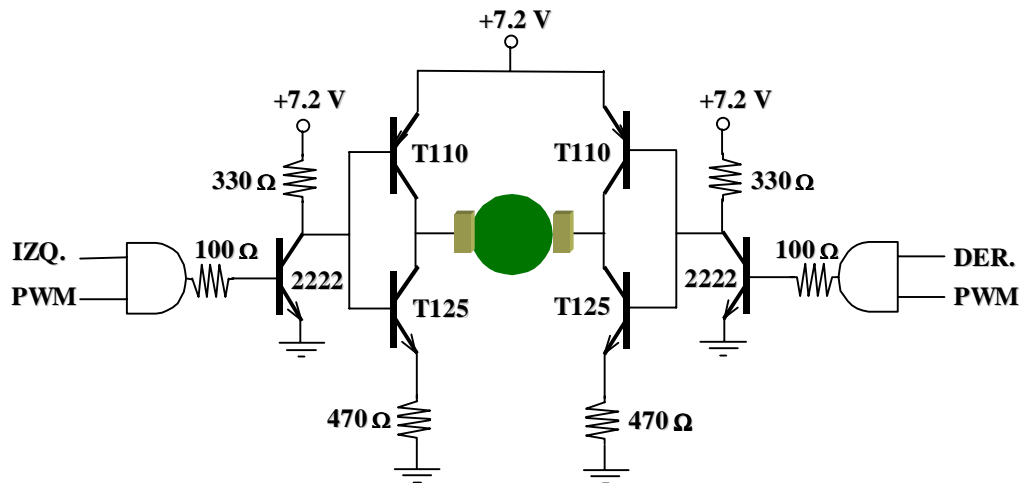


Figura C.1: Diagrama eléctrico del *puente H* para el control de dirección

La siguiente figura muestra el diagrama del convertidor análogo digital para leer la señal del sensor de dirección:

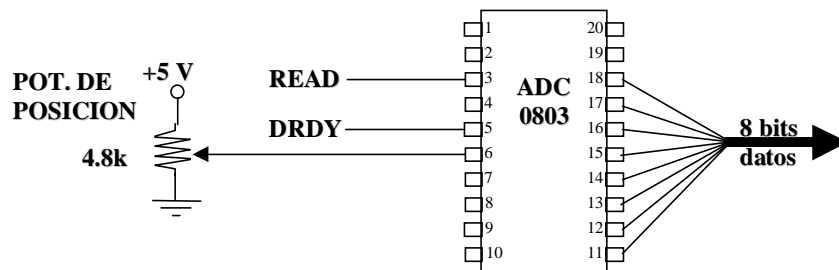


Figura C.2: Diagrama del circuito de conversión, para el sensor de dirección

Apéndice D

Tablas para el cálculo de funciones trigonométricas

En este apéndice se explica la forma en que se calcularon las tablas para implementar las funciones trigonométricas en el microcontrolador. Dichas tablas sirvieron para calcular el seno y coseno de un ángulo, en la subrutina de estimación de parámetros.

Primero, es necesario representar un ángulo en una variable de ocho bits. Siendo así, se dividió el círculo en doscientos cincuenta y seis pasos; por lo tanto, el rango del ángulo va de 0 a 256, que representa de 0 a 2π radianes. El ángulo está discretizado en incrementos de 0.024544 radianes. El valor en radianes de un ángulo representado en una variable de 8 bits, se obtiene usando la siguiente ecuación:

$$\theta = A \frac{2\pi}{256} \quad (D.1)$$

donde θ es el ángulo en radianes, A es el ángulo representado en 8 bits.

El resultado del seno o coseno de un ángulo, sólo puede tomar un valor dentro del rango de -1 a 1. Si el resultado en el microcontrolador se representa únicamente con números enteros, y dependiendo si es un valor positivo se suma, o si es negativo se resta. Entonces, el resultado del seno y coseno sólo puede tomar valores dentro del rango de 0 a 1. Si estos valores se quisieran almacenar en una variable de 8 bits, representando los decimales de la siguiente manera:

2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
----------	----------	----------	----------	----------	----------	----------	----------

Figura D.1: Representación de una variable fraccional usando 8 bits

El valor máximo que se puede representar de esta forma es de 0.99609375.

Usando esta forma de representación numérica tanto para el ángulo como para el resultado, se utilizó una hoja de cálculo para obtener una tabla que relaciona el valor del ángulo, con el coseno del ángulo, ambos representados en variables de 8 bits.

Teniendo en la tabla únicamente un cuarto de ciclo del coseno, se puede reconstruir la respuesta de un ciclo completo del coseno y del seno, usando las siguientes ecuaciones:

Para calcular el Coseno:

Si $(teta \geq 0h)$ y $(teta < 40h)$ entonces $teta = teta$ (positivo, hay que sumar)

Si $(teta \geq 40h)$ y $(teta < 80h)$ entonces $teta = 80h - teta$ (negativo, hay que restar)

Si $(teta \geq 80h)$ y $(teta < C0h)$ entonces $teta = 40h - (C0h - teta)$ (negativo, hay que restar)

Si $(teta \geq C0h)$ y $(teta \leq FFh)$ entonces $teta = 0h - teta$ (positivo, hay que sumar)

Para calcular el Seno:

Si $(teta \geq 0h)$ y $(teta < 40h)$ entonces $teta = 40h - teta$ (positivo, hay que sumar)

Si $(teta \geq 40h)$ y $(teta < 80h)$ entonces $teta = teta - 40h$ (positivo, hay que sumar)

Si $(teta \geq 80h)$ y $(teta < C0h)$ entonces $teta = C0h - teta$ (negativo, hay que restar)

Si $(teta \geq C0h)$ y $(teta \leq FFh)$ entonces $teta = 40h - (0h - teta)$ (negativo, hay que restar)

El valor que se busca en la tabla al final es teta, la cuál solo puede tomar valores de 0 a 64 (0h a 40h en hexadecimal), equivalente a un cuarto de ciclo; la acción que se toma con el resultado, sumar o restar, indica el signo.

Referencias

- [Milliken, Milliken 1995] Milliken, W.F., Milliken D.L., *Race car vehicle dynamics*, SAE, Warrandale, PA, 1995.
- [Latombe 1991] Latombe, Jean-Claude, *Robot Motion Planning*, Kluwer Academic Publishers, U.S.A, 1991.
- [Laumond 1998] Laumond, Jean-Paul, *Robot Motion Planning and Control*, Laboratoire d'Analyse et d'Architecture des Systèmes, LAAS report 97438, 1998.
- [Ogata 1998] Ogata, Katsuhiko, *Ingeniería de control moderna*, Prentice-Hall Hispanoamericana, México, 1998.
- [Dieck 2000] Dieck A., Graciano, *Instrumentación, acondicionamiento electrónico y adquisición de datos*, Editorial Trillas, México, 2000.
- [Kühnel 1998] Kühnel, Claus, *AVR RISC Microcontroller Handbook*, Butterworth-Heinemann, U.S.A., 1998.

