



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y DOCTORADO EN INGENIERÍA

**EXPERIMENTACIÓN CON UNA CONFIGURACIÓN DE ROBOT
MÓVIL DE LLANTAS FRONTALES INDEPENDIENTES
EN SU DIRECCIONABILIDAD**

T E S I S

Que opta para obtener el grado de

MAESTRO EN INGENIERÍA

MECÁNICA - MECATRÓNICA

P R E S E N T A:

ING. OCTAVIO DÍAZ HERNÁNDEZ

TUTOR:

DR. VÍCTOR JAVIER GONZÁLEZ VILLELA



MÉXICO D.F.

ENERO 2010



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Jurado Asignado

Presidente: Dr. López Parra Marcelo
Secretario: Dr. Dorador González Jesús Manuel
Vocal: Dr. González Villela Víctor Javier
1er. Suplente: Dr. Ramírez Reivich Alejandro Cuauhtémoc
2do. Suplente: Dr. Rocha Cózatl Edmundo Gabriel

Lugar donde se realizó la tesis:

Facultad de Ingeniería

**TUTOR DE TESIS:
DR. VÍCTOR JAVIER GONZÁLEZ VILLELA**

FIRMA

ÍNDICE

RESUMEN	6
1. Introducción.....	7
1.1. Aspectos generales	7
1.2. Trabajos previos	8
1.3. El siguiente paso	10
1.4. Objetivos.....	10
1.5. Justificación.....	10
2. Robótica móvil	11
2.1. Robótica.....	11
2.2. Robótica móvil	11
2.2.1. Cinemática de robots con llantas.....	12
2.2.2. Posición del robot.....	13
2.2.2.1. Modelado cinemático de las llantas	14
2.3. Cinemática del robot tipo carro con dos llantas frontales direccionables y una llanta virtual.17	
2.4. Odometría.....	22
3. Requerimientos para el prototipo.....	26
3.1. Actuadores y sensores.....	26
3.1.1. Motores LEGO NXT®	26
3.1.1.2. Modelo matemático.....	28
3.2. Procesador / Controlador /Vínculo de comunicación	28
3.2.1. Ladrillo Inteligente LEGO NXT®.....	28
3.3. Software.....	32
3.3.1. LabVIEW®.....	32
3.3.1.1. Juegos de herramientas (<i>toolkit</i>) para NXT®	34
3.3.2. Matlab / Simulink®	35
4. Sobre la arquitectura del prototipo	37
4.1. Arquitectura del robot	37
4.2. Robot tipo carro con dos llantas frontales direccionables independientemente.....	38
4.3. El programa.....	41
4.4. Control de posición	42
4.5. Implementación de la Odometría.....	43
4.6. Cálculo de los parámetros derivados de la <i>Llanta virtual</i>.....	43
5. Experimentación con el prototipo	45
5.1. Tipos de experimento	45
5.1.1. Operador.....	45
5.1.1.1. Variables de entrada.....	45
5.1.1.2. Variables de salida.....	45
5.1.2. Función arbitraria como entrada.....	46
5.1.2.1. Variables de entrada.....	46
5.1.2.2. Variables de salida.....	47
5.1.3. Obtención de parámetros a partir de SIMULINK®	48
5.1.3.1. Variables de entrada.....	49
5.1.3.2. Variables de salida.....	50
6. Conclusiones.....	53
Apéndice A.....	55
Motores de corriente directa.....	55

A.1 Los principios básicos de funcionamiento.....	55
A.2 Modelado de un motor de corriente directa	57
Apéndice B	59
El programa.....	59
Apéndice C	63
Artículo presentado en el Congreso Internacional Anual de la SOMIM, 23-25 Septiembre 2009, Cd Obregón, Sonora, México.....	63
REFERENCIAS	70

Tabla de figuras.

Figura 1. Esquema de dos llantas direccionables en un robot móvil. Tomada de Mauder, 2008	9
Figura 2. Posición del robot sobre un marco de referencia fijo, adaptado de Gonzalez-Villela, 2006	13
Figura 3. Llantas fijas y orientables centradas. Adaptado de Gonzalez-Villela, 2006	15
Figura 4. Llantas orientables descentradas. Adaptado de Gonzalez-Villela, 2006.....	16
Figura 5. Configuraciones básicas de robots móviles. Modificadas de Gonzalez-Villela, 2006	17
Figura 6. Esquema del Robot tipo carro ubicado en un marco de referencia XY. Tomado de Gonzalez-Villela, 2006	18
Figura 7. Representación de robot móvil. Modificada de Gonzalez-Villela, 2006.....	23
Figura 8. Dibujo de un motor NXT®, tomado de <i>NXT® motor internals, 2009</i>	26
Figura 9. Motor NXT. a) Vista general de las piezas, b) Reducción de engranes (ver Tabla 2), c) Ventana ranurada del <i>encoder</i> de cuadratura, d) Circuito impreso de acondicionamiento de señales de entrada y salida. Figuras tomadas de <i>NXT® motor internals, 2009</i>	27
Figura 10. El ladrillo inteligente NXT de LEGO® MINDSTORMS.....	30
Figura 11. Configuraciones para el movimiento del motor. a) Giro horario b) Giro antihorario c) Freno d) Sin señal. Figuras tomadas de Gasperi, 2007	31
Figura 12. Una onda cuadrada, mostrando las definiciones de y_{min} , y_{max} , T y D	31
Figura 13. Señales del <i>encoder</i> de cuadratura	32
Figura 14. Muestra de LabVIEW® y su versión	33
Figura 15. Ejemplos de las herramientas usadas en LabVIEW®	34
Figura 16. Controles directos desde LabVIEW® hacia el NXT. a) Panel general, b) Panel de conexión USB o <i>bluetooth</i> ®, c) Panel de Salidas, control de los puertos A, B y C de salida.....	35
Figura 17. Muestra de la programación en Simulink® donde un robot sigue una trayectoria	36
Figura 18. Arquitectura general del sistema robótico.....	38
Figura 19. Esquema del Robot tipo carro ubicado en un marco de referencia XY. Tomado de Gonzalez-Villela, 2006	38
Figura 20. Foto de una llanta frontal.....	39
Figura 21. Foto de una llanta trasera y su <i>encoder</i> de cuadratura (modelo EP4). Para detalles ver <i>US DIGITAL™</i> , 2009	39
Figura 22. Vista superior del robot. Punto P	40
Figura 23. Foto del robot lateral	40
Figura 24. Panel de control principal.....	41
Figura 25. Sección del diagrama de bloques funcionales. Ver apéndice B para los detalles.....	41
Figura 26. Esquema de control de posición. Adaptado de Groover, 1990.	42
Figura 27. Control de posición para los ángulos frontales derecho e izquierdo.	43
Figura 28. Esquema en LabVIEW® del módulo de odometría	43
Figura 29. Programación del algoritmo matemático.....	44
Figura 30. Recorte del diagrama de bloques para controlar al robot desde el teclado	45
Figura 31. Diagrama de bloques de lectura de datos externos	47
Figura 32. Fotografías tomadas a los movimientos durante el desplazamiento del robot. Las líneas rojas son extensiones de los ejes de rotación de las llantas. El punto azul es el ICR	51

Tabla de gráficos

Gráfica 1. Valores obtenidos con LabVIEW® para determinar un modelo aproximado del motor NXT®.....	28
Gráfica 2. Onda Senoidal con amplitud de $\frac{1}{2}$	46
Gráfica 3. Movimiento de la llantas derecha(a) e izquierda (b).....	47
Gráfica 4. Velocidades de las llantas (a) derecha e (b) izquierda durante el recorrido	47
Gráfica 5. Recorrido del punto P en el robot. Muestras obtenidas de la odometría (ver secciones 2.4 y 4.5).....	48
Gráfica 6. Forma de la trayectoria en el plano x y	49
Gráfica 7 Comportamiento del ángulo de la llanta virtual durante la trayectoria propuesta	49
Gráfica 8. Comportamiento de la velocidad de la llanta virtual durante la trayectoria propuesta.....	49
Gráfica 9. Comparación entre los ángulos reales contra los de referencia de la llanta (a)derecha y la (b) izquierda durante la trayectoria propuesta	50
Gráfica 10. Comparación entre las velocidades angulares reales contra las de referencia de la llanta (a)derecha y la (b) izquierda durante la trayectoria propuesta	50
Gráfica 11. Recorrido del punto P en el robot. Muestras obtenidas de la odometría (ver secciones 2.4 y 4.5)...	51

RESUMEN

En la actualidad los robots móviles son de gran interés por sus aplicaciones en la ingeniería, donde la coordinación de múltiples llantas ha sido objeto de diversos estudios y sus aproximaciones han sido tanto mecánicas como matemáticas. Se sabe que los robots móviles están sujetos a restricciones noholonómicas, por lo que este estudio requiere usar teoría noholonómica.

Se experimenta con un robot móvil de dos llantas motorizadas frontales que pueden ser orientadas independientemente. Algunos de los parámetros de importancia que coordinan las llantas del robot son determinados por un concepto matemático conocido como **llanta virtual** (rueda sin masa ni motor) establecido a partir de la cinemática de robots móviles.

Se empleó un prototipo de robot móvil capaz de comunicarse con una computadora personal. Fueron usadas piezas prefabricadas de LEGO®, tales como engranes, motores y la unidad de procesamiento de *Mindstorms*® NXT®, etc. La comunicación fue inalámbrica vía *Bluetooth*®, la implementación de algoritmos complejos se realizó en LabVIEW® 8.6 de *National Instruments*.

Se han obtenido resultados en los que se demuestra la coordinación de las llantas a través de las variables asociadas a la llanta virtual. Fue posible mover al robot en un espacio de trabajo mediante instrucciones de un operador. Se generaron parámetros de control para que el robot siguiera una trayectoria.

Las implicaciones del presente estudio en la academia y la industria son relevantes, debido a que el empleo de la robótica móvil para controlar vehículos puede ser extendido a diferentes configuraciones de robots con llantas.

Este modelo de robot corresponde a un vehículo tipo carro con posibilidades de mejorar el sistema de conducción de los automóviles actuales.

1. Introducción

1.1. Aspectos generales

La mecatrónica, que es la integración sinérgica de las ingenierías mecánica, electrónica e informática [Dinsdale, 1989], es usada como herramienta para el diseño de productos inteligentes y flexibles como lo deben ser los robots móviles. Este campo de la robótica está embebido en problemas tales como la integración de sensores, actuadores, control, etc.

Los aspectos mecánicos de un robot móvil incluyen el diseño apropiado de la base (chasis), los mecanismos de impulsión, maniobrabilidad, tracción, estabilidad y la navegación, además de la forma, el tamaño y la controlabilidad del robot [Gopalakrishnan, 2004].

La controlabilidad se refiere a la existencia de controles en lazo abierto (señales de entrada) capaces de dirigir el sistema de un estado inicial q^0 hasta un estado final q^d durante un intervalo de tiempo finito [Kolmanovsky, 1995].

El controlador en los robots móviles es en general un microprocesador, un microcontrolador embebido o una computadora personal (PC), mientras que el lenguaje de control puede ser de bajo o alto nivel [Gopalakrishnan, 2004].

Los métodos para controlar a estos robots son múltiples; en algunos casos se controlan por interacción directa con una persona, conocido como teleoperación [Lim, 2003; Lee, 2006]. En otros casos son vehículos autoguiados o inteligentes, en cuyo caso requieren elementos que los provean de autonomía, por mencionar algunos están sensores de luz infrarroja [Gopalakrishnan, 2004], ultrasónicos [Masek, 2000; Anchour, 2001], de posición (GPS) [Nyholm, 2007], etc.

La inteligencia se les ha proveído gracias a técnicas como lógica difusa [Islam, 2006], redes neuronales artificiales [Hendzel, 2007], algoritmos genéticos [Liu, 2006], procesamiento digital de señales [Fazli, 2004; 2005].

En el presente trabajo se pretende controlar un robot tipo carro con llantas delanteras independientes en su direccionalidad, de forma tal que haya armonía en sus llantas para mover al robot móvil dentro de un espacio de trabajo.

1.2. Trabajos previos

En 1990, Matsumoto y colaboradores hicieron un robot con cuatro llantas direccionables. Proponiendo el control mediante el ángulo de una de las llantas y una variable que consideraba el derrape de la otra llanta.

David B. Reister, en 1993 propuso un algoritmo para controlar las llantas de un robot HERMIES-III, porque sabía que controlando independientemente las llantas, podría haber derrape y en consecuencia errores. Consideró la posibilidad de manejar la orientación de las llantas con un “ente” matemático en común para todas las llantas. Reister propuso que el resultado de dividir la velocidad traslacional entre el radio del centro de rotación de las llantas debería ser el mismo para cada llanta.

En 1993, Burke y Durrant-Whyte, trabajaron con un robot “reconfigurable” de llantas direccionables que se coordinaban con una llanta imaginaria mediante restricciones geométricas, pero existe la necesidad de compensar los efectos del deslizamiento que provocan errores.

Thuilot y D’Andrea-Novel en 1996, mencionan que para coordinar varias llantas direccionables se puede usar una técnica de control conocida como “realimentación de estados”. El seguimiento de trayectorias de un robot móvil a partir del modelado cinemático es posible, sin embargo es demasiado compleja la implementación y existen estados para el robot donde podría no ser controlable.

En 2000, Matumbara en conjunto con Durrant-Whyte, retoman los robots reconfigurables y el concepto de la llanta imaginaria, adicionalmente proponen el uso de la cinemática de las llantas, además de la geometría planteada en 1993.

En 2002, Sekhavat, propone un modelo de robot con llantas direccionables frontales para transportación en terminales aéreas o carros de autoservicio, modelándolo como desde el punto de vista cinemático. Su robot, llamado *Cycab*, tiene llantas traseras motorizadas su rotación está especificada de acuerdo a los ángulos de las llantas delanteras. Sin embargo dejan abierta la posibilidad de un control nuevo que ayude a coordinar al sistema.

En 2006, González-Villela describe un elemento matemático que coordina los movimientos de una o más llantas independientemente direccionables, orientando los ejes de rotación hacia el centro instantáneo de rotación, él emplea técnicas de la cinemática en robots móviles, fundamentos de teoría de control. Logrando simulaciones exitosas de su robot móvil, faltando algunos experimentos que lo comprueben.

En el trabajo de Markus Mauder (2008) se ha reportado el control de dos llantas independientes, dispuestas una en el frente y otra en la parte posterior. Se puede ver la movilidad de las llantas propuestas por Mauder en la Figura 1.

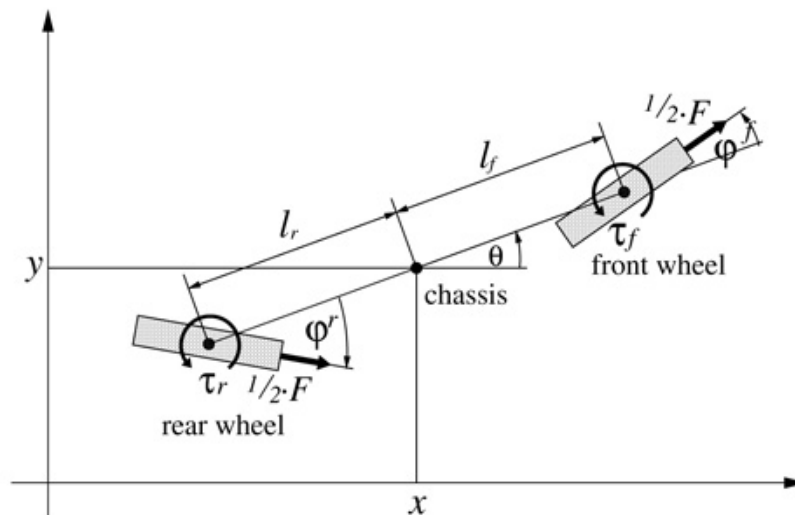


Figura 1. Esquema de dos llantas direccionables en un robot móvil. Tomada de Mauder, 2008

Este control consiste en una retroalimentación para la linealización del sistema no-holonómico basado en el modelo cinemático inverso y superficies deslizantes. Sin embargo es teórico y sus pruebas se limitan a simulaciones matemáticas.

1.3. El siguiente paso

El reto de coordinar llantas direccionables ha sido objeto de estudio en el mundo. Dentro del contexto de esta línea de investigación se ha desarrollado una teoría que se desea comprobar. La “llanta virtual” es un concepto matemático, generado a través de la robótica móvil, capaz de coordinar los ángulos y las velocidades de otras llantas, ya sean motorizadas o no. Se ha determinado la necesidad de recabar datos experimentales a partir de un prototipo de robot móvil. La configuración que se ha planteado en este trabajo son dos llantas frontales independientes en su direccionabilidad y dos llantas traseras, siendo de interés porque representa un vehículo con cuatro ruedas con características similares a un automóvil.

1.4. Objetivos

- Comprobar que la teoría sobre la llanta virtual funciona sobre el prototipo propuesto.
- Ensamblar un robot móvil con llantas delanteras direccionables motorizadas y dos llantas traseras fijas, para recabar datos experimentales.
- Coordinar la orientación de las llantas delanteras.
- Coordinar las velocidades angulares de los motores de cada llanta delantera.

1.5. Justificación

Se experimentará con un robot móvil de llantas frontales direccionables independientemente para demostrar que se puede coordinar los parámetros necesarios de dichas llantas empleando el concepto de la llanta virtual, con el fin de controlar un punto asociado al robot.

En la práctica, el resultado obtenido es de utilidad en el campo de la robótica móvil, donde la coordinación de llantas se ha estudiado ampliamente en épocas recientes y ahora se propone emplear una metodología basada en el estudio formal de los robots móviles.

En la industria automotriz es posible revolucionar la forma de conducción y de diseño de vehículos tipo carro con direccionabilidad en las llantas delanteras.

2. Robótica móvil

2.1. Robótica.

La **robótica** es una ciencia o rama de la tecnología, que estudia el diseño y construcción de máquinas capaces de desempeñar tareas realizadas por el ser humano o que requieren del uso de inteligencia. [McKerrow, 1993]. Los robots actuales son obras de la ingeniería concebidos para realizar trabajos productivos o recreativos, fundamentados esencialmente en conocimientos matemáticos y tecnológicos.

Los robots requieren de la construcción de sistemas de control “segmentados”, es decir, se descomponen el sistema robótico en unidades funcionales que trabajarán en conjunto para realizar tareas. Los componentes que integran a un robot están organizados en la Tabla 1 donde se leen tres rubros a tomar en cuenta en el diseño de los requerimientos de procesamiento y alcanzar objetivos predeterminados.

SENSORES	CONTROL	ACTUADORES
<ul style="list-style-type: none"> • Externos • Internos 	<ul style="list-style-type: none"> • Acondicionamiento de entradas • Ejecución de programa • Cálculo de variables • Acondicionamiento de salidas 	<ul style="list-style-type: none"> • Eléctricos • Mecánicos • Etc.

Tabla 1. Componentes de un sistema robótico

2.2. Robótica móvil

Estudia la cinemática y la dinámica de los robots con capacidad de desplazarse dentro de un espacio de trabajo considerando que no hay **vínculos mecánicos** entre el robot y la base de referencia fija/inercial.

Un punto P es un lugar geométrico definido por coordenadas $[x, y, z]$ en el robot, o referenciado a él, que se ocupa como el punto de análisis de movimiento y el control del robot.

El punto P en los manipuladores estudiados en la robótica convencional se ubica generalmente en el Órgano Terminal, mientras que en la robótica móvil, el punto P casi siempre está situado en el chasis del robot (aunque podría estar fuera de él).

La posición del punto P en ambas robóticas es modificada por fuerzas internas (por ejemplo, uso de motores) o acción de fuerzas externas (por ejemplo, el viento contra las velas de un barco). Al punto P se le asocia un Sistema de Referencia Móvil (SRM), que se moverá junto con él y ayudará a definir elementos de la estructura mecánica del robot.

Este tipo de sistema mecánico se caracteriza por poseer restricciones cinemáticas NO integrables, considerándole como un sistema noholonómico. Es por eso que durante aproximadamente cuatro décadas se han estudiado los algoritmos capaces de controlar a los robots móviles, lo que ha llevado al estudio de los elementos que definen y modifican la posición del robot [Campion,1993].

Cabe mencionar que los robots móviles pueden ser clasificados de distintas formas, se propone la siguiente:

- Terrestres
 - Con llantas
 - Con extremidades
- Aéreos
- Acuáticos
- Híbridos

Sin embargo, su estudio cinemático y dinámico está definido por los principios matemáticos dictados en la robótica móvil.

2.2.1. Cinemática de robots con llantas

Los vehículos con ruedas son la solución más simple y eficiente para conseguir movilidad. En este apartado se estudia el modelo de posición cinemática, que es la representación más simple en ecuaciones de estado, dada para describir globalmente al robot

2.2.2. Posición del robot

Un robot con llantas es un vehículo capaz de realizar movimiento autónomo sin un conductor externo porque está equipado, para su movimiento, con motores manejados por un procesador, además de sensores que proveen información del mundo. Para el análisis, se considera que los robots móviles son no deformables y que se mueven sobre un plano horizontal. La posición del robot en el plano se describe como se muestra en la Figura 2 sobre una base inercial ortonormal¹ arbitraria $(I_x, I_y, 0)$ que es tomada como marco de referencia para ubicar los movimientos del robot. Asimismo se ubica un punto P con un Sistema de Referencia Móvil $(x_1, y_1, 0)$. [Campion,1993]

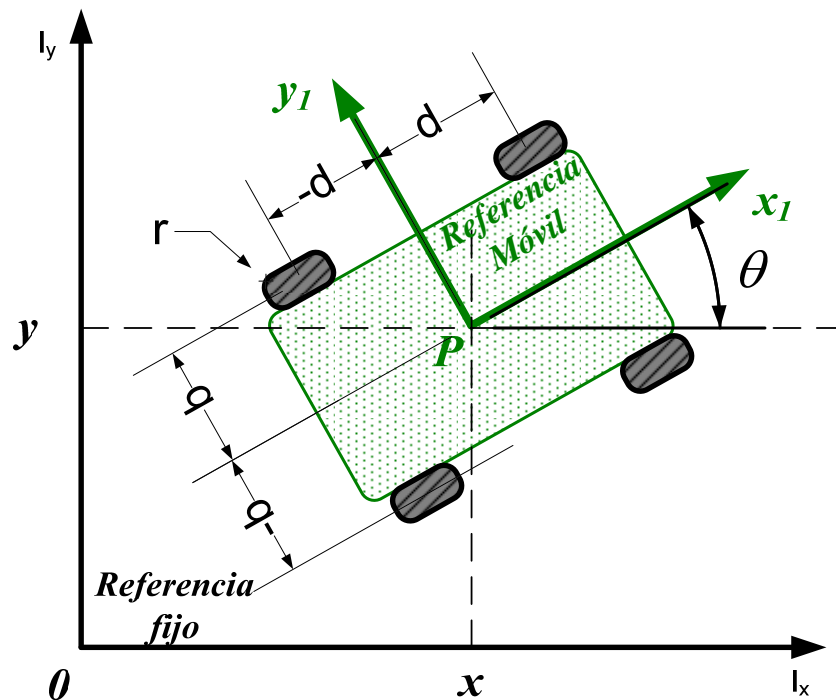


Figura 2. Posición del robot sobre un marco de referencia fijo, adaptado de Gonzalez-Villela, 2006

La posición en el robot está descrita por $\varepsilon = (x, y, \theta)^T$, donde x y y son las coordenadas del punto P sobre la base inercial y θ es el ángulo formado por el eje longitudinal del robot y el

¹ Un conjunto de vectores es **ortonormal** si es a la vez un conjunto con ángulos rectos entre sí y la norma de cada uno de sus vectores es igual a uno.

eje I_x del sistema de referencia fijo. La derivada de ε con respecto al tiempo es $\dot{\varepsilon} = (\dot{x}, \dot{y}, \dot{\theta})^T$, donde (\dot{x}, \dot{y}) son componentes de la velocidad lineal y $\dot{\theta}$ es la velocidad angular del robot.

También se puede definir una matriz de rotación como la siguiente, que ubicará al punto P al sistema de referencia fijo, en todo momento.

$$R(\theta) = \begin{pmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{Ecuación 2.1}$$

2.2.2.1. Modelado cinemático de las llantas

Se asume también que durante el movimiento, el plano de cada llanta se mantiene vertical y la llanta rota sobre su eje horizontal. Se distinguen dos clases básicas de ruedas idealizadas: las llantas **convencionales** y las llantas **“Suecas”** (*Swedish*). En cada caso, se supone que el contacto entre la rueda y el suelo se reduce a un solo punto del plano. Para una rueda convencional, el contacto entre la rueda y el suelo para rodar, idealmente no debe deslizarse. Esto significa que la velocidad del punto de contacto es igual a cero, e implica que sus componentes paralela y ortogonal al plano de la rueda son iguales a cero.

Las llantas convencionales se pueden subclasificar como fijas, centradas orientables y descentradas orientables. Para efectos de este trabajo no se profundizará en las llantas Suecas, para mayor información se puede consultar Campion, 1993.

Fijas. Son aquellas que NO cambian de posición angular con respecto al marco de referencia del robot. En la Figura 3 se aprecia al punto P del que parte un sistema de referencia móvil. El punto A esta definido por coordenadas (d, b) , el ángulo α es la orientación con respecto al eje paralelo a x_1 , cuyo valor en este caso es **constante**. El movimiento angular de la llanta es ϕ y su velocidad angular de la llanta alrededor de su eje (horizontal) es $\dot{\phi}$ y su radio es r , con las dos últimas se define la velocidad lineal de la llanta $\dot{\phi} = \phi \cdot r$.

Restricciones cinemáticas:

En el plano de la llanta: $[\cos \alpha \quad \sin \alpha \quad d \sin \alpha - b \cos \alpha] \dot{\varepsilon} - r \dot{\varphi} = 0$ Ecuación 2.2

Ortogonal al plano de la llanta: $[-\sin \alpha \quad \cos \alpha \quad d \cos \alpha - b \sin \alpha] \dot{\varepsilon} = 0$ Ecuación 2.3

Donde d , b y α son constantes. Mientras que $\varphi(t)$ es un ángulo variante en el tiempo.

Centradas orientables. En la Figura 3 se aprecia al punto P del que parte un sistema de referencia móvil. El punto A esta definido por coordenadas (d,b) , el ángulo α es la orientación con respecto al eje paralelo a x_1 , cuyo valor en este caso es **variable**. El movimiento angular de la llanta es φ y su velocidad angular de la llanta alrededor de su eje (horizontal) es $\dot{\varphi}$ y su radio es r , con las dos últimas se define la velocidad lineal de la llanta $\dot{\phi} = \dot{\varphi} \cdot r$.

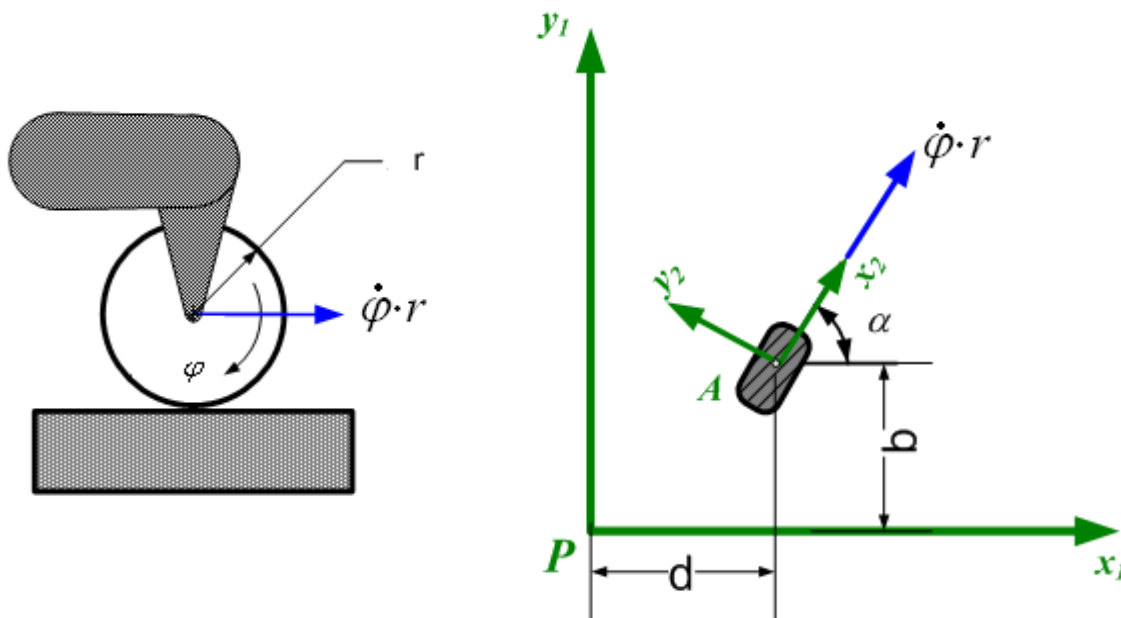


Figura 3. Llantas fijas y orientables centradas. Adaptado de Gonzalez-Villela, 2006

Restricciones cinemáticas:

En el plano de la llanta: $[\cos \alpha \quad \sin \alpha \quad d \sin \alpha - b \cos \alpha] \dot{\varepsilon} - r \dot{\varphi} = 0$ Ecuación 2.4

Ortogonal al plano de la llanta: $[-\sin \alpha \quad \cos \alpha \quad d \cos \alpha - b \sin \alpha] \dot{\varepsilon} = 0$ Ecuación 2.5

Donde d y b son constantes, mientras que $\varphi(t)$ y $\alpha(t)$ son ángulos variantes en el tiempo.

Descentradas orientables. Se les conoce como llantas omnidireccionales, ruedas “locas” o “*castor wheels*”. En la Figura 4 se aprecia al punto P del que parte un sistema de referencia móvil. El punto A esta definido por coordenadas (d,b) , el ángulo α es la orientación con respecto al eje paralelo a x_1 , cuyo valor en este caso es **variable**. En adición a las anteriores, existe una distancia e entre los puntos A y B, usualmente constante. El movimiento angular de la llanta es φ y su velocidad angular de la llanta alrededor de su eje (horizontal) es $\dot{\varphi}$ y su radio es r , con las dos últimas se define la velocidad lineal de la llanta $\dot{\varphi} \cdot r$.

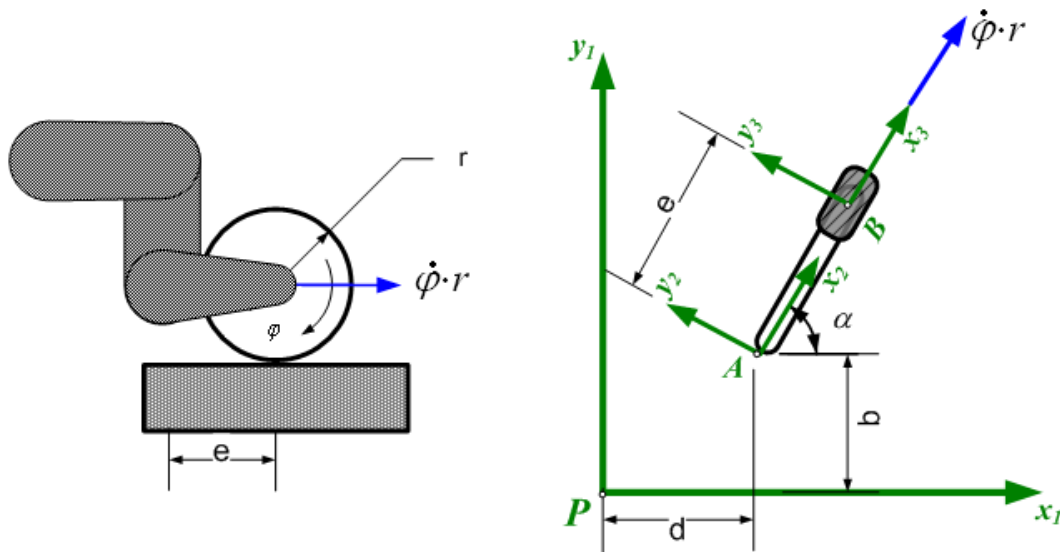


Figura 4. Llantas orientables descentradas. Adaptado de Gonzalez-Villela, 2006

Restricciones cinemáticas:

En el plano de la llanta: $[\cos \alpha \quad \sin \alpha \quad d \sin \alpha - b \cos \alpha] \dot{\varepsilon} - r \dot{\varphi} = 0$ Ecuación 2.6

Ortogonal al plano de la llanta: $[-\sin \alpha \quad \cos \alpha \quad e + d \cos \alpha - b \sin \alpha] \dot{\varepsilon} - e \dot{\alpha} = 0$ Ecuación 2.7

Donde d, b y e son constantes, mientras que $\varphi(t)$ y $\alpha(t)$ son ángulos variantes en el tiempo.

2.2.3. Ejemplos de configuraciones de robots móviles

Las llantas pueden estar motorizadas y dar movimiento al robot. Las combinaciones de los tipos de llantas, su motorización o el número de ellas, generan una gran cantidad de configuraciones de robots móviles. Se presentan algunos tipos a manera de ejemplo. Se

habla de los grados de libertad de movilidad (δ_m , *mobility*) y de direccionabilidad (δ_s , *steerability*). El primero tiene que ver con el grado de libertad instantáneo, se asocia a la velocidad lineal del robot. El segundo es un grado de libertad no instantáneo (requiere de cálculos previos), asociado al grado de orientabilidad del robot y de cuántas llantas son linealmente independientes. Se anota: Tipo (δ_m, δ_s).

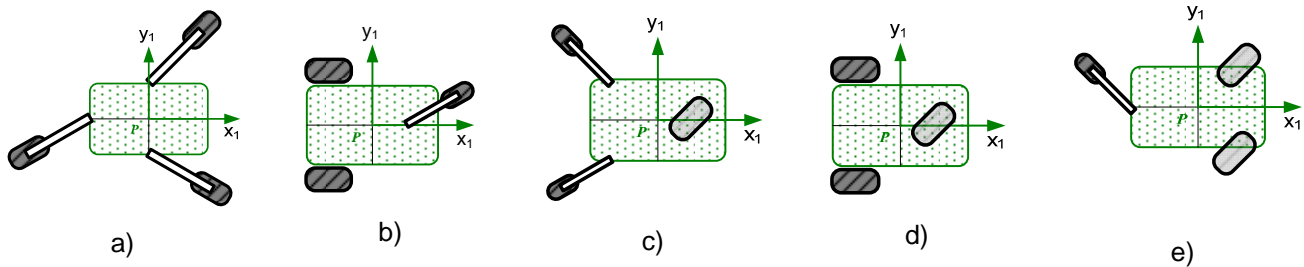


Figura 5. Configuraciones básicas de robots móviles. Modificadas de Gonzalez-Villela, 2006

- a) Robot omnidireccional Tipo (3,0). Tres llantas orientables descentradas.
- b) Robot diferencial, Tipo (2,0). Dos llantas fijas y una orientable descentrada.
- c) Robot Tipo (2,1). Dos llantas orientables descentradas y una fija.
- d) Triciclo. Robot Tipo (1,1). Una llanta orientable centrada y dos fijas.
- e) Tipo (1,2). Una llanta orientable descentrada y dos orientables centradas.

2.3. Cinemática del robot tipo carro con dos llantas frontales direccionables y una llanta virtual.

En la Figura 6 se observa el robot que es el caso de estudio para el presente trabajo. Se tienen dos llantas frontales direccionables independientes motorizadas y que cuentan con un motor adicional para modificar su ángulo α correspondiente. Por otro lado, la llanta ubicada a una distancia d_v del punto P es la **llanta virtual**, es decir, no posee masa, ni tiene motor. Se le asocia un radio r y un $\alpha(t)$ y una $\varphi(t)$ que son variantes en el tiempo, como el resto de las llantas. Por último, las llantas traseras son fijas y no motorizadas.

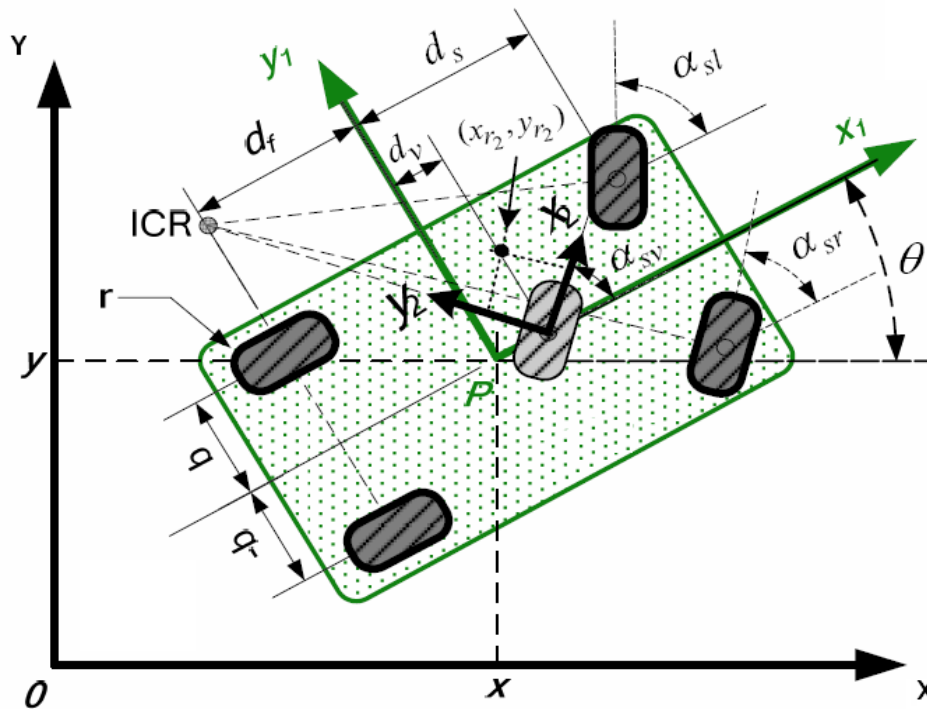


Figura 6. Esquema del Robot tipo carro ubicado en un marco de referencia XY. Tomado de Gonzalez-Villela, 2006

Su cinemática se describe con las ecuaciones antes descritas de cada llanta tomando en cuenta la llanta virtual. Las coordenadas de configuración son:

$$q_1 = [x_1, y_1, \theta, \alpha_{sr}, \alpha_{sl}, \alpha_{sv}, \phi_{fr}, \phi_{fl}, \phi_{sr}, \phi_{sl}, \phi_{sv}]^T \quad \text{Ecuación 2.8}$$

Donde q_1 es el vector de coordenadas generalizadas con respecto a $\{P, x_1, y_1\}$. α_{sr}, α_{sl} y α_{sv} son los ángulos de orientación de las llantas: frontal derecha, frontal izquierda y virtual, respectivamente. $\phi_{fr}, \phi_{fl}, \phi_{sr}, \phi_{sl}$ y ϕ_{sv} son los ángulos de rotación de las llantas: trasera derecha, trasera izquierda, frontal derecha, frontal izquierda y virtual, respectivamente. d_v es la distancia del punto P al eje perpendicular al eje de rotación de la llanta virtual. d_s es la distancia del punto P al eje perpendicular al eje de rotación de las llantas frontales. d_f es la distancia del punto P al eje perpendicular al eje de rotación de las llantas traseras. b es la mitad de la distancia entre las llantas traseras.

Se toman las ecuaciones 2.4 y 2.5 para sustituir las variables y las constantes, considerando un rodaje perfecto (sin deslizamientos), donde $\alpha = 0$ para las llantas fijas, se tiene:

- Restricciones cinemáticas ortogonales al plano de la llanta, sin deslizamiento lateral:

$$\begin{aligned}
 \dot{y}_1 - d_f \dot{\theta} &= 0 \\
 \dot{y}_1 - d_f \dot{\theta} &= 0 \\
 -\sin(\alpha_{sr}) \dot{x}_1 + \cos(\alpha_{sr}) \dot{y}_1 + [d_s \cos(\alpha_{sr}) - b \sin(\alpha_{sr})] \dot{\theta} &= 0 \\
 -\sin(\alpha_{sl}) \dot{x}_1 + \cos(\alpha_{sl}) \dot{y}_1 + [d_s \cos(\alpha_{sl}) + b \sin(\alpha_{sl})] \dot{\theta} &= 0 \\
 -\sin(\alpha_{sv}) \dot{x}_1 + \cos(\alpha_{sv}) \dot{y}_1 + d_v \cos(\alpha_{sv}) \dot{\theta} &= 0
 \end{aligned}$$

Ecuación 2.9

- Restricciones cinemáticas en el plano de la llanta, rodamiento sin deslizamiento:

$$\begin{aligned}
 \dot{x}_1 + b \dot{\theta} - r \dot{\phi}_{fr} &= 0 \\
 \dot{x}_1 + b \dot{\theta} - r \dot{\phi}_{fl} &= 0 \\
 \cos(\alpha_{sr}) \dot{x}_1 + \sin(\alpha_{sr}) \dot{y}_1 + [d_s \sin(\alpha_{sr}) + b \cos(\alpha_{sr})] \dot{\theta} - r \dot{\phi}_{sr} &= 0 \\
 \cos(\alpha_{sl}) \dot{x}_1 + \sin(\alpha_{sl}) \dot{y}_1 + [d_s \sin(\alpha_{sl}) - b \cos(\alpha_{sl})] \dot{\theta} - r \dot{\phi}_{sl} &= 0 \\
 \cos(\alpha_{sv}) \dot{x}_1 + \sin(\alpha_{sv}) \dot{y}_1 + d_v \sin(\alpha_{sv}) \dot{\theta} - r \dot{\phi}_{sv} &= 0
 \end{aligned}$$

Ecuación 2.10

Aquí se introducen las velocidades descritas en:

$$\dot{q}_1 = [\dot{x}_1, \dot{y}_1, \dot{\theta}, \dot{\alpha}_{sr}, \dot{\alpha}_{sl}, \dot{\alpha}_{sv}, \dot{\phi}_{fr}, \dot{\phi}_{fl}, \dot{\phi}_{sr}, \dot{\phi}_{sl}, \dot{\phi}_{sv}]^T$$

Ecuación 2.11

Donde \dot{q}_1 es el vector de velocidades generalizadas vistas desde $\{P, x_1, y_1\}$. Además \dot{x}_1 y \dot{y}_1 son las velocidades lineal y lateral respectivamente del robot. $\dot{\theta}$ es la velocidad angular del robot. Asimismo, $\dot{\alpha}_{sr}, \dot{\alpha}_{sl}$ y $\dot{\alpha}_{sv}$ son las velocidades angulares de las llantas: frontal derecha, frontal izquierda y virtual, respectivamente. $\dot{\phi}_{fr}, \dot{\phi}_{fl}, \dot{\phi}_{sr}, \dot{\phi}_{sl}$ y $\dot{\phi}_{sv}$ son las velocidades rotacionales de las llantas: trasera derecha, trasera izquierda, frontal derecha, frontal izquierda y virtual, respectivamente.

Las ecuaciones 2.9 y 2.10 se pueden representar en una matriz $A \in R^{11 \times 10}$ en términos de \dot{q}_1 y en el vector de velocidades generalizadas, que desarrollado se escribe como a continuación.

$$\begin{bmatrix}
 0 & 1 & d_f & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & d_f & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -\sin \alpha_{sr} & \cos \alpha_{sr} & d_s \cos \alpha_{sr} - b \sin \alpha_{sr} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -\sin \alpha_{sl} & \cos \alpha_{sl} & d_s \cos \alpha_{sl} + b \sin \alpha_{sl} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -\sin \alpha_{sv} & \cos \alpha_{sv} & d_v \cos \alpha_{sv} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 1 & 0 & b & 0 & 0 & 0 & -r & 0 & 0 & 0 \\
 1 & 0 & -b & 0 & 0 & 0 & 0 & -r & 0 & 0 \\
 \cos \alpha_{sr} & \sin \alpha_{sr} & d_s \sin \alpha_{sr} + b \cos \alpha_{sr} & 0 & 0 & 0 & 0 & 0 & -r & 0 \\
 \cos \alpha_{sl} & \sin \alpha_{sl} & d_s \sin \alpha_{sl} - b \cos \alpha_{sl} & 0 & 0 & 0 & 0 & 0 & 0 & -r \\
 \cos \alpha_{sv} & \sin \alpha_{sv} & d_v \sin \alpha_{sv} & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}
 \begin{bmatrix}
 \dot{x}_1 \\
 \dot{y}_1 \\
 \dot{\theta} \\
 \dot{\alpha}_{sr} \\
 \dot{\alpha}_{sl} \\
 \dot{\alpha}_{sv} \\
 \dot{\phi}_{fr} \\
 \dot{\phi}_{fl} \\
 \dot{\phi}_{sr} \\
 \dot{\phi}_{sl} \\
 \dot{\phi}_{sv}
 \end{bmatrix}
 = 0 \quad \text{Ecuación 2.12}$$

Para obtener los grados de libertad de direccionabilidad, movilidad y maniobrabilidad que definen al robot, se obtienen los rangos² de las submatrices relacionadas con las llantas fijas O_f y con las llantas direccionables O_s de tal forma que las propiedades cinemáticas de movimiento se resumen de la siguiente forma:

$$\text{Movilidad} = 3 - \text{rango de } O_f = 3 - 2 = 1$$

$$\text{Direccionabilidad} = \text{rango de } O_s = 1$$

$$\text{Maniobrabilidad} = \text{Movilidad} + \text{Direccionabilidad} = 2$$

Para el grado de movilidad se define $\eta = \frac{r\dot{\phi}_{sv}}{d_v - d_f}$, donde $r\dot{\phi}_{sv}$ representa la velocidad lineal

de la llanta virtual y coincide con la velocidad lineal del robot. Mientras que $\frac{1}{d_v - d_f}$ es un

escalamiento para hacer más simple la representación en variables de estado, que se verán adelante.

De la misma forma se define $\zeta = \dot{\alpha}_{sv}$ para el grado de direccionabilidad, donde se sabe que $\dot{\alpha}_{sv}$ es la velocidad angular de orientación de la llanta virtual.

² RANGO de una matriz es el número de líneas de esa matriz (filas o columnas) que son linealmente independientes.

Si se resuelve la ecuación 2.12 usando las dos variables anteriores, se tiene:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{\theta} \\ \dot{\alpha}_{sr} \\ \dot{\alpha}_{sl} \\ \dot{\alpha}_{sv} \\ \dot{\phi}_{fr} \\ \dot{\phi}_{fl} \\ \dot{\phi}_{sr} \\ \dot{\phi}_{sl} \\ \dot{\phi}_{sv} \end{bmatrix} = \begin{bmatrix} d_v \cos \theta \cos \alpha_{sv} - d_f \cos(\theta + \alpha_{sv}) & 0 \\ d_v \sin \theta \cos \alpha_{sv} - d_f \sin(\theta + \alpha_{sv}) & 0 \\ \sin \alpha_{sv} & 0 \\ 0 & \frac{(d_s - d_f)(d_v - d_f)}{[b \sin(\alpha_{sv}) + (d_v - d_f) \cos \alpha_{sv}]^2 + (d_f - d_s)^2 \sin(\alpha_{sv})^2} \\ 0 & \frac{(d_s - d_f)(d_v - d_f)}{[b \sin(\alpha_{sv}) - (d_v - d_f) \cos \alpha_{sv}]^2 + (d_f - d_s)^2 \sin(\alpha_{sv})^2} \\ \frac{(d_v - d_f) \cos \alpha_{sv} + b \sin \alpha_{sv}}{r} & 1 \\ \frac{(d_v - d_f) \cos \alpha_{sv} - b \sin \alpha_{sv}}{r} & 0 \\ \frac{r}{[b \sin(\alpha_{sv}) + (d_v - d_f) \cos \alpha_{sv}]^2 + (d_f - d_s)^2 \sin(\alpha_{sv})^2} & 0 \\ \frac{r}{[b \sin(\alpha_{sv}) - (d_v - d_f) \cos \alpha_{sv}]^2 + (d_f - d_s)^2 \sin(\alpha_{sv})^2} & 0 \\ \frac{d_v - d_f}{r} & 0 \end{bmatrix} \begin{bmatrix} \eta \\ \zeta \end{bmatrix}$$

Ecuación 2.13

De la ecuación 2.13 se deducen las fórmulas para calcular los ángulos que deben coordinar las llantas delanteras

$$\alpha_{sr}(\alpha_{sv}) = a \tan 2 \left[\frac{-(d_f - d_s) \text{sen}(\alpha_{sv})}{b \text{sen}(\alpha_{sv}) - (d_f - d_v) \cos(\alpha_{sv})} \right] \quad \text{Ecuación 2.14}$$

$$\alpha_{sl}(\alpha_{sv}) = a \tan 2 \left[\frac{-(d_f - d_s) \text{sen}(\alpha_{sv})}{-b \text{sen}(\alpha_{sv}) - (d_f - d_v) \cos(\alpha_{sv})} \right] \quad \text{Ecuación 2.15}$$

Y las velocidades que se aplicarán a los motores de las llantas delanteras.

$$\dot{\phi}_{sr} = \left[[b \text{sen}(\alpha_{sv}) + (d_v - d_f) \cos(\alpha_{sv})]^2 + (d_f - d_s)^2 \sin(\alpha_{sv})^2 \right]^{\frac{1}{2}} \frac{\dot{\phi}_{sv}}{d_v - d_f} \quad \text{Ecuación 2.16}$$

$$\dot{\phi}_{sl} = \left[[b \text{sen}(\alpha_{sv}) - (d_v - d_f) \cos(\alpha_{sv})]^2 + (d_f - d_s)^2 \sin(\alpha_{sv})^2 \right]^{\frac{1}{2}} \frac{\dot{\phi}_{sv}}{d_v - d_f} \quad \text{Ecuación 2.17}$$

Para estudiar los detalles matemáticos, referirse a Gonzalez-Villela, 2006.

Resumen de parámetros:

Variables

α_{sv}	Ángulo de la llanta virtual.
α_{sr}	Ángulo de la llanta frontal derecha.
α_{sl}	Ángulo de la llanta frontal izquierda.
$\dot{\phi}_{sv}$	Velocidad de la llanta virtual
$\dot{\phi}_{sr}$	Velocidad de la llanta frontal derecha.
$\dot{\phi}_{sl}$	Velocidad de la llanta frontal izquierda.

Constantes

d_v	Distancia del punto P al eje perpendicular al eje de rotación de la llanta virtual.
d_s	Distancia del punto P al eje perpendicular al eje de rotación de las llantas frontales.
d_f	Distancia del punto P al eje perpendicular al eje de rotación de las llantas traseras.
b	La mitad de la distancia entre las llantas traseras.

2.4. Odometría

La **odometría** es el estudio de la estimación de la posición de vehículos con ruedas durante la navegación. Para realizar esta estimación se usa información sobre la rotación de las ruedas para estimar cambios en la posición a lo largo del tiempo. La palabra odometría se compone por las palabras griegas *hodos* ("viajar", "trayecto") y *metron* ("medida"). Este método se basa en la suposición de que las revoluciones de las ruedas pueden ser traducidas en un desplazamiento lineal relativo al suelo.

La localización del robot usando la predicción odométrica es bastante precisa en ausencia de deslizamiento y derrape de la llanta. Estos efectos se reducen cuando la velocidad se mantiene baja y los movimientos son pocos.

Para analizar los aspectos generales de la odometría en los robots móviles, se requiere de localizar un punto y obtener ecuaciones de la cinemática del robot a estudiar.

En el presente estudio se maneja un tipo de robot cuyas ecuaciones resultan muy complejas para el análisis de la reconstrucción odométrica. Sin embargo, es posible hacerse analogía con la del robot diferencial.

El robot móvil diferencial es un chasis con dos llantas traseras instrumentadas que rotan respecto a un eje perpendicular al plano de la Figura 7.

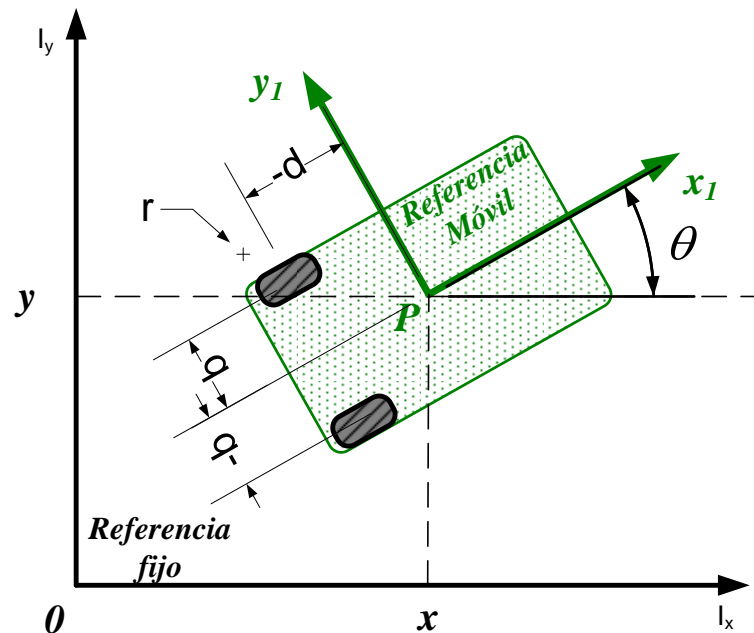


Figura 7. Representación de robot móvil. Modificada de Gonzalez-Villela, 2006

A partir de la cinemática se obtienen las ecuaciones que proporcionan información de su posición en el plano con respecto a una base inercial, ver apartado 2.2.2.

La posición del punto P en el robot está descrita por $\varepsilon = (x, y, \theta)^T$ y su derivada con respecto al tiempo es $\dot{\varepsilon} = (\dot{x}, \dot{y}, \dot{\theta})^T$, cabe mencionar que θ es el ángulo formado por el eje longitudinal del robot y el eje X del sistema de referencia fijo (ya se había comentado previamente), mientras que $\dot{\theta}$ es la velocidad angular del robot.

Para la odometría es necesario disponer de información de las dos llantas traseras, la derecha (r) y la izquierda (l) acerca de sus desplazamientos angulares respectivamente son ϕ_r y ϕ_l , asimismo de sus velocidades angulares $\dot{\phi}_r$ y $\dot{\phi}_l$.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi}_r \\ \dot{\phi}_l \end{bmatrix} = \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \\ 0 & 1 \\ 1/r_w & b/r_w \\ 1/r_w & -b/r_w \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad \text{Ecuación 2.18}$$

v es la velocidad lineal del robot

ω es la velocidad angular, con respecto a la base inercial.

b es la distancia del eje x hacia el eje perpendicular a la llanta.

d es la distancia del eje de las llantas al punto P.

r_w es el radio de ambas llantas

Obteniendo las ecuaciones para cada variable en términos de velocidad lineal y angular.

$$\begin{aligned} \dot{x} &= v \cos \theta - \omega d \sin \theta \\ \dot{y} &= v \sin \theta + \omega d \cos \theta \\ \dot{\theta} &= \omega \\ \dot{\phi}_r &= v/r_w + b\omega/r_w \\ \dot{\phi}_l &= v/r_w - b\omega/r_w \end{aligned} \quad \text{Ecuación 2.19}$$

Si, $v = \dot{x}_1$ y $\omega = \dot{\theta}_1 = \dot{\theta}$, las ecuaciones para cada variable están en términos de incrementos de velocidad lineal $\frac{\Delta x_1}{\Delta t}$ y angular $\frac{\Delta \theta}{\Delta t}$, donde Δt es factor común en ambos lados de las ecuaciones, por lo que se simplifica:

$$\begin{aligned} \Delta x &= \Delta v \cos \theta - \Delta \omega d \sin \theta \\ \Delta y &= \Delta v \sin \theta + \Delta \omega d \cos \theta \\ \Delta \theta &= \Delta \omega \\ \Delta \phi_r &= \Delta v / r_w + b \Delta \omega / r_w \\ \Delta \phi_l &= \Delta v / r_w - b \Delta \omega / r_w \end{aligned} \quad \text{Ecuación 2.20}$$

Resolviendo el sistema de ecuaciones $\Delta\phi_r = \Delta x_1 / r_w + b\Delta\theta / r_w$ y $\Delta\phi_l = \Delta x_1 / r_w - b\Delta\theta / r_w$ conociendo los incrementos angulares provenientes de los *encoders* de cada llanta, se tiene

$$\Delta x_1 = \frac{r_w}{2}(\Delta\phi_r + \Delta\phi_l) \text{ y } \Delta\theta = \frac{r_w}{2b}(\Delta\phi_r - \Delta\phi_l).$$

Los incrementos Δx_1 y $\Delta\theta$ son sustituidos en las dos primeras restricciones de ecuación 2.20, para obtener Δx y Δy . La forma en que se determinan las coordenadas $\varepsilon = (x, y, \theta)^T$ es de tipo iterativo pues $\Delta x = x_i - x_{i-1}$ y $\Delta y = y_i - y_{i-1}$, tal que:

$$x_i = \Delta x_1 \cos \theta - \Delta\theta d \sin \theta + x_{i-1} \quad \text{Ecuación 2.21}$$

$$y_i = \Delta x_1 \sin \theta + \Delta\theta d \cos \theta + y_{i-1} \quad \text{Ecuación 2.22}$$

Asimismo:

$$\theta_i = \Delta\theta + \theta_{i-1} \quad \text{Ecuación 2.23}$$

3. Requerimientos para el prototipo

3.1. Actuadores y sensores

3.1.1. Motores LEGO NXT®

Es una pieza prefabricada que posee una serie de engranes para aumentar la fuerza angular o Par (*Torque*) además de que este tipo de servomotor posee un *Encoder* de Cuadratura para proporcionar información de la posición angular al subsistema correspondiente. Se tiene un engrane con 12 ventanas acoplada directamente al motor de corriente directa, teniendo una relación de 10:32 con respecto al motor. Posteriormente se une a una reducción de engranes con una relación de 1:48 en total hasta el eje de salida. De tal forma que por una vuelta de dicho eje de salida, el *encoder* realiza $48 \cdot 10 / 32 = 15$ vueltas, y cada detector óptico (*TACHO 0 y 1*) percibe $15 \cdot 12 = 180$ pulsos. En el programa contenido en el ladrillo inteligente NXT (que se describirá más adelante), procesa la información para obtener los 360° con +/- un grado de error. En la Figura 8 se observa su estructura interna.

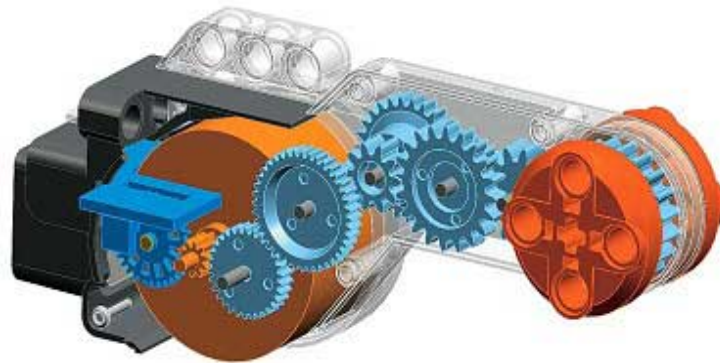


Figura 8. Dibujo de un motor NXT®, tomado de *NXT® motor internals, 2009*

3.1.1.1. Organización interna

Las siguientes fotos muestran un motor de este tipo desarmado, sin embargo, debe recordarse que al remover las cubiertas, estos motores dejan de ser funcionales.

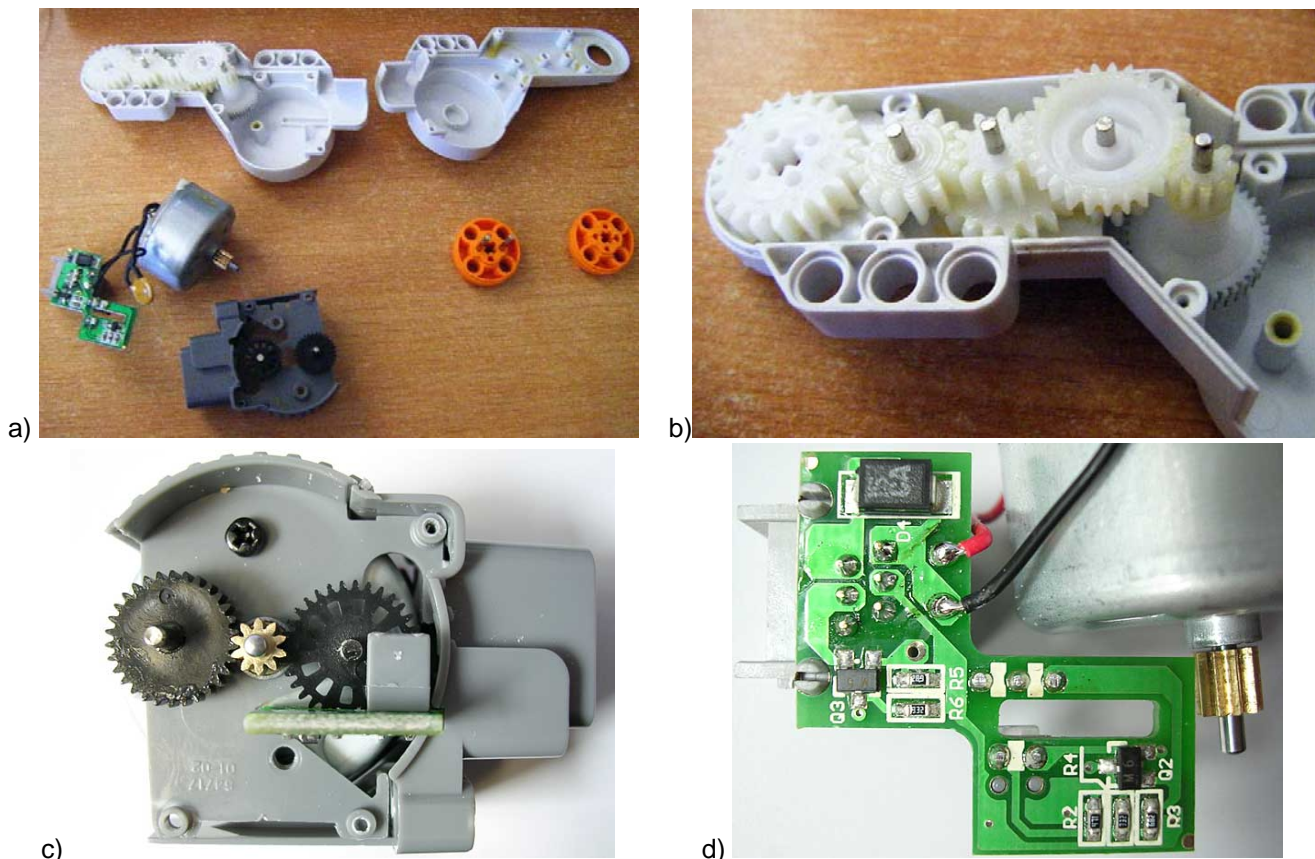


Figura 9. Motor NXT. a) Vista general de las piezas, b) Reducción de engranes (ver Tabla 2), c) Ventana ranurada del *encoder* de cuadratura, d) Circuito impreso de acondicionamiento de señales de entrada y salida. Figuras tomadas de *NXT® motor internals, 2009*

Engrane	Numero de dientes	reducción
1	10:30:40	= 1:4
2	9:27	= 1:3
3	10:20	= 1:2
4	10:13:20	= 1:2
	Total	1:48

Tabla 2. Detalle de engranes

A este motor se le puede energizar hasta con 12 volts, pero sólo se recomienda hasta 9 volts, la corriente sin carga con este voltaje es de 60 mA y su velocidad es de 170 RPM.

Voltaje [V]	Corriente [A]	Velocidad angular [RPM]	Torque o par [Nm]	Potencia mecánica* [W]	Potencia eléctrica** [W]	Eficiencia %
4.5	0.6	33	0.167	0.58	2.7	21.4
7	0.55	82	0.167	1.44	3.85	37.3
9	0.55	117	0.167	2.03	4.95	41
12	0.58	177	0.167	3.10	6.96	44.5

Tabla 3. Características eléctricas y mecánicas

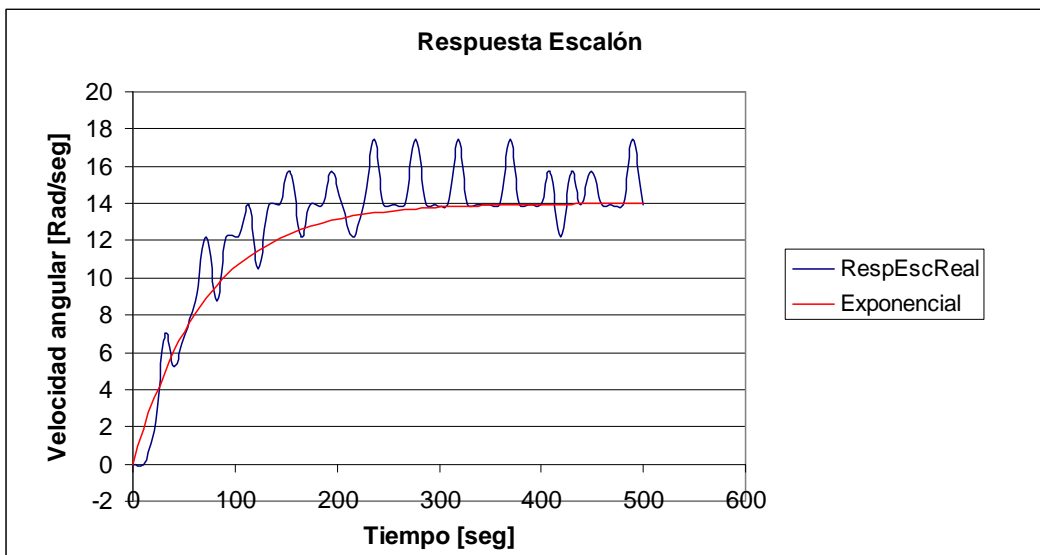
* Potencia mecánica [W] = Velocidad angular [rad/seg] * Par [Nm]
 ** Potencia eléctrica [W] = Voltaje [V] * Corriente [A]

Cuando el rotor está inmobilizado su torque es de 50 N.m y su corriente es de 2 A. La Tabla 3 proporciona información adicional de este motor [NXT® motor internals,2009].

3.1.1.2. Modelo matemático

El modelo que se determina con el análisis descrito para el motor NXT en el Apéndice A, se requiere para hacer aproximaciones a las constantes para realizar el controlador de posición, más adelante en el capítulo 4.

$$G(s) = \frac{\theta(s)}{V_{in}(s)} = \frac{K_1}{s + K_2} = \frac{24.4388512}{s + 14.0024644}$$



Gráfica 1. Valores obtenidos con LabVIEW® para determinar un modelo aproximado del motor NXT®

Nótese de la Gráfica 1 que los valores reales oscilan y la “Exponencial” es una aproximación de donde se tomaron los datos.

3.2. Procesador / Controlador /Vínculo de comunicación

3.2.1. Ladrillo Inteligente LEGO NXT®

Este dispositivo electrónico posee capacidades de procesamiento digital comunicación alámbrica e inalámbrica, indicadores visuales y auditivos, posee además puertos de entradas y salidas. Es la segunda generación de herramientas para la enseñanza/aprendizaje provistas por LEGO® en su línea de juguetes didácticos MINDSTORMS®.

Comenzando con el procesador, el ladrillo NXT de LEGO® *Mindstorms* (*Intelligent Brick from Mindstorms LEGO®*) es un microprocesador Atmel® ARM7 de 32 Bits que ejecuta programas a 48 Mhz y cuenta con una memoria no-volátil de 256 kilobytes y 64 kilobytes de RAM. Lo que permite conservar múltiples archivos y al sistema operativo, aún cuando no tenga baterías.

Cabe mencionar en este rubro que el NXT® tiene un procesador adicional de 8 bits con 4 kilobytes de memoria no-volátil y 512 bytes de RAM para controlar a los servomotores de manera independiente.

Posee un display de cristal líquido (*Liquid Crystal Display*, LCD) con 100 pixeles de largo y 64 pixeles de alto, que se puede manipular cada pixel entre prendido y apagado. El ciclo de cambio es de 17 milisegundos.

Los puertos para motores enlistados con las letras A, B, C se describirán más adelante. Combinan entradas con salidas (Motor con *encoder*).

Los puertos de entrada son enumerados del 1 al 4, pueden conectarse a sensores ultrasónicos, de contacto, de luz, de color y acelerómetros, micrófonos, giroscopios, entre los más comunes.

El puerto de comunicación USB 2.0 con una velocidad de transmisión de hasta 480 Mbps puede emplearse para la programación desde la computadora o bien, enviar comandos directos desde esta última que controlen al robot, sin embargo, en la práctica, el puerto de comunicación inalámbrico *Bluetooth®* Clase II V2.0 podría ser más útil en aplicaciones donde el dispositivo ya armado debe estar en movimiento, su velocidad es de 3 Mbps como máximo, mientras que su alcance es de 10 a 100 metros, dependiendo del fabricante. En la Figura 10 se ve al ladrillo NXT. La comunicación *Bluetooth®* es el vínculo entre la computadora personal y el robot.



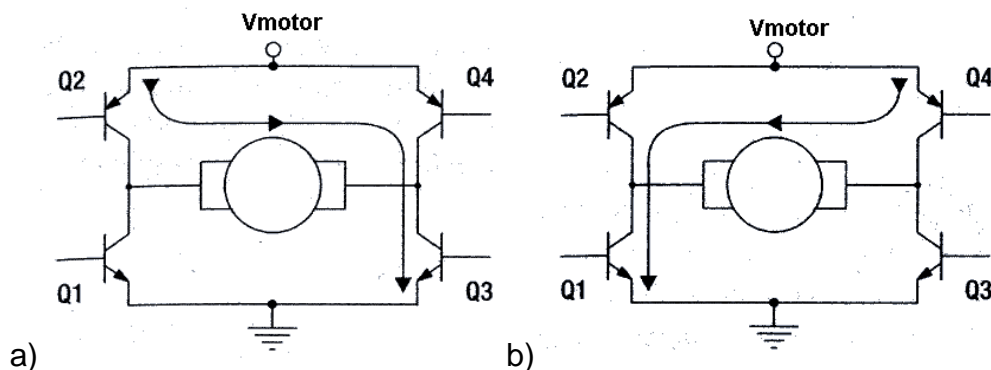
Figura 10. El ladrillo inteligente NXT de LEGO® MINDSTORMS

A manera de referencia, es útil mencionar que el conector los puertos de salida tienen una distribución que se describe brevemente a continuación. En la **Tabla 4** están los pines y colores para identificar cada función.

Pin	Color	Nombre	Pin	Color	Nombre
1	Blanco	Motor 1	4	Verde	4.3 Volts
2	Negro	Motor 2	5	Amarillo	TACHO0
3	Rojo	Tierra	6	Azul	TACHO1

Tabla 4. Detalle de pines de puertos para motores NXT®

Los pines 1 y 2 proveen al motor de señales necesarias para moverse provenientes del ladrillo inteligente NXT, el cual posee un circuito conocido como Puente H (en particular es el LB1836M y LB1930M), formado por cuatro transistores, en la Figura 11 se enumeran del 1 al 4 y el circuito de control está diseñado de tal forma que los transistores 1 y 2 por un lado y 3 y 4 por otro, nunca estén polarizados simultáneamente.



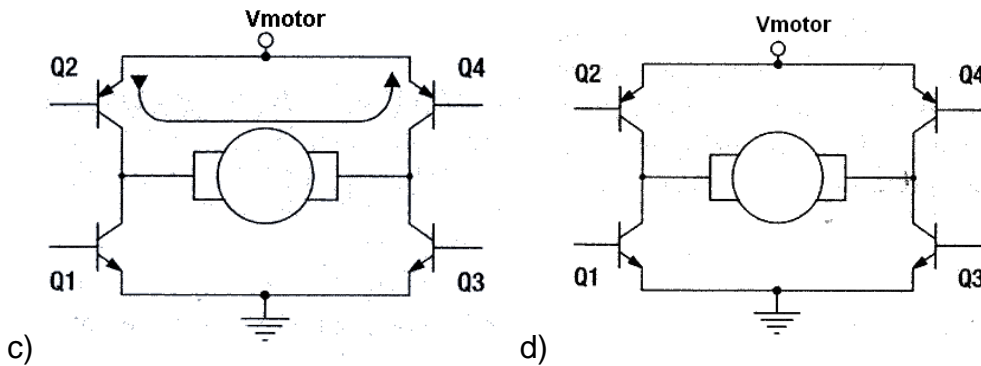


Figura 11. Configuraciones para el movimiento del motor. a) Giro horario b) Giro antihorario c) Freno d) Sin señal. Figuras tomadas de Gasperi, 2007

Dentro del ladrillo inteligente de NXT® se generan PWM para controlar a los motores con una frecuencia de 7.8 Hz, aproximadamente 128.2 μs de periodo, lo cual confiere linealidad en la relación entre el voltaje aplicado y la velocidad.

La velocidad del motor es regulada mediante la modulación de ancho de pulso (*Pulse width modulation, PWM*). Éste es una señal o fuente de poder implica la modulación de su ciclo de trabajo (*duty cycle*), ya sea para transmitir información sobre un canal de comunicaciones o para el control la cantidad de poder que se envía a una carga. El PWM usa una onda cuadrada que cambia su valor promedio de la forma-de-onda. Si se considera una forma-de-onda cuadrada $f(t)$ con un valor bajo de y_{min} , una valor alto y_{max} y un ciclo de trabajo D (ver Figura 12), el valor promedio de la forma-de-onda está dado por:

$$\bar{y} = \frac{1}{T} \int_0^T f(t) dt$$

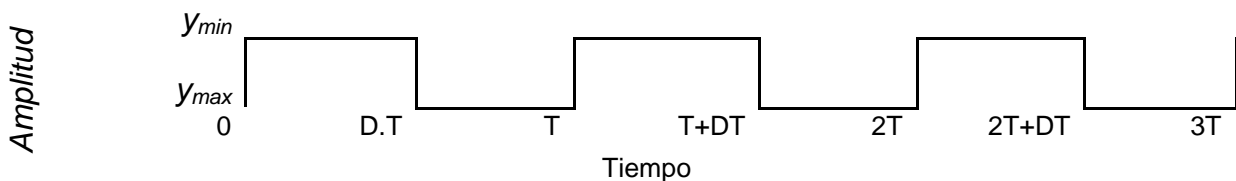


Figura 12. Una onda cuadrada, mostrando las definiciones de y_{min} , y_{max} , T y D

Como $f(t)$ es una onda cuadrada, sus valores son:

$$y_{max} \text{ para } 0 < t < D \cdot T$$

$$y_{min} \text{ para } D \cdot T < t < T$$

La expresión anterior se convierte entonces en:

$$\bar{y} = \frac{1}{T} \left(\int_0^{DT} y_{\max} dt + \int_{DT}^T y_{\min} dt \right) = \frac{D \cdot T \cdot y_{\max} + T \cdot (1-D) \cdot y_{\min}}{T} = D \cdot y_{\max} + (1-D) \cdot y_{\min}$$

Esta última expresión se puede simplificar en $\bar{y} = D \cdot y_{\max}$ debido a que $y_{\min} = 0$, razón por la cual el valor promedio de la entrada de voltaje al motor, es directamente dependiente del ciclo de trabajo.

Continuando con los pines descritos en la **Tabla 4** el número tres de color rojo es Tierra, mientras que el pin 4 de color verde es la diferencia de potencial con 4.3 Volts que proporcionan energía a todos los sensores. Sobre los pines 5 y 6, (colores amarillo y azul) son entradas para el *encoder de cuadratura*: TACHO0 Y TACHO1 que en la Figura 13 se ha ilustrado el comportamiento según el sentido de giro del motor.

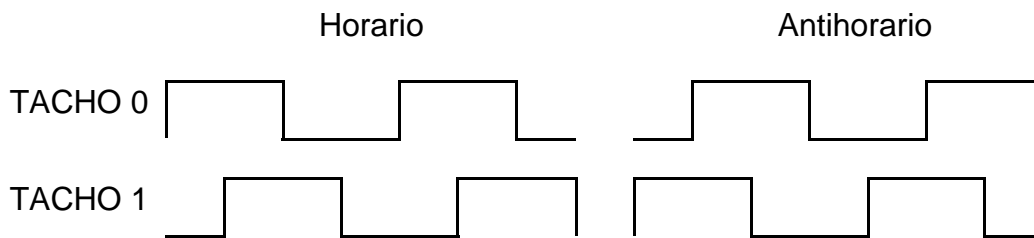


Figura 13. Señales del *encoder* de cuadratura

3.3. Software

3.3.1. LabVIEW®

Es un entorno de programación gráfica, creado por *National Instruments* (NI) y usado para desarrollar sistemas de medida, pruebas y control usando íconos gráficos e intuitivos y cables que parecen un diagrama de flujo. LabVIEW® ofrece la integración con múltiples dispositivos de diferentes fabricantes y brinda herramientas compatibles con ellos como bibliotecas integradas para análisis avanzado y visualización de datos. La plataforma LabVIEW® es escalable a través de múltiples objetivos y sistemas operativos, desde su introducción en 1986.

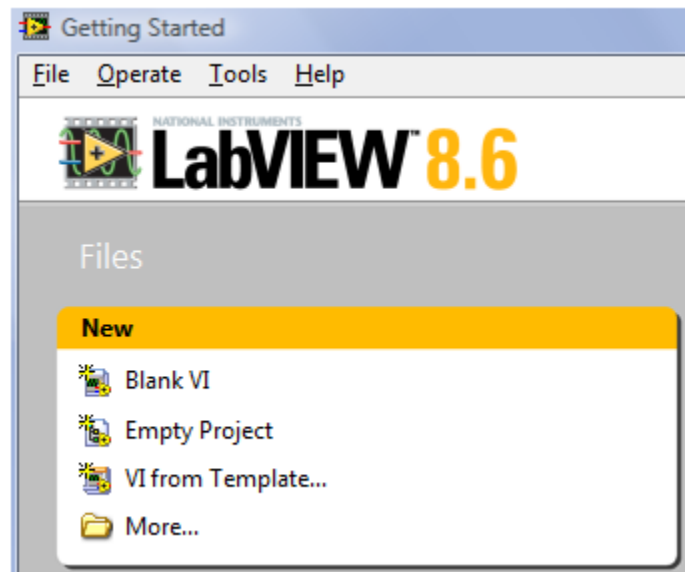


Figura 14. Muestra de LabVIEW® y su versión

Con LabVIEW®, se puede adquirir señales del mundo real, realizar análisis para obtener datos significativos y comunicar o almacenar resultados. Además es posible comunicarse con el exterior, mediante dispositivos tales como módulos de salida o dispositivos tales como el NXT de LEGO®. La versión que se empleó en este proyecto es LabVIEW® 8.6.

Las herramientas usadas de este sistema fueron estructuras de programación como *DoWhile* y *Formula node*, arreglos numéricos, controles, indicadores, gráficas, operadores aritméticos; creación de objetos como teclados, archivos txt. En la Figura 15 se observan algunas de las herramientas.

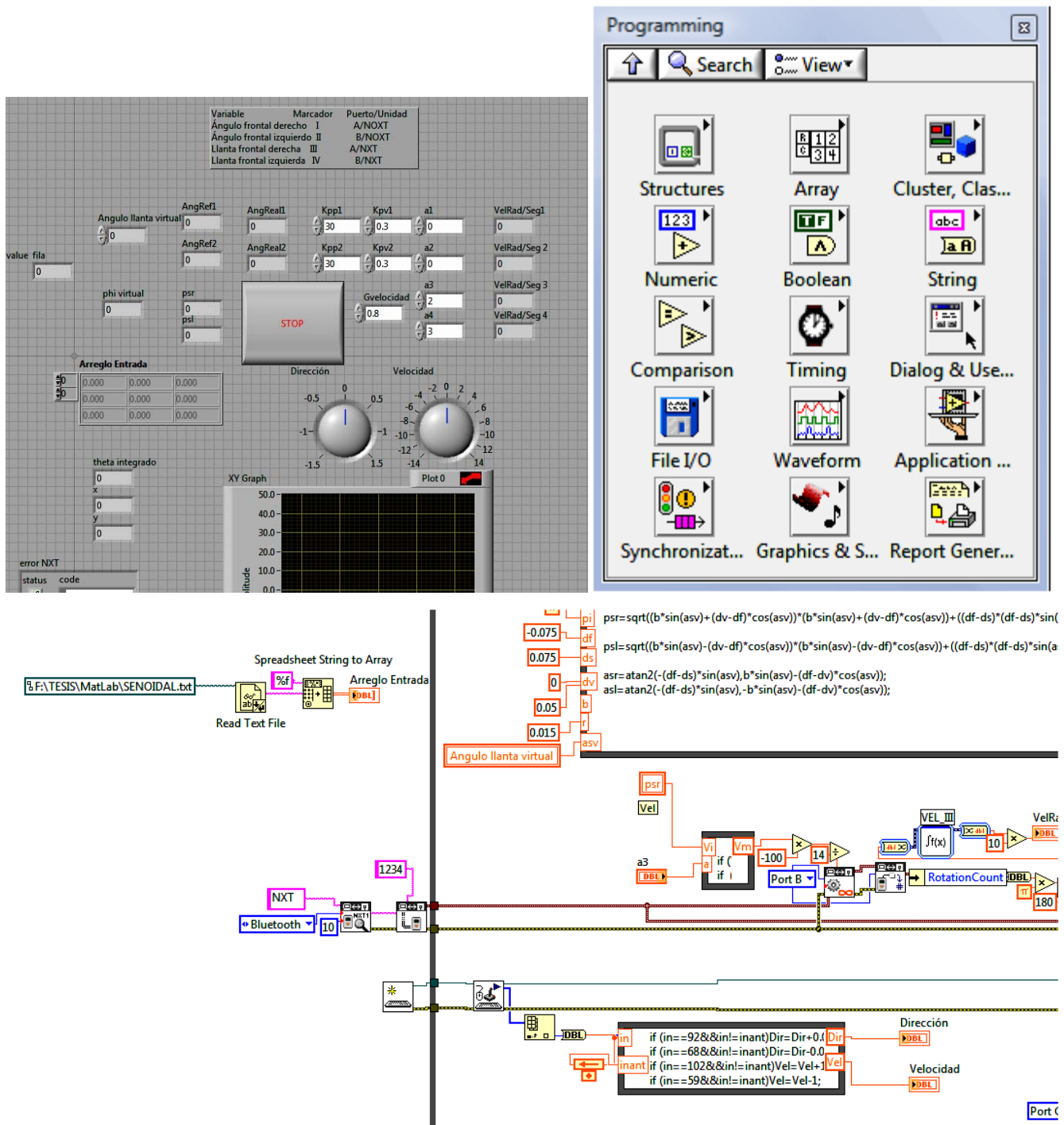


Figura 15. Ejemplos de las herramientas usadas en LabVIEW®

3.3.1.1. Juegos de herramientas (toolkit) para NXT®

El software de LEGO® MINDSTORMS® NXT® emplea un lenguaje de programación gráfico llamado NXT-G® que está basado en LabVIEW® 7.1 de National Instruments. La

programación alto nivel diseñado en bloques que están constituidos por diagramas simples de LabVIEW®, de tal forma que es posible crear bloques personalizados e incluso enviar instrucciones directamente desde un panel de control, haciendo posibles cosas que con el programa NXT-G®, simplemente estaría limitado.

Recientemente las versiones de LabVIEW® sólo requieren de “addons” para hacer funcionar al NXT, en la figura se ilustran los juegos de herramientas (*toolkit*). Habrá que tomar en consideración que las variables y estructuras de programación que el NXT acepta de las herramientas de LabVIEW® si se desean compilar y descargar programas que se ejecutarán por el NXT. Sin embargo el envío de comandos directos a través de una conexión (alámbrica o inalámbrica) aumenta la complejidad del programa que puede ejecutar el prototipo.

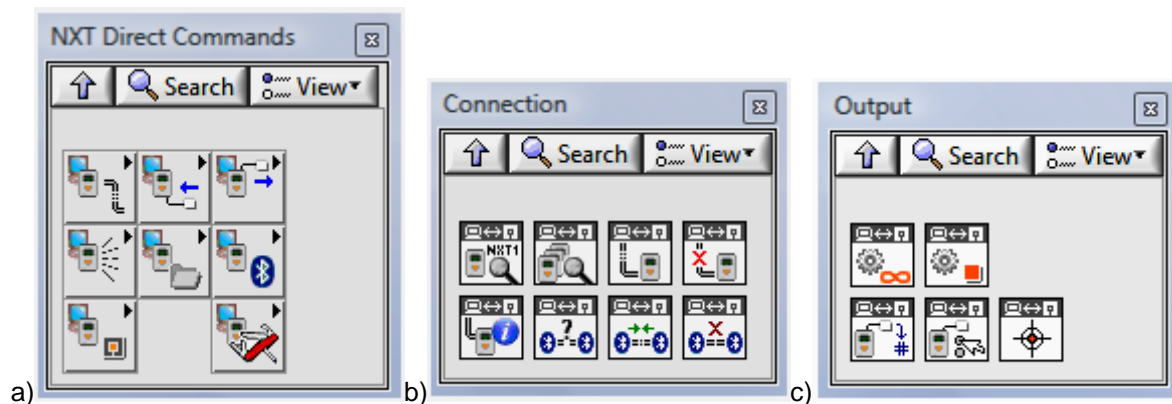


Figura 16. Controles directos desde LabVIEW® hacia el NXT. a) Panel general, b) Panel de conexión USB o bluetooth®, c) Panel de Salidas, control de los puertos A, B y C de salida.

3.3.2. Matlab / Simulink®

MATLAB® es un lenguaje de alto nivel con un ambiente interactivo que permite realizar tareas computacionalmente intensivas mucho más rápido que con lenguajes de programación tradicionales tales como C, C++, o Fortran. La versión empleada es la 2009a.

Simulink® Es una plataforma para simulación y diseño basado en modelos de sistemas dinámicos y embebidos. Proporciona un entorno gráfico interactivo y un conjunto de librerías de bloques personalizables que permiten diseñar, simular, implementar y probar una gran variedad de sistemas con variación temporal, entre los que se incluyen sistemas de comunicaciones, control, procesamiento de señales, vídeo e imágenes.

Para el presente trabajo se ha empleado herramientas elaboradas por Víctor Javier González Vilella donde un robot con las características del prototipo de este estudio sigue trayectorias formuladas en términos de x y y , pero calcula los parámetros que deben seguir los motores, tanto los ángulos como las velocidades.

En la Figura 17 se muestra una trayectoria generada y seguida por un robot programados en Simulink®.

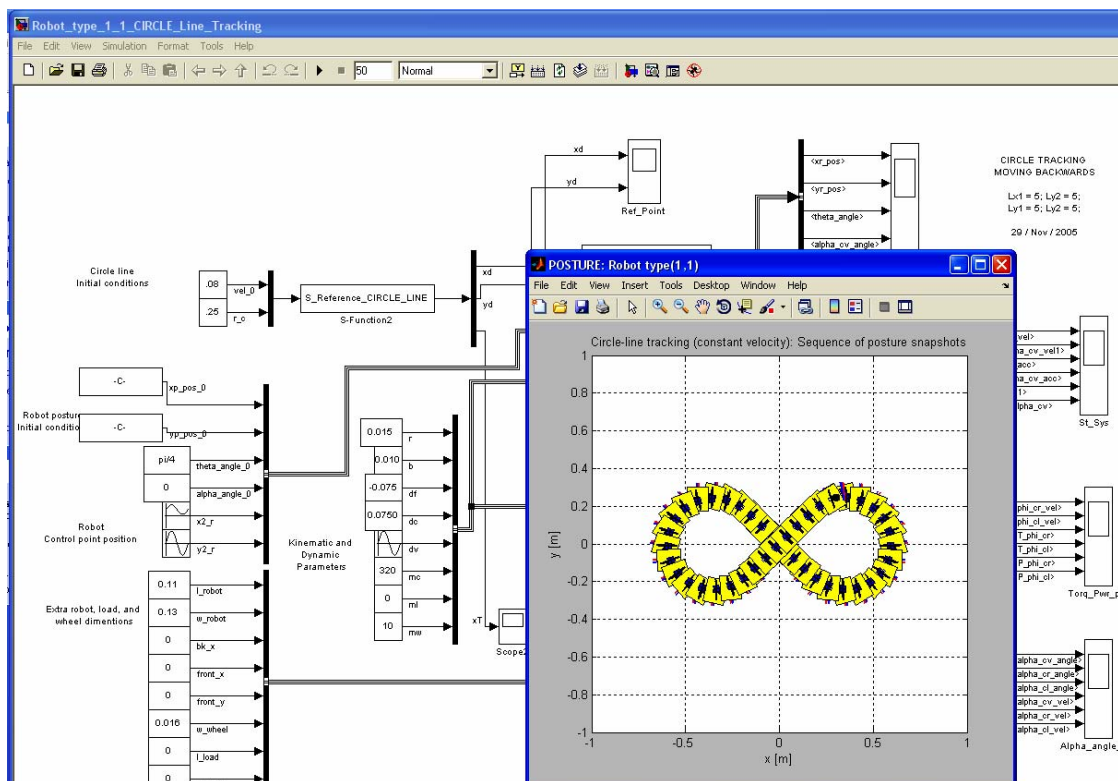


Figura 17. Muestra de la programación en Simulink® donde un robot sigue una trayectoria

4. Sobre la arquitectura del prototipo

4.1. Arquitectura del robot

Antes de comenzar ver la experimentación, es necesario tener una visión general de los componentes para realizar los experimentos para manejar un robot móvil. En la Tabla 5 se muestran con más detalle los componentes del sistema.

ROBOT MÓVIL		
Subsistema	Materiales	
Procesador Principal	Computadora personal	Programas de uso general, MatLab®, Simulink®, LabVIEW®, Juego de herramientas NXT® para LabVIEW®
Programación	LabVIEW®	Procesamiento de las señales de ángulo, Odometría, Algoritmos de control
Comunicación entre Procesador y Robot	Ladrillo inteligente NXT® MINDSTORMS LEGO®	Provee de comunicación vía <i>bluetooth®</i> (inalámbrico)
Actuación y sensado	Motores NXT®	<i>Encoders</i> y recepción de PWM
Fase de potencia	Ladrillo inteligente NXT®	Transmite señales de potencia al motor, comandadas por LabVIEW®

Tabla 5. Componentes con los que se implementó el sistema.

En la Figura 18 se resume cómo se comunican los sistemas haciendo notar que en la computadora personal se llevarán a cabo casi todos los algoritmos de control, que se verán más adelante.

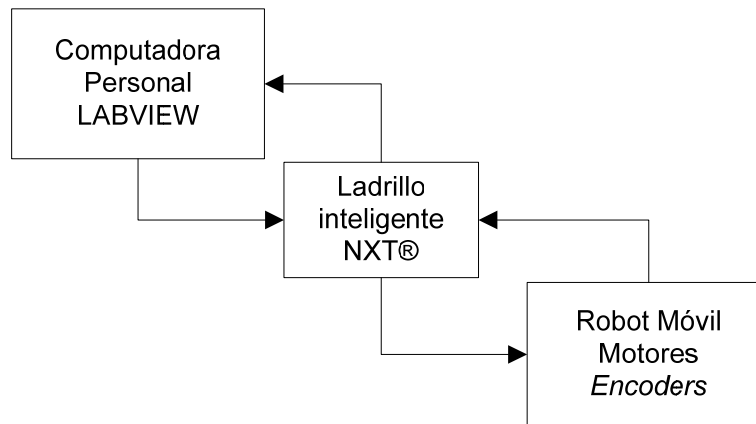


Figura 18. Arquitectura general del sistema robótico.

4.2. Robot tipo carro con dos llantas frontales direccionables independientemente

En la sección 2.3 del presente estudio se ha detallado el análisis cinemático del robot con el que se experimentará. De la Figura 6 se ha tomado la Figura 19 para hacer referencia rápida a los elementos que componen al prototipo.

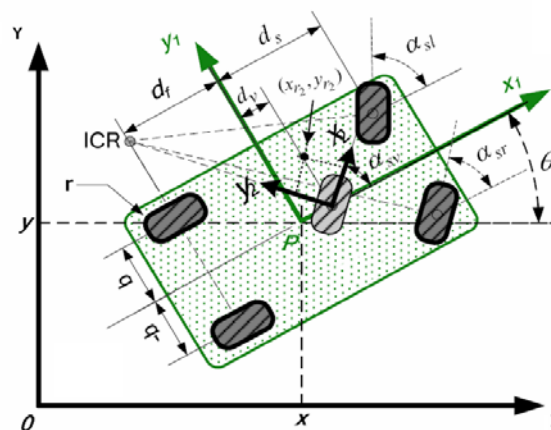


Figura 19. Esquema del Robot tipo carro ubicado en un marco de referencia XY. Tomado de Gonzalez-Villela, 2006

Las llantas frontales están motorizadas. En la Figura 20 está el motor NXT® acoplado a tres engranes con igual número de dientes de forma que la relación de entrada/salida es 24:24:24; en esa misma figura se observa en una línea roja que esquematiza al eje sobre del cuál se orienta la llanta, como ya se ha descrito en el capítulo 2 (Llantas Centradas Orientables).

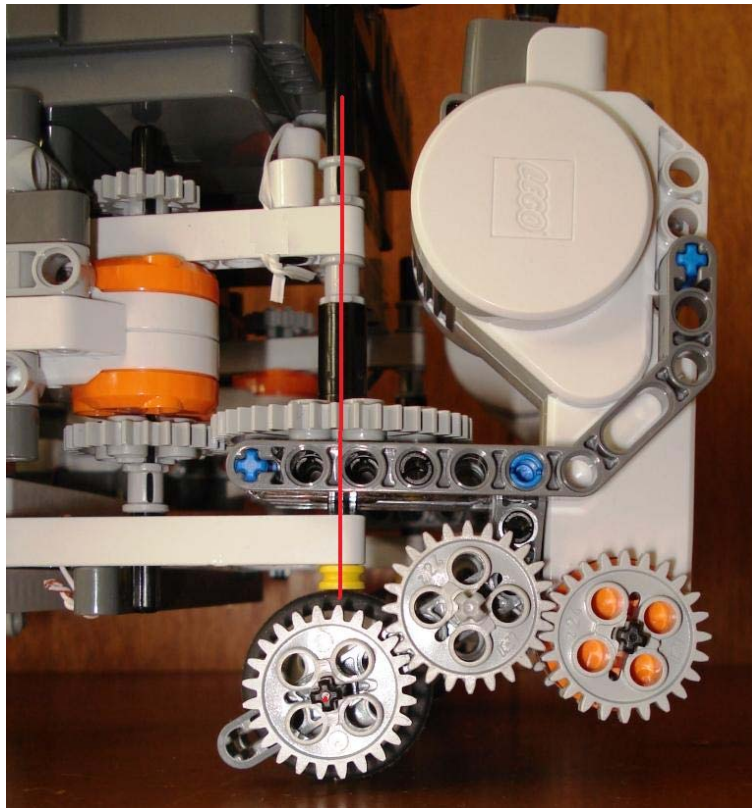


Figura 20. Foto de una llanta frontal

En la Figura 21 se muestra cómo se acopló el instrumento utilizado para medir la rotación de las llantas traseras no motorizadas, en el capítulo 2 se clasificaron como centradas fijas.



Figura 21. Foto de una llanta trasera y su *encoder* de cuadratura (modelo EP4). Para detalles ver *US DIGITAL™*, 2009

En la Figura 22 está el robot visto desde arriba, con las llantas frontales en distintos ángulos de orientación. El punto P se definió en el centro geométrico del robot. Algunas constantes para el estudio se definen más adelante con respecto al punto P.

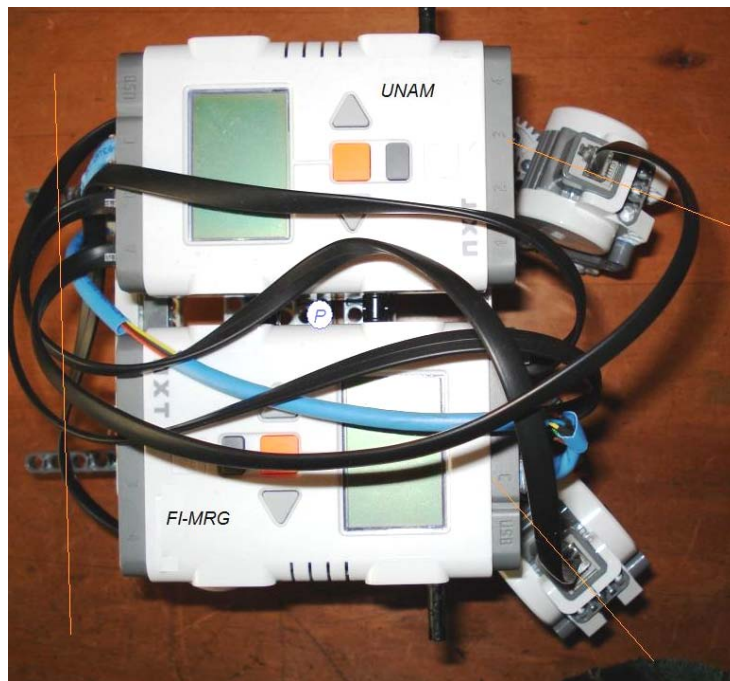


Figura 22. Vista superior del robot. Punto P

En la Figura 23 se ve al robot lateralmente donde se aprecia el motor que proporciona la orientación a las llantas frontales, en este caso hay una reducción de engranes cuya relación de entrada/salida es de 24:40. Son a estos motores que se les debe controlar la posición, ver sección 4.4.

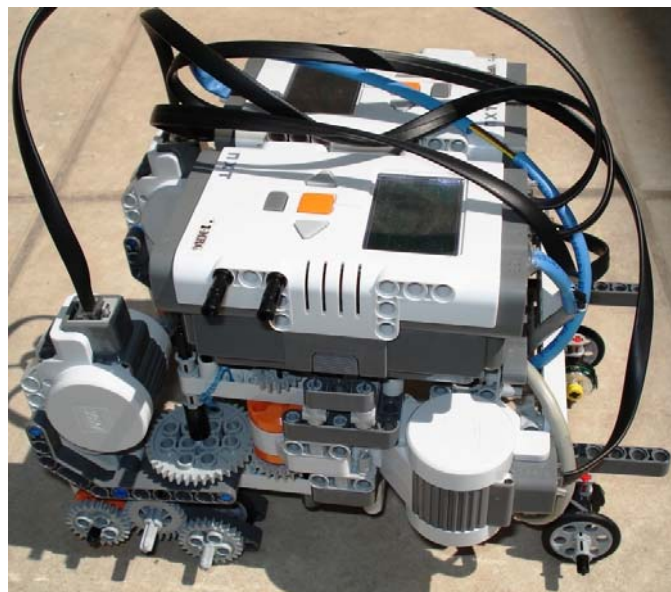


Figura 23. Foto del robot lateral

Las dimensiones finales del prototipo son: $d_v = 0[m]$, $d_s = 0.075[m]$, $d_f = -0.075[m]$, $b = 0.05[m]$.

4.3. El programa

Escrito y diseñado en LabVIEW® 8.6, el programa consta de módulos que controlan la posición de las ángulos frontales, la velocidad de las llantas, el cálculo de la odometría, los cálculos matemáticos que requiere el robot para coordinarse, además de establecer el vínculo con los ladrillos inteligentes NXT®. En Figura 24 y Figura 25 se observan, respectivamente, el panel de control y el diagrama de bloques donde está programado todo el algoritmo

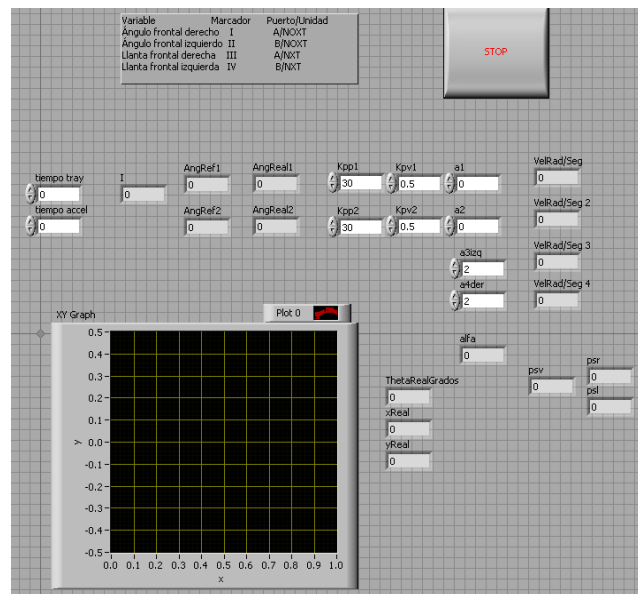


Figura 24. Panel de control principal

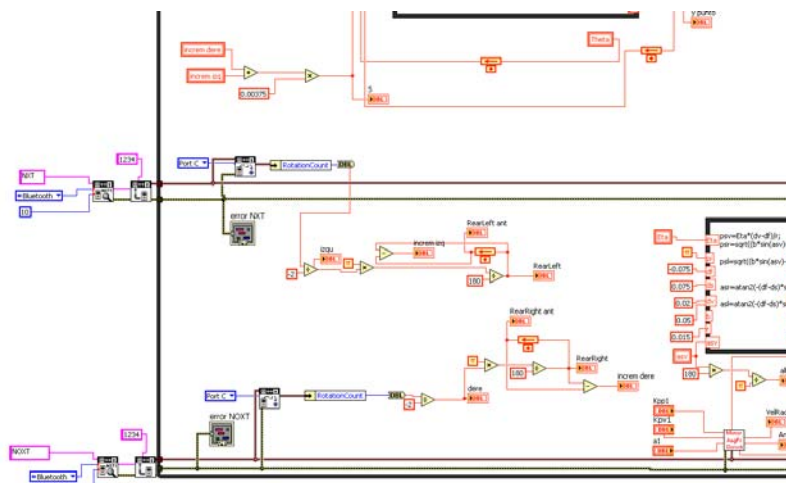


Figura 25. Sección del diagrama de bloques funcionales. Ver apéndice B para los detalles.

El robot es conectado según lo indica en la Tabla 6 donde se observa que se emplean dos NXT® nombrados diferente para que los programas puedan identificarlos. Los puertos de los motores están completamente usados, aunque los últimos dos no son motores, las llantas traseras no están motorizadas, pero si instrumentadas con *encoders* de cuadratura compatibles, de forma que los puede leer y decodificar el ladrillo correspondiente.

Variable	Marcador	Puerto/Unidad
Ángulo frontal derecho	I	A/NOXT
Ángulo frontal izquierdo	II	B/NOXT
Llanta frontal derecha	III	A/NXT
Llanta frontal izquierda	IV	B/NXT
Llanta trasera derecha	V	C/NOXT
Llanta trasera izquierda	VI	C/NXT

Tabla 6. Conexiones en los diferentes ladrillos de LEGO®

4.4. Control de posición

Se ha mencionado que está programado un módulo de control de posición para los ángulos de las llantas frontales, encargados de darles dirección. En la Figura 26 se muestra un esquema de control donde el parámetro de referencia es un ángulo, que es comparado con el ángulo real del motor (obtenido del *encoder* de cuadratura). El ángulo del motor es derivado para obtener la velocidad.

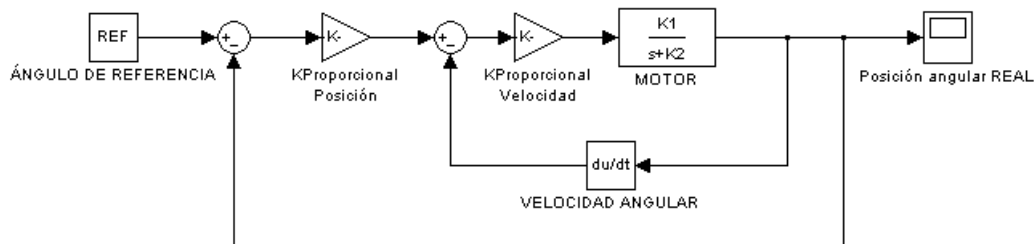


Figura 26. Esquema de control de posición. Adaptado de Groover, 1990.

Este algoritmo es programado el LabVIEW® como se muestra a continuación, las unidades se trabajan en unidades internacionales. El ángulo obtenido del Motor NXT® está en grados, pero se convierte en radianes para ser derivado y cerrar la doble malla de control. La velocidad que llega al motor debe estar en términos de PWM que está en el rango [-100%, 100%], por lo cual se hace un escalamiento, conociendo la velocidad máxima del motor, que corresponde a dicho 100%, en este caso se sabe que es de 14 rad/seg.

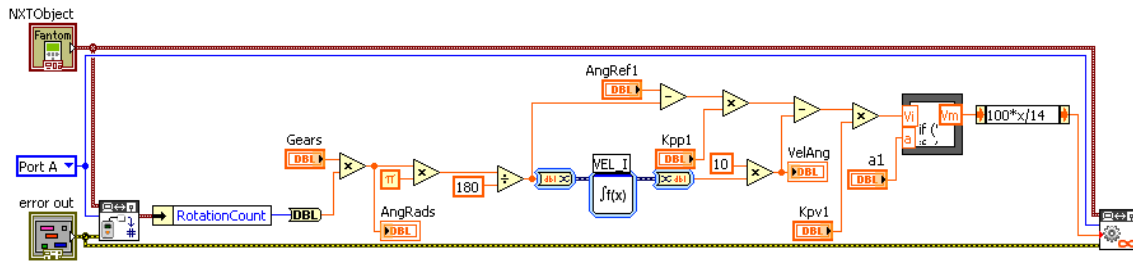


Figura 27. Control de posición para los ángulos frontales derecho e izquierdo.

4.5. Implementación de la Odometría

Las ecuaciones implementadas están en el capítulo 2, pero en la Figura 28 se ven las herramientas empleadas en el programa base.

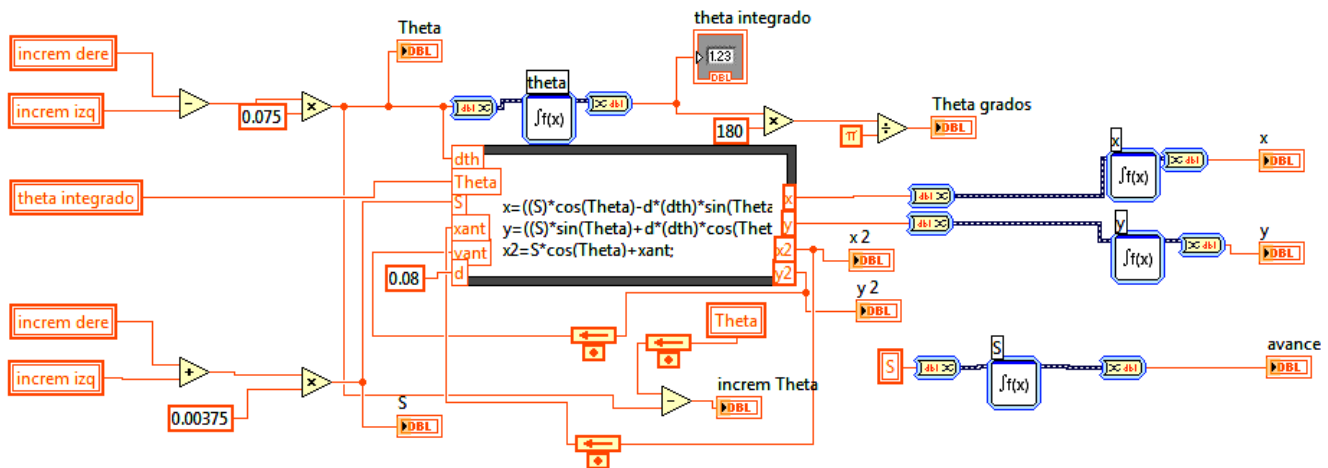


Figura 28. Esquema en LabVIEW® del módulo de odometría

4.6. Cálculo de los parámetros derivados de la **Llanta virtual**

La implementación del control de los experimentos requirió programar las ecuaciones que calculan el ángulo de las llantas frontales a partir del ángulo de la llanta virtual (Ecuaciones 2.14 y 2.15), de tal forma que coincidan todos sus ejes de rotación en un punto conocido como el Centro Instantáneo de Rotación (*Instantaneous Center of rotation, ICR*, ver Figura 19). Así mismo, a través de la velocidad angular de la llanta virtual se calculan las velocidades angulares a las que deben ir las llantas frontales (Ecuaciones 2.16 y 2.17).

Los experimentos realizados, que se verán más adelante, tienen su base en introducir tanto el ángulo y la velocidad angular de la llanta virtual. En la Figura 29 están contenidas las ecuaciones para el cálculo de los ángulos y velocidades de referencia.

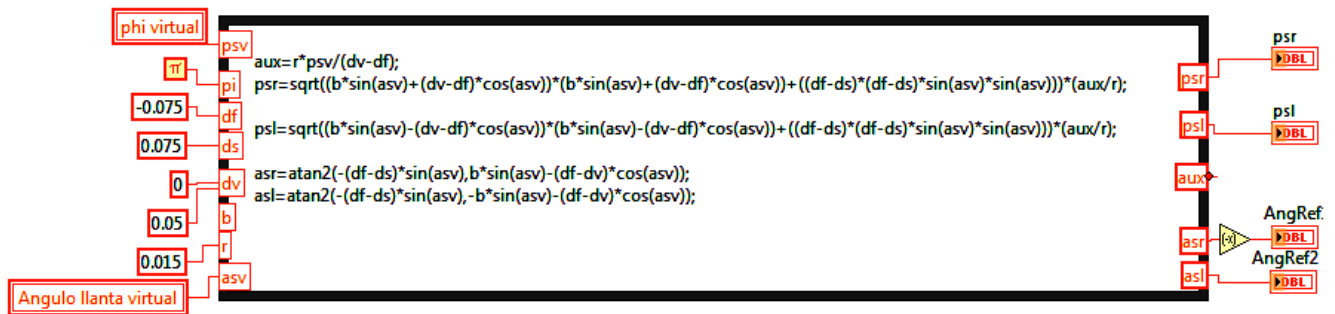


Figura 29. Programación del algoritmo matemático.

en cada experimento muestra los lugares por los que pasó el robot en coordenadas cartesianas, además, un cambio en el ángulo θ del robot con respecto al plano, también es una salida visible en el mundo real.

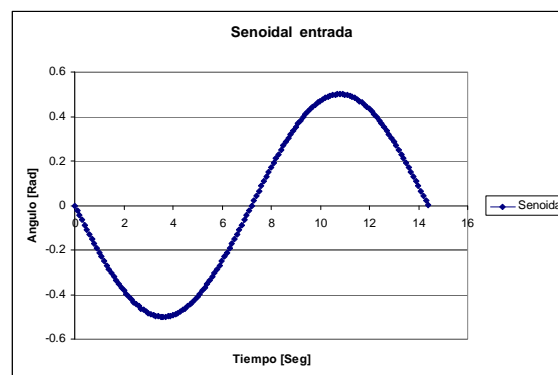
Esta primera interacción con el robot ayudó a comprender el comportamiento de las ecuaciones implementadas, y se comprobó que los centros instantáneos de rotación coinciden al hacer movimientos circulares.

5.1.2. Función arbitraria como entrada

El segundo ensayo consiste en introducir una serie de datos obtenidos de alguna función en términos del tiempo, tales como seno y coseno, entre otros.

5.1.2.1. Variables de entrada

En la Gráfica 2 se muestra una onda senoidal con 0.5 de amplitud, que corresponde a la variación del ángulo de la llanta virtual (α_{sv}), su valor cambia cada 100 milisegundos. El programa en LabVIEW® calcula los valores de ángulo de referencia al que deben llegar los motores que direccionan a las llantas frontales. La velocidad de la llanta virtual ($\dot{\phi}_{sv}$) es de 10 [rad/seg] como constante, pero la velocidad de las llantas cambia de acuerdo al ángulo de orientación.



Gráfica 2. Onda Senoidal con amplitud de $\frac{1}{2}$

En la Figura 31 se muestra una sección del diagrama de bloques donde se introducen los datos de α_{sv} y $\dot{\phi}_{sv}$.

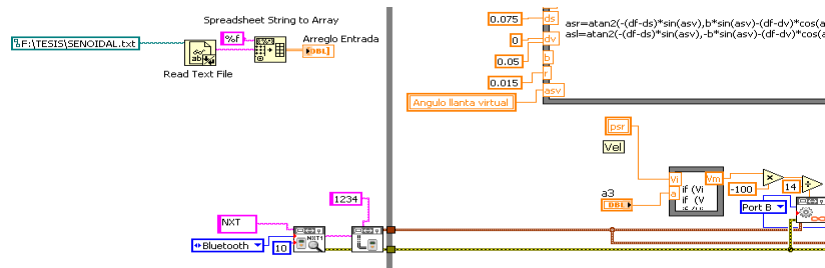
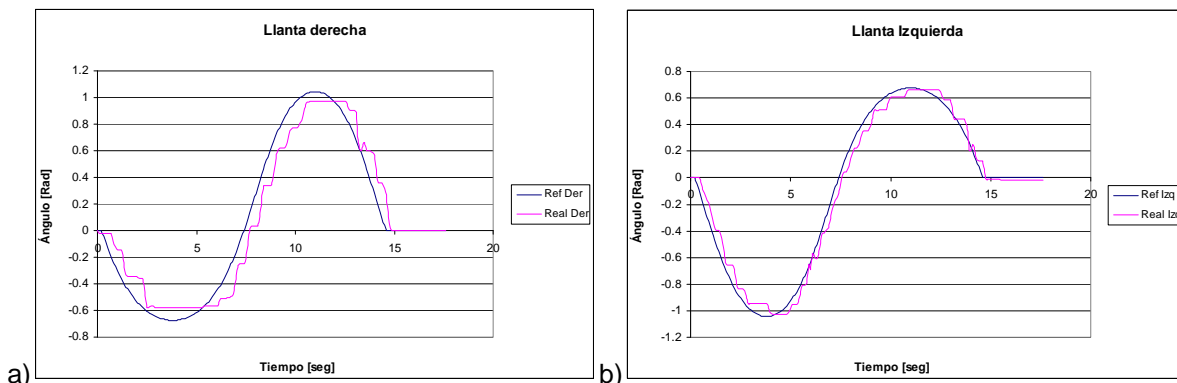


Figura 31. Diagrama de bloques de lectura de datos externos

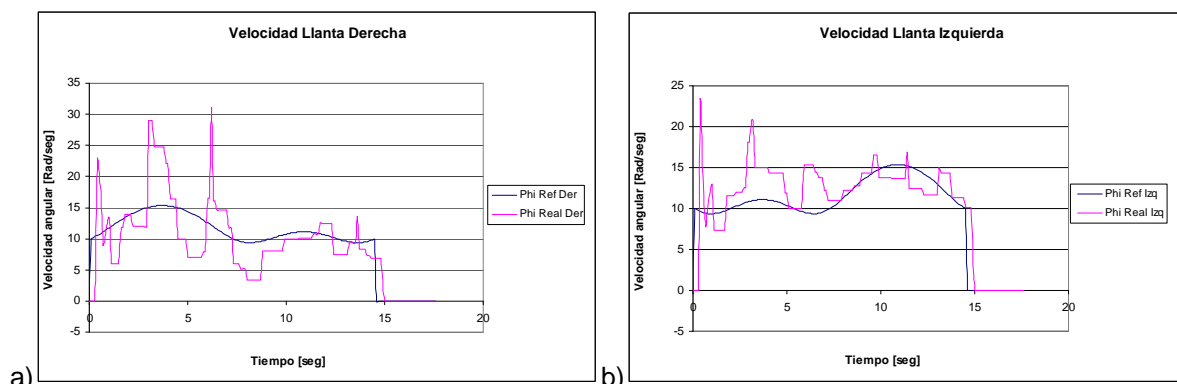
5.1.2.2. Variables de salida

Cada llanta debe tener ángulo y velocidad diferentes durante el recorrido. En las gráficas mostradas a continuación se comparan la referencia con el ángulo o velocidad de cada llanta.



Gráfica 3. Movimiento de la llantas derecha(a) e izquierda (b)

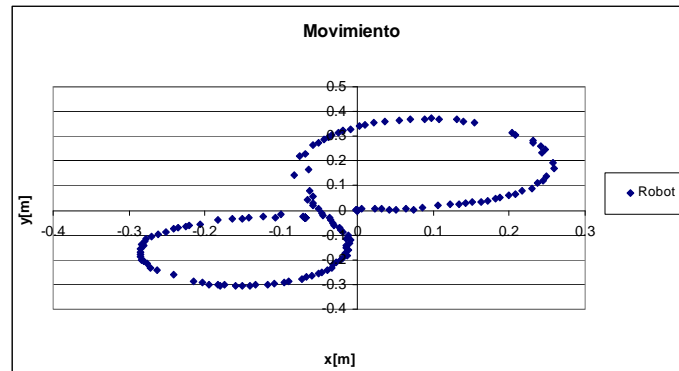
El hallazgo relevante en estas gráficas es que los ángulos se han podido aproximar a los valores de referencia, pero no igualar. Se observan retrasos en el seguimiento de los ángulos.



Gráfica 4. Velocidades de las llantas (a) derecha e (b) izquierda durante el recorrido

Las velocidades angulares mostradas en la Gráfica 4 a y b son la comparación de la velocidad de referencia con la “real”, esta última es calculada mediante la derivada de la

posición angular dentro del programa base. Las variaciones alrededor de los valores de referencia son cambios abruptos imperceptibles en el modelo físico. El hallazgo de estos resultados es que las velocidades reales se encuentran dentro del rango de los de referencia.



Gráfica 5. Recorrido del punto P en el robot. Muestras obtenidas de la odometría (ver secciones 2.4 y 4.5)

El hallazgo a partir de este experimento es que se llega a una posición final en el plano coordinando las llantas (velocidades y ángulos). Se expone que el prototipo puede seguir datos generados externamente.

5.1.3. Obtención de parámetros a partir de SIMULINK®

Para este ejercicio se requirió proponer una trayectoria por la cual el robot debería recorrer en un plano XY, para lo cual se crearon ecuaciones para x y y en términos del tiempo como las que se describen a continuación.

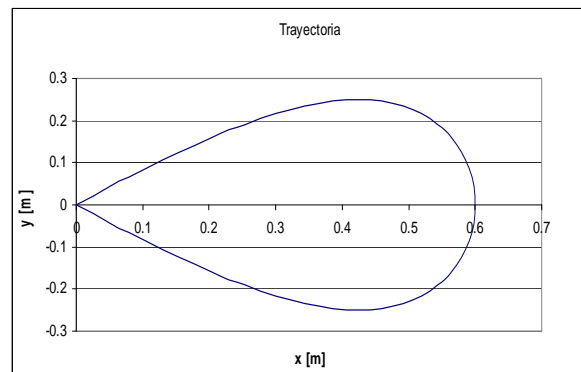
$$x = A_x \text{sen}(\varpi_x t)$$

$$y = A_y \text{sen}(\varpi_y t)$$

Al asignar un valor a las constantes A_x , ϖ_x , A_y y ϖ_y como los siguientes:

$$A_x = 3/5, \varpi_x = 1/4, A_y = 1/4, \varpi_y = 1/2$$

Nótese que una restricción importante es $\varpi_y = 2\varpi_x$ para que se la trayectoria sea como la mostrada en la Gráfica 6.

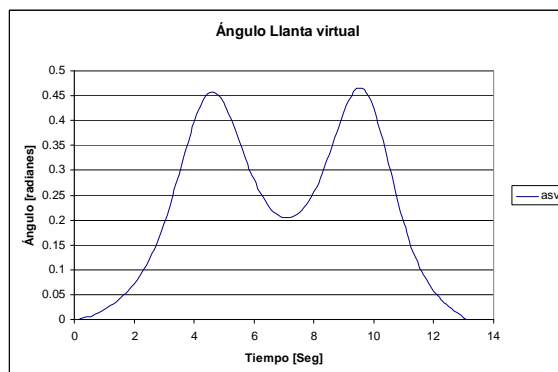


Gráfica 6. Forma de la trayectoria en el plano $x y$.

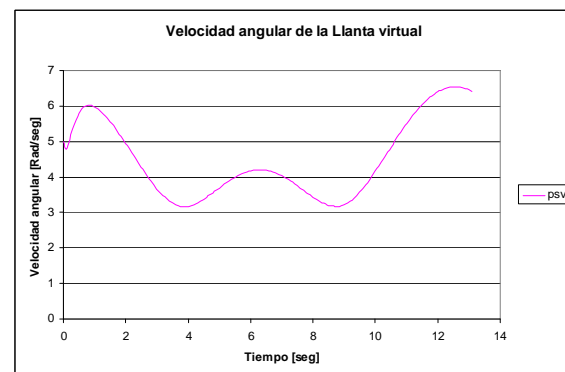
5.1.3.1. Variables de entrada

Las ecuaciones en términos del tiempo para x y y (que describen la trayectoria de la Gráfica 6) son introducidas en un modelo (programado en Simulink®) que calcula una **transformación** hacia el dominio de las aceleraciones del robot móvil. Proceso durante el cual se calculan tanto el ángulo y la velocidad angular de la llanta virtual. Para detalles referirse al trabajo de Gonzalez-Villela, 2006. Con MATLAB® se genera un archivo de texto que contiene estos dos parámetros y el programa base en LabVIEW®8.6 calcula los ángulos y las velocidades de cada llanta.

En la Gráfica 7 y la Gráfica 8 están las entradas para el robot, el ángulo y la velocidad de la llanta virtual, respectivamente.



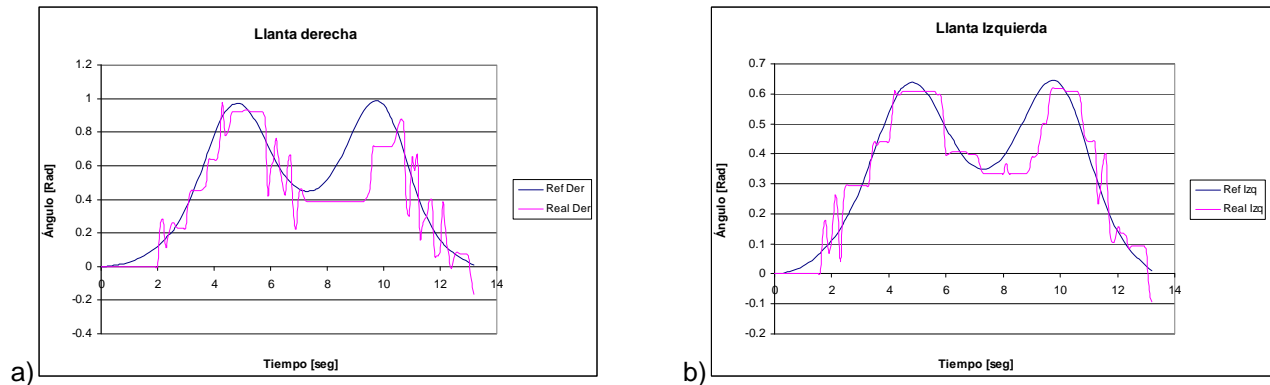
Gráfica 7 Comportamiento del ángulo de la llanta virtual durante la trayectoria propuesta



Gráfica 8. Comportamiento de la velocidad de la llanta virtual durante la trayectoria propuesta

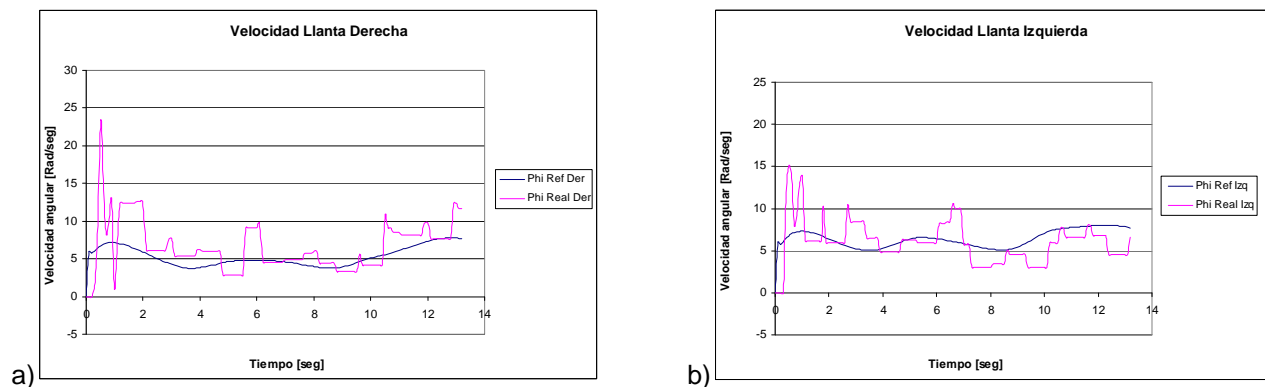
5.1.3.2. Variables de salida

Los ángulos de las llantas en el robot se comparan con los ángulos de referencia calculados a partir de α_{sv} , en las siguientes gráficas se muestran los resultados obtenidos.



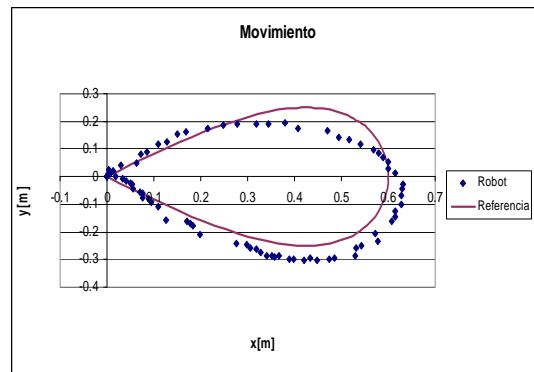
Gráfica 9. Comparación entre los ángulos reales contra los de referencia de la llanta (a)derecha y la (b) izquierda durante la trayectoria propuesta

En Gráfica 9 a y b, los ángulos se aproximan a los valores de referencia, pero con variaciones considerables. Se observan retrasos y oscilaciones en el seguimiento de los ángulos. El controlador de posición es lento para seguir valores de referencia que cambian rápidamente.



Gráfica 10. Comparación entre las velocidades angulares reales contra las de referencia de la llanta (a)derecha y la (b) izquierda durante la trayectoria propuesta

En Gráfica 10 a y b, las velocidades angulares se aproximan a los valores de referencia con variaciones dentro del rango de cada referencia. El prototipo físico no muestra cambios perceptibles ocasionados por estas transiciones.



Gráfica 11. Recorrido del punto P en el robot. Muestras obtenidas de la odometría (ver secciones 2.4 y 4.5)

En la Gráfica 11 se observa que el robot ha seguido la trayectoria de manera aproximada, cabe mencionar que el robot realiza sus operaciones sin el conocimiento de los errores entre su posición real y la de referencia (lazo abierto).

Con el fin de ilustrar los movimientos que tienen las llantas y su coincidencia con el ICR durante la trayectoria, se han colocado líneas imaginarias sobre fotos obtenidas del video donde el robot realiza el movimiento.

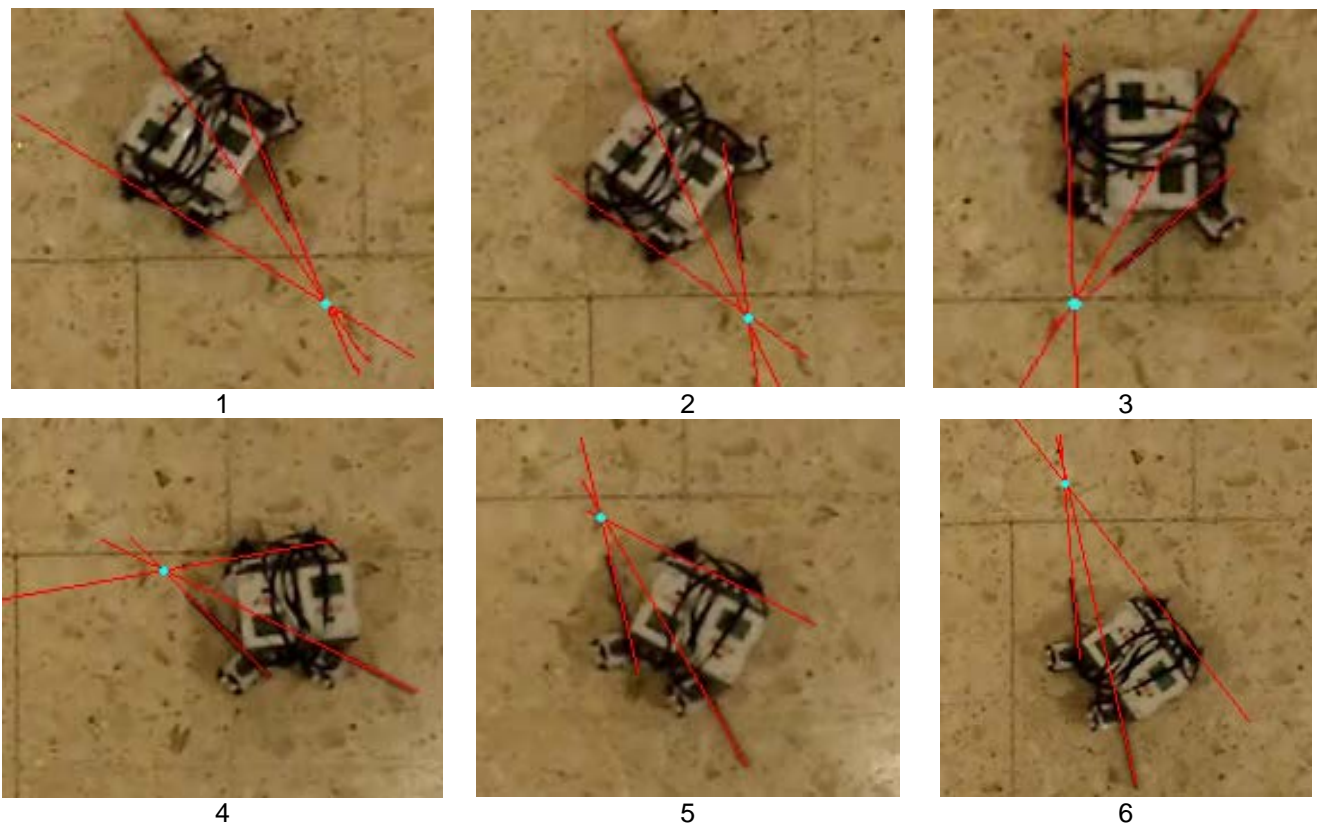


Figura 32. Fotografías tomadas a los movimientos durante el desplazamiento del robot. Las líneas rojas son extensiones de los ejes de rotación de las llantas. El punto azul es el ICR

Para todos los experimentos se encontró que el ángulo de la llanta que está dentro del radio de curvatura es mayor con respecto al externo. Mientras que la velocidad angular de esa llanta es menor a la que está del lado contrario. Dicho de otra forma, cuando una llanta está orientada con un ángulo mayor al contrario con una menor velocidad angular que la opuesta, y viceversa.

Los incrementos angulares entre una iteración y la siguiente llegaron a ser amplios, ocasionando “sobrepasos” que no pudieron ser compensados por el controlador.

La resolución del *encoder* también afectó en el desempeño porque no era menor a 0.5 de grado. Además, el espaciado entre los dientes de los engranes ocasiona un efecto conocido como “*Backslash*”, provocando errores.

A pesar que no se hay la precisión apropiada durante todo el procedimiento, es posible ver que en los resultados se verifican coincidencias en los ICR's de los ejes de las llantas.

También se pueden observar que ha habido errores, pero pueden ser debidos a las fricciones, a la falta de potencia en los motores y a la falta de precisión de los controladores de posición, que el propio sistema en LABVIEW® o LEGO® no permite implementar.

6. Conclusiones

El valor de una llanta virtual es que con ella es posible controlar un punto (x_{r2}, y_{r2}) referenciado al robot móvil. La configuración de robot móvil en estudio (tipo 1,1) requiere tomar una llanta orientable que coordine la dirección del resto de las llantas. Sin embargo, las llantas físicas del robot no se pueden ser reubicadas, al contrario de una llanta virtual, la cual se puede colocar en cualquier lugar del robot e incluso en cualquier punto del plano XY. Ahora bien, este robot al igual que otros, son controlables a partir de un punto (x_{r2}, y_{r2}) , cuya característica necesaria es que no esté ubicado en el origen del sistema de referencia que lo contiene (Sistema (x_2, y_2) , ver **Figura 19**), es decir, el punto está contenido en el sistema de referencia de la llanta, excepto en el origen, de lo contrario las ecuaciones de control cambian y se entra en singularidades. Por lo tanto, una llanta virtual se puede colocar cercana al punto de control y coordinar al resto de las llantas con el fin de mover al robot. Adicionalmente se puede mencionar que se pueden coordinar un número n de llantas físicas con un número m de llantas virtuales.

La experimentación con un robot móvil de llantas frontales independientemente direccionables con un prototipo formado con piezas prefabricadas ha llegado a determinar que es posible usar las herramientas matemáticas, proporcionadas por la cinemática de robots móviles, para controlar el movimiento de dicho robot en un plano. En este caso particular el elemento matemático es la llanta virtual, elemento a partir del cual se han obtenido todos los parámetros necesarios para controlar el posicionamiento de cada llanta y su respectiva velocidad angular.

El hecho de tener únicamente dos variables a controlar por el operador hace muy sencilla la maniobrabilidad y sería posible implementar este control en un automóvil con las llantas independientemente direccionables para dar vueltas a casi cualquier radio de giro. Esto le da un valor de carácter industrial y de impacto social al presente estudio.

El trabajo a futuro es el diseño formal y fabricación de un nuevo modelo, además de un algoritmo que cierre el lazo de control.

Diseño y fabricación de un modelo. Se encontró con la necesidad de hacer un diseño del chasis del robot, además de implementar un control de velocidad y posición para los motores con resoluciones pequeñas, tanto como milésimas de grado. Se deberán adecuar los torques de los motores, dependiendo de la dinámica del robot y la velocidad de respuesta a los algoritmos.

El programa. Más que volver a hacer un programa, lo que hace falta es que el robot cargue consigo las unidades de procesamiento en tiempo real, no sólo para controlar las llantas, sino para realizar algoritmos complejos tales como integración, derivación, manejos de matrices, algunas funciones trigonométricas. Además que dependencia de una computadora sea mínima. En caso de requerir un programa especializado como MATLAB®, sería altamente recomendable implementar tarjetas de tiempo real capaces tanto de seguir los algoritmos como mantener una comunicación constante con el robot, probablemente que la computadora sea parte de la carga del chasis.

Apéndice A

Motores de corriente directa

A.1 Los principios básicos de funcionamiento

Cuando en un campo magnético, una corriente pasa por un conductor, se ejerce una fuerza sobre el conductor (figura A.1). Para un conductor de longitud L que lleva una corriente I en un campo magnético que tiene una densidad de flujo B y es perpendicular al conductor, la fuerza ejercida F es igual a BIL .

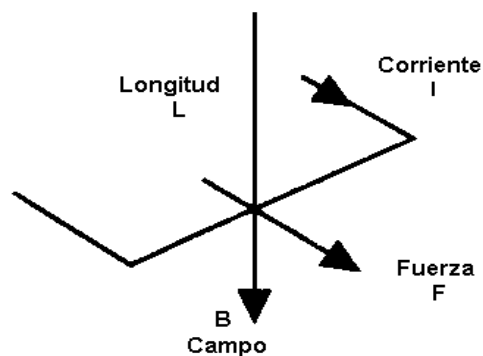


Figura A.1. Fuerza sobre un conductor por el que pasa una corriente. Tomada de Bolton, Mecatrónica, 2001

Cuando un conductor se desplaza dentro de un campo magnético, sobre él se induce una Fuerza Electromotriz (F.E.M.). Véase figura A.2.

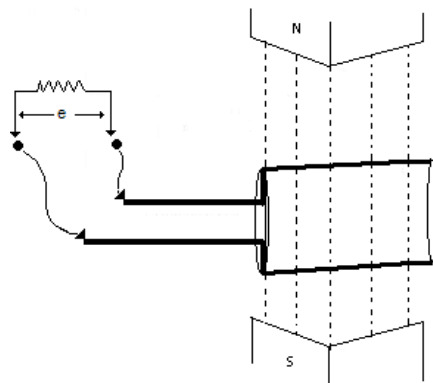


Figura A.2. F.E.M. inducida. Tomada de Bolton, Mecatrónica, 2001

La F.E.M. inducida, e , es igual a la velocidad con la que cambia el flujo magnético Φ (el flujo magnético es igual al producto de la densidad de flujo por el área) que cubre el conductor

(ley de Faraday), es decir, $e = -d\Phi / dt$. El signo menos indica que la dirección de la F.E.M. es en sentido opuesto al cambio que la produce (ley de Lenz); es decir, la dirección de la F.E.M. inducida es tal que produce una corriente que crea campos magnéticos que tienden a neutralizar el cambio en el flujo magnético asociado al devanado que produjo la F.E.M. Por ello, con frecuencia se le conoce como fuerza contraelectromotriz.

En la figura A.3 se observan los elementos básicos de un motor de cd: Una espiral de alambre que gira de manera libre en medio del campo de un imán permanente. Cuando por el devanado pasa una corriente, las fuerzas resultantes ejercidas en sus lados y en ángulo recto al campo provocan fuerzas que actúan a cada lado produciendo una rotación. Sin embargo, para que ésta continúe, cuando el devanado pasa por la posición vertical se debe invertir la dirección de la corriente. En un motor de cd convencional, los devanados de alambre se montan en las ranuras de un cilindro de material magnético conocido como armadura. La armadura está montada en cojinetes y puede girar. Ésta se monta en el campo magnético producido por los polos de campo que pueden ser, en pequeños motores, por ejemplo, imanes permanentes o electroimanes, cuyo magnetismo se obtiene mediante una corriente que circula por los devanados de campo.

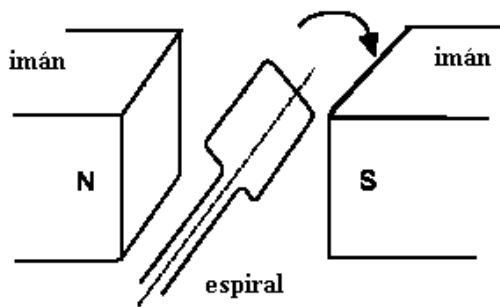


Figura A.3. Elementos básicos, Motor CD

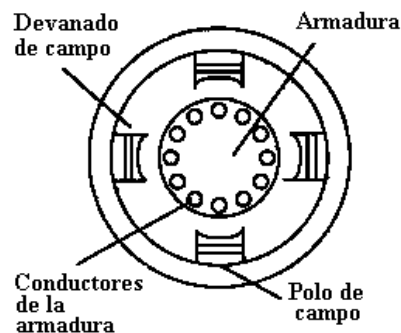


Figura A.4. Corte transversal, Motor CD

Figuras tomadas de Bolton, Mecatrónica, 2001

La figura A.4 muestra el principio básico del funcionamiento de un motor de cd de cuatro polos, cuyo campo magnético se produce por devanados que transportan corriente. Los extremos de los devanados de la armadura se conectan con los segmentos adyacentes de un anillo segmentado conocido como conmutador y el contacto eléctrico con los segmentos se logra mediante contactos de carbón conocidos como escobillas. Conforme la armadura gira, el conmutador invierte la corriente de cada uno de los devanados al desplazarse por los

polos de campo. Esto es necesario a fin de que las fuerzas que actúan en el devanado sigan actuando en la misma dirección y la rotación continúe. La dirección de rotación del motor de cd se invierte al invertir la corriente de armadura o la corriente de campo.

A.2 Modelado de un motor de corriente directa

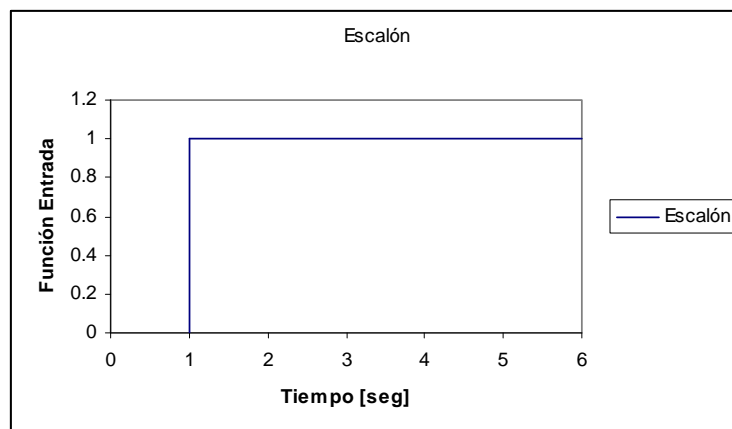
Si se considera al motor como un sistema de primer orden, el modelo básico es:

$$G(s) = \frac{\dot{\theta}(s)}{V_{in}(s)} = \frac{K_1}{s + K_2} \text{ Ecuación A.1}$$

Donde $\dot{\theta}$ es la velocidad angular generada por un voltaje de entrada V_{in} , mientras que las constantes K_1 y K_2 son definidas a partir de una prueba de laboratorio obteniendo los datos de velocidad angular durante a la respuesta a Escalón de magnitud conocida.

A manera de introducción, del modelado se tiene que un Escalón unitario ($U(t)$) es una función cuyo cambio es instantáneo:

$$U(t) = \begin{cases} 0 & \text{si } t < 1 \\ 1 & \text{si } t \geq 1 \end{cases} \text{ Ecuación A.2}$$

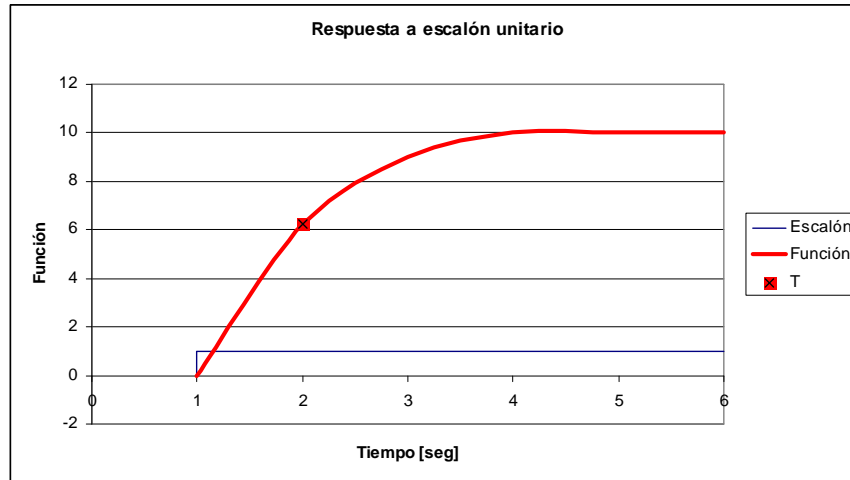


Gráfica A.1. Escalón unitario U(s)

Se sabe que de un sistema de primer orden con modelo sencillo como el que se muestra a continuación, donde $G(s)$ es la función de transferencia, $Y(s)$ es la función de salida y $U(s)$ es la entrada (escalón, en este caso):

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{Ts + 1} \text{ Ecuación A.3}$$

se puede obtener el polo: $P_1 = -\frac{1}{T}$, donde T equivale al tiempo en que se alcanza el 63.2% de $G(s)$ en estado permanente. En la gráfica A.2 se observa que $T = 2\text{seg}$ y $G(s)$ en estado permanente es igual a 10.



Gráfica A.2. Ejemplo de respuesta al escalón

Si experimentalmente se encuentra $G(s)$ en estado permanente ($Y(\infty)$) y se conoce T ; entonces se pueden determinar las constantes K_1 y K_2 de acuerdo a las siguientes ecuaciones:

$$Y(\infty) = \frac{K_1}{K_2} * M \quad \text{Ecuación A.4}$$

donde M es la magnitud del escalón

$$T = \frac{1}{K_1} \quad \text{Ecuación A.5}$$

Apéndice B

El programa

Se presentan imágenes del panel de control y el diagrama de bloques principal.

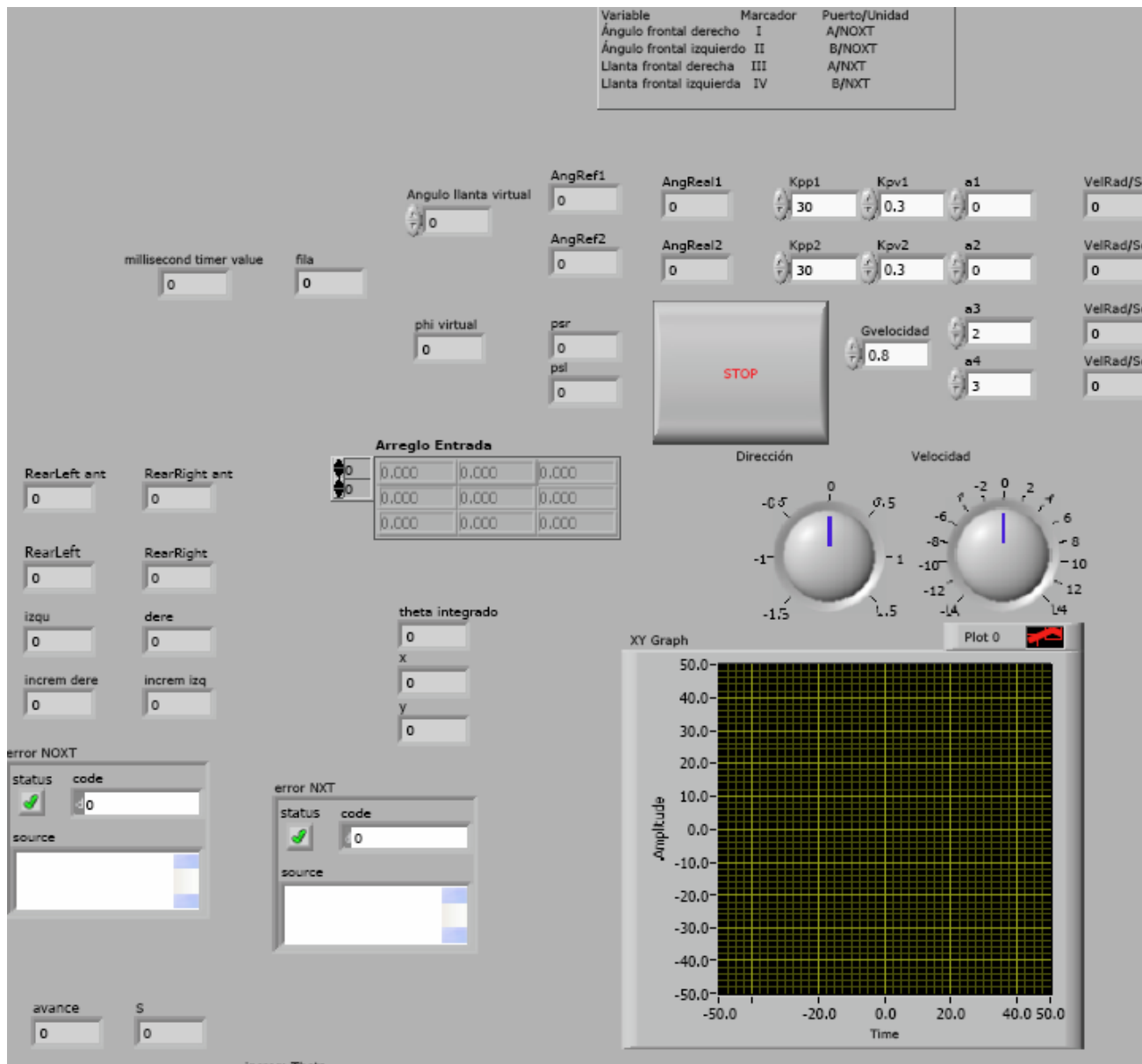


Figura B.1. Panel de control implementado en LabVIEW® 8.6

BLOCK DIAGRAM

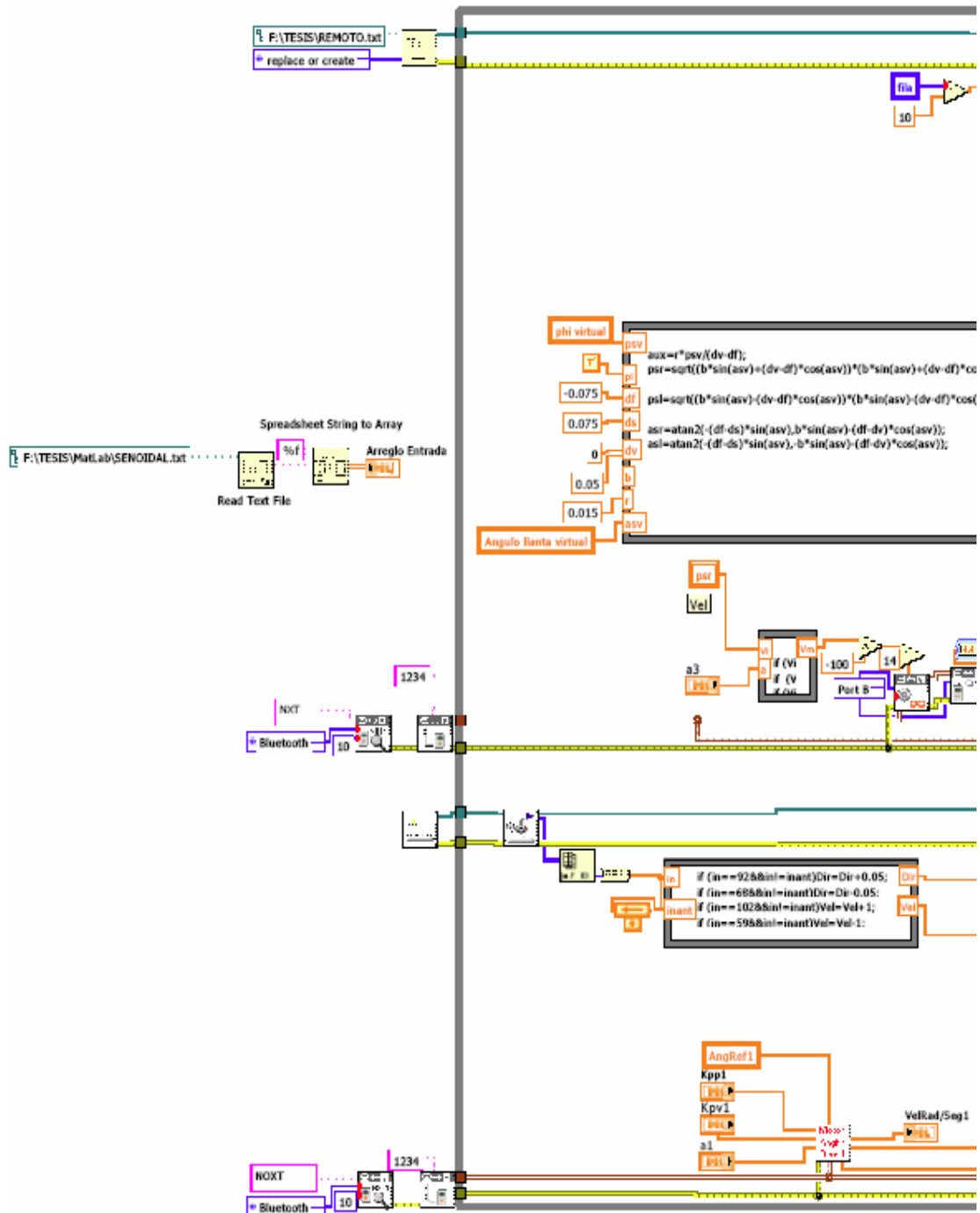


Figura B.2. Diagrama de bloques (1/4) implementado en LabVIEW® 8.6

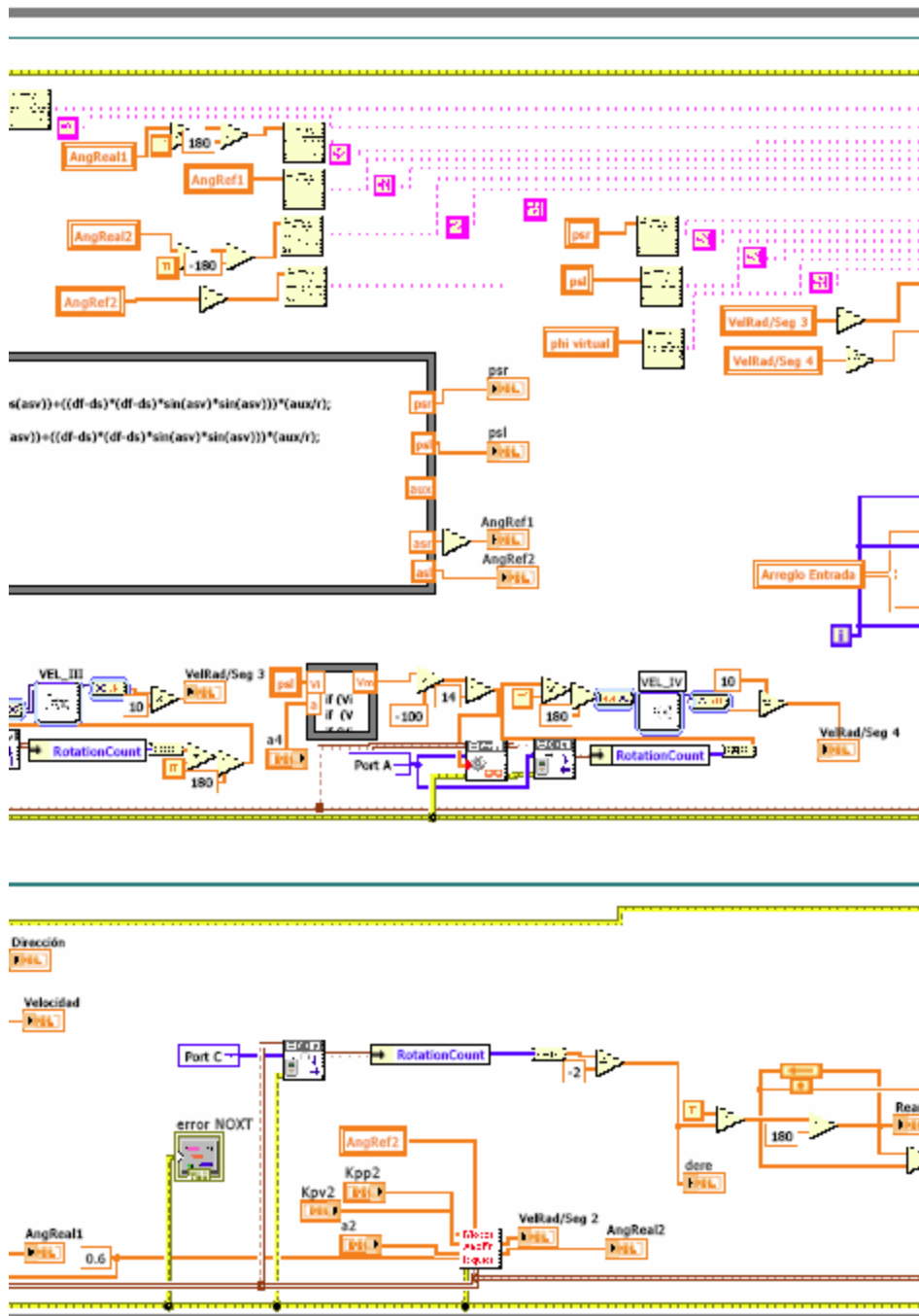


Figura B.3. Diagrama de bloques (2/4) implementado en LabVIEW®

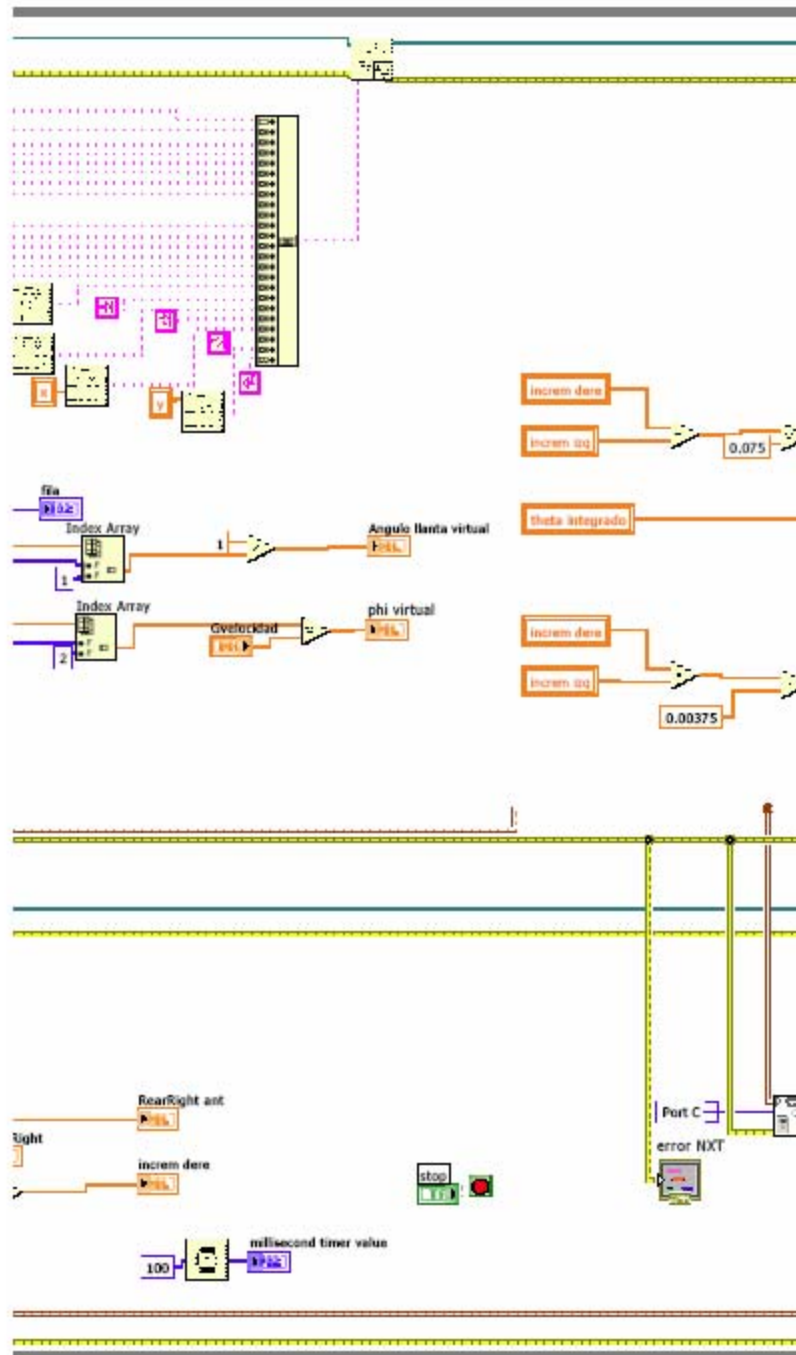


Figura B.4. Diagrama de bloques (3/4) implementado en LabVIEW® 8.6

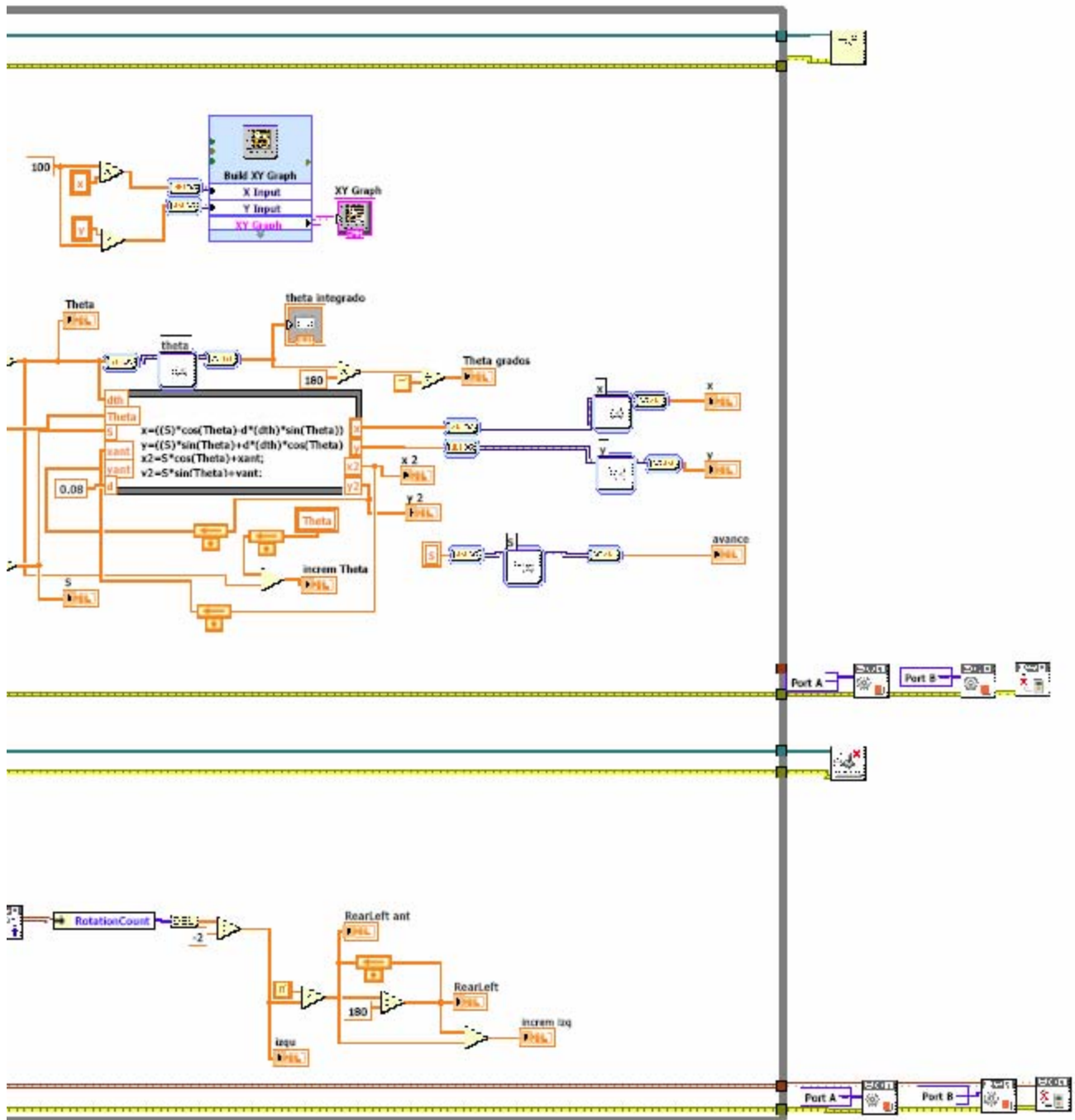


Figura B.5. Diagrama de bloques (4/4) implementado en LabVIEW® 8.6

Apéndice C

Artículo presentado en el Congreso Internacional Anual de la SOMIM, 23-25 Septiembre 2009, Cd Obregón, Sonora, México.

Experimentación con un robot móvil en una plataforma de simulación

Ing. Octavio Díaz Hernández* Dr. Víctor Javier González Villela**
 Departamento de Mecatrónica, Facultad de Ingeniería, UNAM
 Lab. Ing. Alberto Camacho, Anexo de Ingeniería
 Cd. Universitaria, MÉXICO, 04510, D.F.
 *zidh@hotmail.com,**vjgv@unam.mx

Resumen

En este trabajo es estudiado un robot móvil con llantas traseras motorizadas. Se ha requerido del uso de herramientas tales como una computadora personal y un medio de comunicación capaz de interactuar con un robot móvil a través de dicha computadora. El primer paso es probar la comunicación con un modelo simple de robot móvil ampliamente estudiado para continuar con el trabajo de coordinar llantas frontales independientes en su direccionabilidad.

Abstract

A mobile robot with rear motorized wheels is studied. It has required tools such as a personal computer, a communication media capable to interact with a mobile robot. The first approach is to probe communication with a simpler model thoroughly studied to continue with the coordination of frontal steerable wheels project.

Palabras clave: Robótica Móvil, Simulación, Interacción Computadora-Robot, Adquisición de Datos.

Introducción

Actualmente los robots móviles abarcan áreas de estudio tan básicas como juguetes y hasta exploradores del espacio. No hay duda de que las soluciones para modelar o experimentar con ellos son abundantes, tales como el uso de la electrónica digital combinacional básica hasta la implementación de sistemas complejos digitales con microprocesadores.

Las simulaciones de robots móviles, en programas de cómputo, que intentan emular su comportamiento cinemático y dinámico, con fundamento en la robótica móvil, evita la construcción temprana de modelos físicos, permite probar teorías como las del control antes de ser implementadas, o bien ser usadas como herramienta en el diseño de actuadores, etc.

En el rubro de la simulación se ha hecho importante poder ver el efecto en un sistema físico, capaz de realizar operaciones o acciones dirigidas, durante dichas simulaciones. Además, los sistemas computacionales han llegado rápidamente a altos niveles de procesamiento de la información capaces de manejar algoritmos como los de las redes neuronales, lógica difusa, algoritmos genéticos, etc., los cuales pueden ser aplicados a los robots móviles, pero necesitan el apoyo de un procesamiento especializado. Por lo tanto es indispensable estudiar la interacción de lenguajes de programación de alto nivel con los robots móviles.

Los puertos de una computadora personal han proveído de vías de comunicación con dispositivos mecatrónicos, el más común en aplicaciones educativas ha sido el puerto serial, aunque el puerto paralelo también tiene merito para ser mencionado. No obstante, la aparición del puerto serial universal (*Universal Serial Bus, USB*) ha hecho desaparecer a los dos primeros de las computadoras más recientes, por lo que se ha explorado la opción de tener interfaces de comunicación USB, tales como tarjetas de adquisición de datos.

En el presente trabajo se presenta una nueva forma de trabajar con modelos de simulación en conjunto con los modelos físicos. La fusión de tecnologías aquí descritas tales como la tarjeta adquisidora USB de *National Instruments* y el *SIMULINK®* son herramientas útiles que no están limitadas para propósitos académicos, sino que pueden trasladarse a aplicaciones industriales y otros sistemas mecatrónicos. A continuación se muestra una forma de robot móvil que con esta investigación, es capaz de llegar de un punto a otro, siendo el primer paso para lograr seguimiento de trayectorias, planeación de movimiento, evasión de obstáculos, teleoperación, etc.

Arquitectura del robot móvil

Antes de comenzar, es necesario tener una visión general de los componentes para realizar los experimentos para manejar un robot móvil. En la tabla 1 se muestran con más detalle los componentes del sistema.

ROBOT MÓVIL		
Subsistema	Materiales	
Procesador Principal	Computadora personal	Programas de uso general, MatLab, Simulink, LabView, Measurement and instrumentation, controladores de tarjeta de adquisición.
Programación	Simulink®	Procesamiento de las señales de <i>encoder</i> , Odometría, control de motores, Algoritmos de control
Comunicación entre Procesador y Robot	Tarjeta de adquisición de datos <i>National Instruments®</i>	<i>USB NI-DAQ 6009</i>
Actuación y sensado	Motores NXT®	Requiere 5 Volts para el <i>encoder</i>
Electrónica	Fuente de poder	
Fase de potencia	Puente H, L293D	Transmite señales de potencia al motor

Tabla 1. Componentes con los que se implementó el sistema.

En la figura 1 se resume como se comunican los sistemas haciendo notar que en la computadora personal se llevarán a cabo casi todos los algoritmos de control, que se verán más adelante.

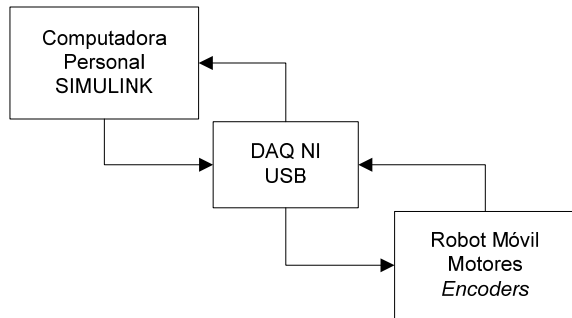


Figura 1. Arquitectura propuesta incluyendo al robot móvil

Simulink ®

Es una plataforma para simulación y diseño basado en modelos de sistemas dinámicos y embebidos. Proporciona un entorno gráfico interactivo y un conjunto de librerías de bloques personalizables que permiten diseñar, simular, implementar y probar una gran variedad de sistemas con variación temporal, entre los que se incluyen sistemas de comunicaciones, control, procesamiento de señales, vídeo e imágenes¹.

Se usó la versión 7.1 de *Simulink (MatLab® R2008a)* la cual contiene una “caja de herramientas” o “*toolbox*” de adquisición de datos (*Data Acquisition Toolbox, DAT*) cuya versión es la 2.12, que permite programar entradas y salidas que pueden ya sea analógicas o digitales de acuerdo a la interfaz física que se tenga.

Tarjeta de adquisición de datos

Cualquier puerto de comunicaciones que una computadora personal tenga, es potencialmente una interfaz física con la que se puede interactuar con el exterior, sin embargo en la actualidad, puertos como el puerto paralelo o el puerto serial han sido substituidos por el Bus Serial Universal (*Universal Serial Bus, USB*) dando paso a tarjetas con este tipo de comunicación, por lo que en este trabajo se emplea una tarjeta *National Instruments* modelo USB-6009, con ocho entradas analógicas (14-bits, 48 kilomuestras por segundo) y dos salidas analógicas (12-bits, 48 kilomuestras por segundo), doce puertos digitales de entrada/salida compatibles con voltajes TTL y un contador de 32 bits².



Figura 2. Tarjeta NI DAQ USB-6009, tomada de hoja de especificaciones

Motores NXT®

Es fundamental comentar que este tipo de servomotor posee un *Encoder* de Cuadratura para proporcionar información de la posición angular al subsistema correspondiente. Se tiene un engrane con 12 ventanas acoplada directamente al motor de corriente directa, teniendo una relación de 10:32 con respecto al motor. Posteriormente se une a una reducción de engranes con una relación de 1:48 en total hasta el eje de salida. De tal forma que por una vuelta de dicho eje de salida, el *encoder* realiza $48 \cdot 10 / 32 = 15$ vueltas, y cada detector óptico (*TACHO 0 y 1*) percibe $15 \cdot 12 = 180$ pulsos. En el programa que se describirá más adelante, se usa esta resolución y se multiplica por dos para obtener los 360°. En la figura 3 se observa su estructura interna.

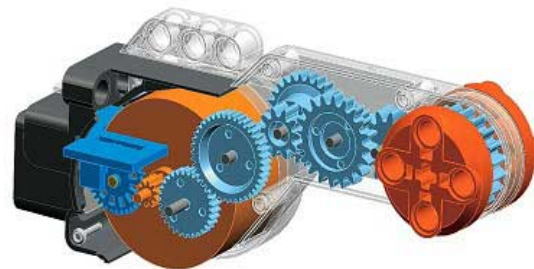


Figura 3. Dibujo de un motor NXT®, tomado de *NXT® motor internals [2009]*

A este motor se le puede energizar hasta con 12 volts, pero sólo se recomienda hasta 9 volts³, la corriente sin carga con este voltaje es de 60 mA y su velocidad es de 170 RPM. Cuando el rotor está inmovilizado su torque es de 50 N.m y su corriente es de 2 A.

Programación

Procesamiento de la señal del *encoder* de cuadratura en *Simulink*®

Por cada motor se tienen dos señales provenientes del *encoder* que se pueden ver en la figura como A y B, las cuales tienen un retraso (una con respecto de la otra) de acuerdo al sentido del motor, si es horario o antihorario.

El circuito que se ve en la Figura 5 es implementado en *Simulink*® con un arreglo de *Flip Flops* tipo D. A partir de A y B, se activa un tren de pulsos cuando el motor gira en sentido horario y otro cuando gira en sentido antihorario. La primera incrementa un conteo y la segunda decrementa dicho conteo. Este número final del contador, es el ángulo. Así mismo se hace un cálculo de la velocidad angular.

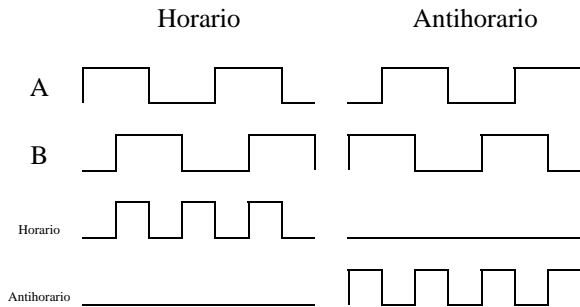


Figura 4. Señales del *encoder* de cuadratura y la decodificación con el circuito⁵.

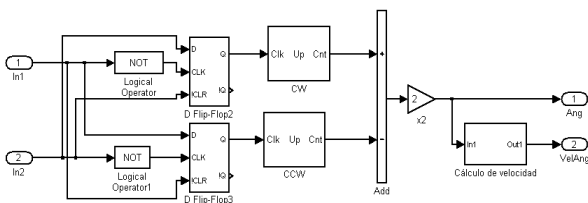


Figura 5. Implementación para el procesamiento del *encoder*.

Para el “cálculo de la velocidad” se implementó una rutina de derivación discreta de la salida del contador (el ángulo).

$$\dot{\varphi}_i = \frac{\theta_i - \theta_{i-1}}{t_i - t_{i-1}} \quad (1)$$

donde $\dot{\varphi}$ es la velocidad angular, θ es el ángulo, t es el tiempo.

Señales para manejo del motor

Es necesario crear en el ambiente de programación una señal con periodo fijo (T) y tiempos de encendido (t_H) y apagado (t_L) variables, es decir, método digital conocido como Modulación por Ancho de Pulso (*Pulse Width Modulation, PWM*). En la siguiente figura se detalla el modo en que se programó con un contador y un comparador simple.

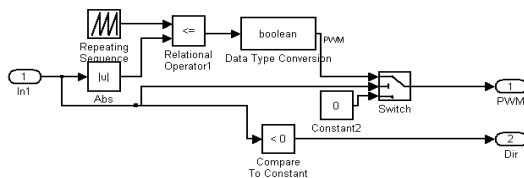


Figura 6. Generación del ancho de pulso y la dirección para los motores.

Con la señal de PWM es posible manejar la velocidad del motor, mientras que la dirección de giro (D) está dada por una señal binaria, donde “0” es “Sentido Antihorario” y “1” es “Sentido Horario”. La electrónica de potencia se implementó con un circuito integrado L293D, como se ve en la figura. haciendo uso del PWM como el pin de habilitación (EN) y las entradas (IN) son D y el opuesto de D (\bar{D})

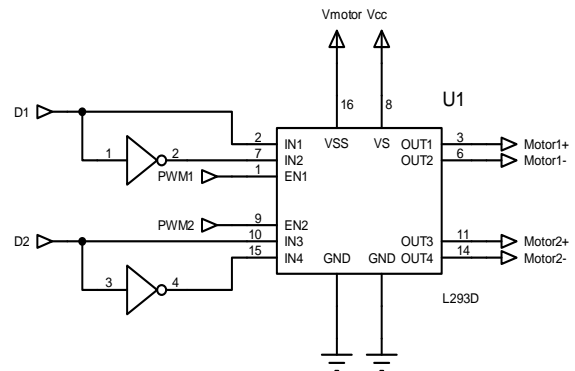


Figura 7. Circuito electrónico de potencia para dos motores con un Puente H, L293D⁴

Estudio de la posición de un robot móvil diferencial

Se trata de un robot con dos llantas motorizadas direccionables, rotan respecto a un eje perpendicular al plano de la figura 8.

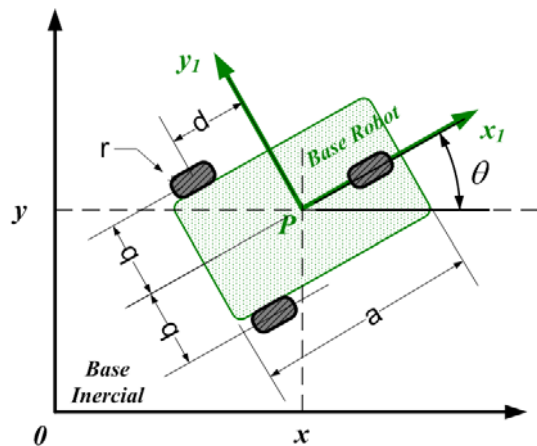


Figura 8. Representación de robot móvil

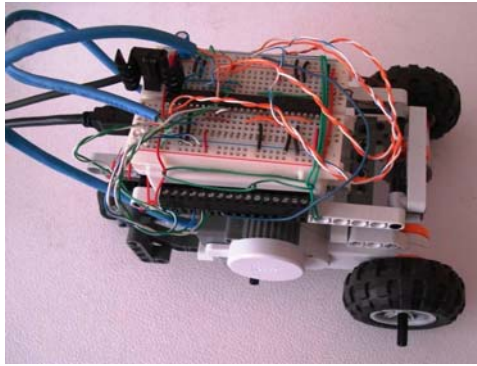


Figura 9. Robot Móvil para experimentación

A partir de la cinemática se obtienen las ecuaciones proporcionan información de su posición en el plano con respecto a una base inercial.

La posición del punto P en el robot está descrita por $\varepsilon = (x, y, \theta)^T$ y su derivada con respecto al tiempo es $\dot{\varepsilon} = (\dot{x}, \dot{y}, \dot{\theta})^T$, cabe mencionar que θ es el ángulo formado por el eje longitudinal del robot y el eje X del sistema de referencia fijo, mientras que $\dot{\theta}$ es la velocidad angular del robot.

En este caso se puede disponer de información de dos llantas traseras, la derecha (r) y la izquierda (l). Sus desplazamientos angulares respectivos son ϕ_r y ϕ_l , mientras que sus velocidades angulares son $\dot{\phi}_r$ y $\dot{\phi}_l$.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi}_r \\ \dot{\phi}_l \end{bmatrix} = \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \\ 0 & 1 \\ 1/r_w & b/r_w \\ 1/r_w & -b/r_w \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (2)$$

v es la velocidad lineal del robot

ω es la velocidad angular, con respecto a la base inercial.

b es la distancia del eje x hacia el eje perpendicular a la llanta.

d es la distancia del eje de las llantas al punto P.

r_w es el radio de ambas llantas

Obteniendo las ecuaciones para cada variable en términos de velocidad lineal y angular.

$$\begin{aligned} \dot{x} &= v \cos \theta - \omega d \sin \theta \\ \dot{y} &= v \sin \theta + \omega d \cos \theta \\ \dot{\theta} &= \omega \end{aligned} \quad (3)$$

$$\dot{\phi}_r = v / r_w + b \omega / r_w$$

$$\dot{\phi}_l = v / r_w - b \omega / r_w$$

Si, $v = \dot{x}_1$ y $\omega = \dot{\theta}_1 = \dot{\theta}$, las ecuaciones para cada variable están en términos de incrementos velocidad lineal $\frac{\Delta x_1}{\Delta t}$ y angular $\frac{\Delta \theta}{\Delta t}$, donde Δt es factor común en ambos lados de las ecuaciones, por lo que se simplifica.

$$\begin{aligned} \Delta x &= \Delta x_1 \cos \theta - \Delta \theta d \sin \theta \\ \Delta y &= \Delta x_1 \sin \theta + \Delta \theta d \cos \theta \\ \Delta \theta &= \Delta \theta \end{aligned} \quad (4)$$

$$\Delta \phi_r = \Delta x_1 / r_w + b \Delta \theta / r_w$$

$$\Delta \phi_l = \Delta x_1 / r_w - b \Delta \theta / r_w$$

Resolviendo el sistema de ecuaciones $\Delta \phi_r = \Delta x_1 / r_w + b \Delta \theta / r_w$ y $\Delta \phi_l = \Delta x_1 / r_w - b \Delta \theta / r_w$ conociendo los incrementos angulares provenientes de los *encoders* de cada llanta, se tiene $\Delta x_1 = \frac{r_w}{2} (\Delta \phi_r + \Delta \phi_l)$ y $\Delta \theta = \frac{r_w}{2b} (\Delta \phi_r - \Delta \phi_l)$.

Los incrementos Δx_1 y $\Delta \theta$ son sustituidos en las dos primeras ecuaciones de (4), para obtener Δx y Δy . La forma en que se determinan las coordenadas $\varepsilon = (x, y, \theta)^T$ es de tipo iterativo pues $\Delta x = x_i - x_{i-1}$ y $\Delta y = y_i - y_{i-1}$, tal que:

$$x_i = \Delta x_1 \cos \theta - \Delta \theta d \sin \theta + x_{i-1} \quad (5)$$

$$y_i = \Delta x_1 \sin \theta + \Delta \theta d \cos \theta + y_{i-1} \quad (6)$$

Asimismo:

$$\theta_i = \Delta \theta + \theta_{i-1} \quad (7)$$

La **odometría** es el estudio de la estimación de la posición de vehículos con ruedas durante la navegación. Para realizar esta estimación se usa información sobre la rotación de las ruedas para estimar cambios en la posición a lo largo del tiempo. La palabra odometría se compone por las palabras griegas *hodos* ("viajar", "trayecto") y *metron* ("medida"). Este método se basa en la suposición de que las revoluciones de las ruedas pueden ser traducidas en un desplazamiento lineal relativo al suelo.

La localización del robot usando la predicción odométrica (*dead reckoning*) es bastante preciso en ausencia de deslizamiento y derrape de la llanta. Estos efectos se reducen cuando la velocidad se mantiene baja.

Con el conocimiento relativo de $\varepsilon = (x, y, \theta)^T$ se puede mover el robot hacia un punto llamado P_g en el plano con coordenadas (x_f, y_f, θ_f) . El error resultante es $e_x = x_f - x$, $e_y = y_f - y$ y $e_\theta = \theta_g - \theta$. Sin embargo para el control del robot propuesto en [6] se necesita además el error de distancia al punto P_g definido como $e_d = \sqrt{e_x^2 + e_y^2}$ y el de ángulo $\theta_g = \tan^{-1}[e_y/e_x]$.

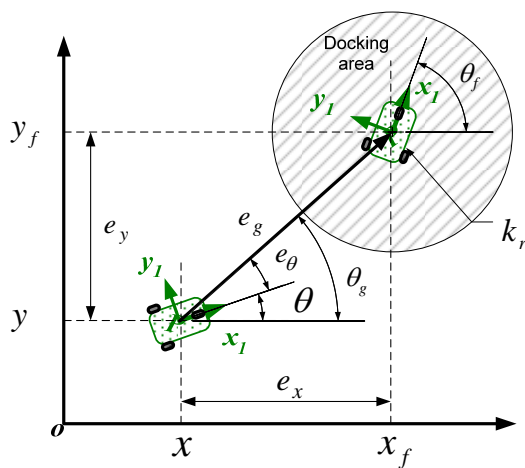


Figura 10. Definición del punto P_g y su Área de Llegada (*Docking Area*). Tomada de [6]

De tal manera que se definieron ciertas reglas para las velocidades lineal y angular.

$$v = \begin{cases} v_{\max} & \text{si } |e_d| > k_r \\ \frac{v_{\max}}{k_r} \cdot e_d & \text{si } |e_d| \leq k_r \end{cases};$$

$$\omega = \omega_{\max} \sin(e_\theta)$$

Donde v_{\max} es la velocidad lineal máxima, ω_{\max} es la velocidad angular máxima del robot, mientras que k_r es el radio del Área de Llegada (*Docking Area*).

Resultados

El robot móvil y su control, se han logrado implementar con otros sistemas robóticos muy similares, sin embargo el presente trabajo ha servido para demostrar que la arquitectura del sistema aquí planteado funciona y así poder experimentar con otros tipos de robots móviles más complejos.

En la figura 11 se observa la representación del punto P del robot móvil que se mueve hacia el punto P_g encerrado en el área de llegada con radio k_r . El punto P está representando al robot móvil de tipo diferencial que inicia con coordenadas y ángulo en cero dentro del programa de simulación.

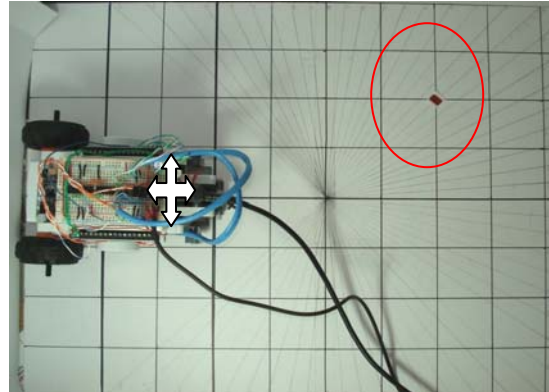


Figura 11 La cruz indica el punto P del robot que se encuentra en posición (0,0), El área de llegada y el punto de referencia, donde el eje x es el horizontal, mientras que el eje y es vertical

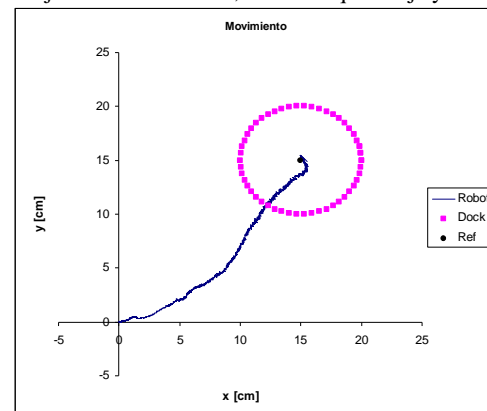


Figura 12. Simulación de la trayectoria seguida por el punto P hasta alcanzar P_g (Ref).

Los valores propuestos para los experimentos fueron:

- $b = 5.5 \text{ cm}$
- $d = 13 \text{ cm}$
- $r_w = 2.8 \text{ cm}$
- $v_{\max} = 40 \text{ \% PWM}$
- $\omega_{\max} = 0.8 \text{ rad/seg}$

El movimiento “errático” de la trayectoria trazada por el robot móvil antes de llegar al punto de referencia, en la Figura 11, puede ser provocado por la comunicación retrasada entre la computadora y el robot. Por otro lado es posible que los valores de v_{\max} y ω_{\max} puedan ser manipulados con el fin de linearizar la trayectoria del robot.

REFERENCIAS

- 1) Dinsdale, J., Yamazaki, K. Mechatronics and ASIC's. Annals of CIRP 38, 1989.
- 2) Gopalakrishnan, B., Tirunellayi, S. Design and development of an autonomous mobile smart vehicle: a mechatronics application. Mechatronics, 14, pp 491-514, 2004
- 3) Kolmanovsky, I. McClamroch, N.H. Developments in nonholonomic control problems, IEEE Control Systems Magazine, Volume: 15, Issue: 6, pp 20-36, 1995
- 4) Lee, D., Martinez-Palafox, O., Spong, M. Bilateral Teleoperation of a Wheeled Mobile Robot over Delayed Communication Network. Proceedings of the 2006 IEEE International Conference on Robotics and Automation Orlando, Florida - May 2006
- 5) Lim, J., Ko, J., Lee, J. Internet-based teleoperation of a mobile robot with force-reflection. Control Applications Proceedings of 2003 IEEE Conference. Vol: 1, pp 680- 685 June 2003
- 6) Nyholm Jørgensen, R., Nørremark, M. HortiBot - A Plant Nursing Robot. Página electrónica consultada en diciembre de 2007 disponible en <<http://hortibot.com>>
- 7) Islam, S., Azim, A., Jahan, S., Toman, M. Design and Synthesis of Mobile Robot Controller using Fuzzy Logic. Int. Conf. on Software Engineering 2006, Kuala Lumpur, Malaysia Proc. 2006
- 8) Hendzel, Z. An adaptive critic neural network for motion control of a wheeled mobile robot. Nonlinear Dyn 50, pp 849–855. 2007
- 9) Mauder, M. Robust tracking control of nonholonomic dynamic systems with application to the bi-steerable mobile robot. Automatica 44, pp 2588–2592, 2008
- 10) Reister, D., Unseren, M. Position and Constraint Force Control of a Vehicle with Two or More Steerable Drive Wheels. IEEE Transactions On Robotics And Automation, Vol. 9, No. 6, December 1993
- 11) N. Matsumoto and M. Tomizuka, "Vehicle lateral velocity and yaw rate control with two independent control inputs," in *Proc. 1990 Amer. Control Conf.*, pp. 1868-1875, 1990
- 12) Sekhavat, S.; Hermosillo, J. Cycab bi steerable cars a new family of differentially flat systems. Advanced Robotics, Volume 16, Number 5, pp. 445-462(18), 2002
- 13) Burke, T., Durrant-Whyte, H. Kinematics For Modular Wheeled Mobile Robots. Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems Yokohama, Japan July 26-30, 1993
- 14) Thuilot, B. D'Andrea-Novel, B., Micaelli, A. Modeling and Feedback Control of Mobile Robots. IEEE Transactions On Robotics And Automation, Vol. 12, No. 3, June 1996

- 15) Mutambara, A., Durrant-Whyte, H. Estimation and Control for a Modular Wheeled Mobile Robot. IEEE Transactions On Control Systems Technology, Vol. 8, No. 1, January 2000
- 16) Lamiroux, F.; Sekhavat, S.; Laumond, J.-P.; Motion planning and control for Hilare pulling a trailer. IEEE Transactions on Robotics and Automation, Volume 15, Issue 4, Aug. pp 640-652, 1999
- 17) Brooks, R. A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, Volume 2, Issue 1, Mar, pp 14-23, 1986
- 18) McKerrow, P. Introduction to Robotics, Addison-Wesley, 1993
- 19) Campion, G.; Bastin, G.; Dapos; Andrea-Novel, B. Structural properties and classification of kinematic and dynamic models of wheeled mobile robots IEEE International Conference on Robotics and Automation, 1993. Proceedings., 1993 Volume , Issue , 2-6 May 1993 Page(s):462 - 469 vol.1
- 20) Ollero, A. Robótica, Manipuladores y robots móviles. Marcombo, España, 2001. pp 168-178
- 21) Diaz-Hernandez, O., Gonzalez-Villela, VJ. Experimentación con un robot móvil en una plataforma de simulación. Memorias XV Congreso Internacional de la SOMIM, 23-25 septiembre, Cd. Obregón, Sonora., pp 859-565, 2009
- 22) Bolton, W. Mecatrónica. Sistemas de control electrónico en ingeniería mecánica y eléctrica. 3ª Ed., Alfaomega, México, 2001
- 23) Diaz-Hernandez, O. Apuntes de Modelado de sistemas Mecatrónicos, Posgrado de ingeniería, UNAM, semestre 2008-1
- 24) NXT® motor internals, página web consultada en Enero de 2009, disponible en <
<http://www.philohome.com/nxtmotor/nxtmotor.htm>>
- 25) US-DIGITAL™, página web consultada en Marzo de 2009, disponible en <
<http://usdigital.com/products/encoders/incremental/rotary/kit/e4p/#purchasing>>
- 26) Alciatore, D. Introducción a la mecatrónica y los sistemas de medición. Tercera edición. McGraw-Hill, México, 2008
- 27) Gasperi, M., Hurbain, P. Extreme NXT: Extending the LEGO® MINDSTORMS® NXT® to the Next Level, Editorial APRES, Estados Unidos, 2007
- 28) Hansen, J. NXT® Power programming, VARIANT press, Canada, 2007
- 29) Groover, M., Weiss, M. Robótica industrial, Mcgraw Hill, México, 1990.
- 30) Gonzalez-Villela, V.J. Research on a semiautonomous mobile robot for loosely structured environments focused on transporting mail trolleys. Thesis (Ph.D.) Loughborough University, Department of Mechanical Engineering, 2006