



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA**

CARRERA DE INGENIERÍA EN MECATRÓNICA

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO MECATRÓNICO**

**TEMA: IMPLEMENTACIÓN DE TÉCNICAS DE SIMULACIÓN
HARDWARE Y SOFTWARE IN-THE-LOOP PARA REALIZAR
PRUEBAS EN TIEMPO REAL DE UN MODELO DE VEHÍCULO
UTILIZANDO LA PLATAFORMA DSPACE**

AUTOR: SANTILLÁN ARIAS, GUIDO VINICIO

DIRECTOR: ING. SEGURA SANGUCHO, LUIS JAVIER

CODIRECTOR: DR.- ING. OERTEL, CHRISTIAN

SANGOLQUÍ – ECUADOR

2016




**DEPARTAMENTO DE CIENCIAS DE LA
ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA EN MECATRÓNICA**

CERTIFICACIÓN

Certifico que el trabajo de titulación, "**IMPLEMENTACIÓN DE TÉCNICAS DE SIMULACIÓN HIL&SIL PARA REALIZAR PRUEBAS EN TIEMPO REAL DE UN MODELO DE VEHÍCULO UTILIZANDO LA PLATAFORMA DSPACE**" realizado por el señor **SANTILLAN ARIAS GUIDO VINICIO**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar al señor **SANTILLÁN ARIAS GUIDO VINICIO** para que lo sustente públicamente.

Sangolquí, 09 de enero del 2017



LUIS JAVIER SEGURA SANGUCHO
DIRECTOR



**DEPARTAMENTO DE CIENCIAS DE LA
ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA EN
MECATRÓNICA**

AUTORÍA DE RESPONSABILIDAD

Yo, **SANTILLÁN ARIAS GUIDO VINICIO**, con cédula de identidad N° 1720023207, declaro que este trabajo de titulación "**IMPLEMENTACIÓN DE TÉCNICAS DE SIMULACIÓN HIL&SIL PARA REALIZAR PRUEBAS EN TIEMPO REAL DE UN MODELO DE VEHÍCULO UTILIZANDO LA PLATAFORMA DSPACE**" ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 09 de enero del 2017

GUIDO VINICIO SANTILLÁN ARIAS

C.C. 172002320-7



**DEPARTAMENTO DE CIENCIAS DE LA
ENERGÍA Y MECÁNICA
CARRERA DE INGENIERÍA EN
MECATRÓNICA**

AUTORIZACIÓN

Yo, **SANTILLÁN ARIAS GUIDO VINICIO**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación “**IMPLEMENTACIÓN DE TÉCNICAS DE SIMULACIÓN HIL&SIL PARA REALIZAR PRUEBAS EN TIEMPO REAL DE UN MODELO DE VEHÍCULO UTILIZANDO LA PLATAFORMA DSPACE**” cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Sangolquí, 09 de enero del 2017

GUIDO VINICIO SANTILLÁN ARIAS

C.C. 172002320-7

DEDICATORIA

A mi familia Guido, Ritha y Michelle.

AGRADECIMIENTO

A mi familia Guido, Ritha y Michelle, por su apoyo, comprensión, amor, ayuda en los momentos difíciles, y por ayudarme con los recursos necesarios para estudiar. Me han dado todo lo que soy como persona, mis valores, mis principios, mi carácter, mi empeño, mi perseverancia, mi coraje para conseguir mis objetivos.

A la Universidad de las Fuerzas Armadas – ESPE, Carrera de Ingeniería en Mecatrónica y a todos docentes por compartir sus conocimientos y colaboración en mi formación profesional.

A mi director de tesis, Ing. Luis Segura, por su esfuerzo y dedicación, quien, con sus conocimiento y experiencia, ha logrado que yo pueda terminar mis estudios con éxito.

A la Unidad de Relaciones Interinstitucionales y a la Unidad de Bienestar Estudiantil por darme la oportunidad de realizar mi trabajo de titulación en el exterior.

A mis compañeros, por ser una fuente de motivación y aprendizaje a lo largo de la carrera.

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN.....	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN.....	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE DE CONTENIDOS.....	vii
ÍNDICE DE TABLAS.....	xv
ÍNDICE DE FIGURAS.....	xvi
RESUMEN.....	xxii
ABSTRACT	xxiii
CAPÍTULO 1	24
1. GENERALIDADES	24
1.1. ANTECEDENTES	24
1.2. OBJETIVOS	25
1.2.1. GENERAL.....	25
1.2.2. ESPECÍFICOS	25
1.3. JUSTIFICACIÓN E IMPORTANCIA	25
CAPÍTULO 2	27
2. MARCO TEÓRICO	27
2.1. INTRODUCCIÓN A HARDWARE Y SOFTWARE IN-THE-LOOP	27
2.2. BENEFICIOS DE HIL & SIL	28
2.3. Aplicaciones generales	30

2.4.	PRINCIPALES CAMPOS DE APLICACIÓN DE HIL&SIL	31
2.4.1.	Desarrollo de subsistemas de vehículos.	31
2.4.2.	Diseño y desarrollo de unidades de control electrónicas (ECU)	31
2.4.3.	Aplicaciones en vehículos eléctricos	31
2.4.4.	Manejo de potencia en vehículos	31
2.4.5.	Aplicaciones varias	32
2.5.	COMPONENTES DE SISTEMAS HIL	32
2.5.1.	Modelo de la planta	32
2.5.2.	Sensores	33
2.5.3.	Computadora en tiempo real	33
2.5.4.	Cargas	33
2.5.5.	Computadora anfitrión	34
2.5.6.	Diagnóstico de la ECU	34
2.5.7.	Unidad de suministro de energía	34
2.5.8.	Unidad de inserción de fallas (FIU)	34
2.6.	TARJETA PROCESADORA DSPACE DS1006	35
2.6.1.	Propiedades	36
2.6.2.	Aplicaciones típicas	37
2.6.3.	DSPACE ControlDesk software	37
2.7.	OBJETIVOS DESEADOS EN EL DESARROLLO DE VEHÍCULOS ..	37
2.7.1.	Sistema de dirección asistida eléctricamente (EPS)	38
2.7.2.	Sistema de control de suspensión	39
2.7.3.	Control de climatización de un vehículo	40
2.7.4.	Sistema activo de control de roll (ARC)	41
2.7.5.	Sistema de control de estabilidad	42

2.7.6.	Sistema de control del motor.....	43
2.7.7.	Control de crucero adaptativo (ACC).....	43
2.7.8.	Sistema avanzado de asistencia al conductor.....	44
2.8.	TIPOS DE ELEMENTOS DISPONIBLES EN SIMULINK	45
2.8.1.	Funciones S	45
2.8.2.	Funciones de código de MATLAB	46
2.9.	MODELAMIENTO DE SISTEMAS	46
2.9.1.	Definición de un sistema	47
2.10.	DINÁMICA DE VEHÍCULOS	47
2.11.	CONCEPTOS COMUNES EN DINÁMICA DE VEHÍCULOS	49
2.11.1.	Ángulo de roll.	49
2.11.2.	Ángulo de pitch.	49
2.11.3.	Ángulo de yaw.....	49
2.11.4.	Fuerza de tracción [Fu].	50
2.11.5.	Fuerza lateral [Fs].	50
2.11.6.	Torque de alineación [Mz].	50
2.11.7.	Deslizamiento longitudinal.....	51
2.11.8.	Deslizamiento lateral.	51
2.12.	MODELOS DE VEHÍCULOS.....	51
2.12.1.	Modelo “Single-track”	53
2.12.2.	Modelo “Twin-track”	54
2.12.3.	Modelo completo de vehículo y sistemas multicuerpo.....	55
2.13.	HERRAMIENTAS PARA CONSTRUIR MODELOS OFFLINE.....	56
2.13.1.	MSC ADAMS	56
2.13.2.	Simpack.....	57

CAPÍTULO 3	59
3. IMPLEMENTACIÓN DE UN MODELO DE PRUEBA OFFLINE	59
3.1. INTRODUCCIÓN AL MODELO.....	59
3.2. ESPECIFICACIONES DEL MODELO	60
3.3. BLOQUES DE FUNCIÓN EN SIMULINK	60
3.3.1. Modelo versión 1 – Usando bloques de función de MATLAB	60
3.3.2. Modelo versión 2 – Usando bloques de funciones-S nivel 2.....	62
3.3.3. Modelo versión 3 – Usando bloques de funciones-S en C	64
3.4. IMPLEMENTACIÓN DEL MODELO EN SIMULINK	66
3.4.1. Entradas y salidas del modelo.....	66
3.4.2. Sistema de dirección.....	66
3.4.3. Variación del yaw	68
3.4.4. Controlador de yaw	69
3.4.5. Modelo de vehículo	70
3.4.6. Ecuaciones de movimiento	74
3.4.7. Modelo de llanta.....	75
3.4.7.1. Transformación de sistemas	76
3.4.7.2. Ecuación dinámica para el movimiento angular de la llanta	76
3.4.7.3. Fuerzas presentes en las llantas.....	77
3.4.7.4. RMOD-K Formula	78
3.5. PARÁMETROS DE SIMULACIÓN DEL MODELO	79
3.5.1. Especificaciones de la computadora anfitrión.....	80
CAPÍTULO 4	81
4. ANÁLISIS EN TIEMPO REAL DE UN MODELO COMPLEJO	81
4.1. DINÁMICA DEL MODELO COMPLEJO VEHÍCULO	81

4.2.	SISTEMA DE FRENOS CON CONTROL ANTIBLOQUEO (ABS).....	83
4.2.1.	Sistema de frenos	83
4.2.2.	Sistema antibloqueo de frenos (ABS).....	84
4.2.3.	Modelo matemático.....	86
4.2.4.	Modelo en Simulink.....	87
4.3.	SUSPENSIÓN CON CONTROL ACTIVO DEL CHASIS (ABC)	90
4.3.1.	Sistema de suspensión	90
4.3.2.	Control Activo del Chasis	90
4.3.3.	Modelo matemático.....	90
4.3.3.1.	Fuerzas en las llantas	92
4.3.3.2.	Fuerzas en la suspensión activa.	93
4.3.3.3.	Efectos de roll y pitch en el vehículo	94
4.3.3.4.	Espacio de estados.....	95
4.3.4.	Controlabilidad en espacio de estados.....	99
4.3.5.	Observabilidad en espacio de estados.....	99
4.3.6.	Diseño por ubicación de polos	99
4.3.7.	Observadores del estado	100
4.3.8.	Implementación del controlador y del observador	102
4.3.8.1.	Análisis de controlabilidad y observabilidad	103
4.3.8.2.	Polos deseados para el controlador	104
4.3.8.3.	Polos deseados para el observador	105
4.3.8.4.	Diagrama de Bloques.....	105
4.3.9.	Modelo en Simulink.....	106
4.4.	PARÁMETROS DE SIMULACIÓN DEL MODELO	107
	CAPÍTULO 5	111

5. TRANSFERENCIA DEL MODELO A LA PLATAFORMA DSPACE...	111
5.1. DSPACE CONTROL DESK	111
5.2. TRANSFERENCIA DEL MODELO DE VEHÍCULO A DSPACE	112
5.2.1. Definición del directorio en MATLAB	112
5.2.2. Habilitación de la plataforma RTI	113
5.2.3. Habilitación del toolbox RTI para el equipo DS1006.....	113
5.2.4. Configuración de los parámetros del toolbox y de la simulación....	114
5.2.5. Compilación y generación del código del modelo.....	114
5.2.6. Conexión con el equipo DS1006 y con el software ControlDesk. ..	115
5.2.7. Creación de un nuevo experimento en ControlDesk	116
5.2.8. Cargar el archivo compilado al equipo DS1006.....	116
5.2.9. Creación de la interfaz grafica.....	116
5.2.10. Enlace de las variables que se quieran medir	118
5.2.11. Configuración de parámetros en la interfaz gráfica.	118
5.2.12. Correr la simulación	119
5.2.13. Capturar datos	119
CAPÍTULO 6	121
6. PRUEBAS Y ANÁLISIS DE RESULTADOS.....	121
6.1. MODELO COMPLEJO	121
6.1.1. Simulaciones y pruebas	121
6.1.2. Test 1.....	122
6.1.2.1. Análisis movimiento - Test 1.....	123
6.1.2.2. Análisis de suspensión y ABC - Test 1.....	126
6.1.3. Test 2.....	127
6.1.3.1. Análisis movimiento - Test 2.....	127

6.1.3.2. Análisis de freno y ABS - Test 2.....	129
6.1.3.3. Análisis de suspensión y ABC - Test 2.....	130
6.1.4. Test 3.....	131
6.1.4.1. Análisis movimiento - Test 3.....	131
6.1.4.2. Análisis de freno y ABS - Test 3.....	133
6.1.4.3. Análisis de suspensión y ABC - Test 3.....	134
6.1.5. Test 4.....	135
6.1.5.1. Análisis movimiento - Test 4.....	135
6.1.5.2. Análisis de freno y ABS - Test 4.....	137
6.1.5.3. Análisis de suspensión y ABC - Test 4.....	138
6.1.6. Test 5.....	139
6.1.6.1. Análisis movimiento - Test 5.....	139
6.1.6.2. Análisis de suspensión y ABC - Test 5.....	141
6.1.7. Test 6.....	142
6.1.7.1. Análisis movimiento - Test 6.....	142
6.1.7.2. Análisis de suspensión y ABC - Test 6.....	144
6.1.8. Medición de los parámetros de rendimiento en Simulink.....	145
6.2. DSPACE	146
6.3. MEDICIÓN DE LOS PARÁMETROS DE RENDIMIENTO.....	148
6.3.1. Simulink	148
6.3.2. DSPACE	148
6.4. LIMITACIONES DEL MODELO.....	148
6.5. EXPORTACIÓN DE DATOS Y ANIMACIÓN.....	149
CAPÍTULO 7	150
7. CONCLUSIONES Y TRABAJO FUTURO	150

7.1. CONCLUSIONES.....	150
7.2. TRABAJO FUTURO.....	151
REFERENCIAS BIBLIOGRÁFICAS.....	152

ÍNDICE DE TABLAS

Tabla 1. Grados de libertad de algunos modelos de vehículo.....	52
Tabla 2. Subsistemas, entradas y condiciones de un modelo.....	52
Tabla 3. Nomenclatura de los parámetros utilizados en el modelo	67
Tabla 4. Parámetros de simulación para el modelo GO-KART	79
Tabla 5. Parámetros de los frenos	86
Tabla 6. Parámetros para el modelado de la suspensión.	92
Tabla 7. Parámetros de simulación.....	122
Tabla 8. Pruebas del freno y del ABS	122
Tabla 9. Grados de libertad de algunos modelos de vehículo.....	145
Tabla 10. Grados de libertad de algunos modelos de vehículo	145
Tabla 11. Medición del rendimiento del modelo en Simulink.....	148
Tabla 12. Medición del rendimiento del modelo en ControlDesk.....	148

ÍNDICE DE FIGURAS

Figura 1. Diagrama de bloques de una aplicación HIL&SIL.	28
Figura 2. Aplicaciones en vehículos.	30
Figura 3. Aplicaciones en la industria aeroespacial.	30
Figura 4. Modelado de subsistemas de un vehículo.	32
Figura 5. DS1006 Vista frontal.	35
Figura 6. DS1006 Vista posterior.	35
Figura 7. Librerías en Simulink para aplicaciones HIL&SIL.	36
Figura 8. Sistema asistido de dirección.	38
Figura 9. Esquema de un concepto de sistema HIL.	42
Figura 10. Aplicación de ACC.	44
Figura 11. Movimiento del vehículo.	49
Figura 12. Fuerzas y momentos presentes en la llanta.	50
Figura 13. Modelo bicicleta.	53
Figura 14. Modelo de doble vía.	54
Figura 15. BMV serie 5.	55
Figura 16. Proceso de modelado.	56
Figura 17. SIMPACK para automóviles.	58
Figura 18. SIMPACK para HIL&SIL.	58
Figura 19. GO-KART real (Technische Fachschule Brandenburg).	59
Figura 20. Diagrama del modelo de vehículo.	60
Figura 21. Bloque de función de MATLAB.	61
Figura 22. Parámetros de la función.	61
Figura 23. Incompatibilidad entre versiones de MATLAB.	62

Figura 24. Bloque de función-S nivel 2 de MATLAB.	62
Figura 25. Ventana de edición del bloque.	63
Figura 26. Bloque de función-S en C.	64
Figura 27. Ventana de edición del bloque.	64
Figura 28. Estructura de programación la función-S en C.	65
Figura 29. Bloque del sistema de dirección en Simulink.	66
Figura 30. Grafica del ángulo de dirección vs tiempo de simulación.	68
Figura 31. Bloque de la variación de ángulo de yaw en Simulink.	69
Figura 32. Comparación del valor real con el teórico.	69
Figura 33. Bloque del controlador en Simulink.	70
Figura 34. Gráfica de las salidas del controlador.	70
Figura 35. Esquema del vehículo.	71
Figura 36. Bloques de las ecuaciones de movimiento del vehículo.	75
Figura 37. Bloque para el cálculo de las fuerzas de las llantas.	75
Figura 38. Transformación de coordenadas.	76
Figura 39. Fuerzas que afectan a la rueda.	77
Figura 40. Modelo de la llanta en Simulink.	78
Figura 41. Configuración parámetros de la simulación.	80
Figura 42. Bloques de la dirección y yaw del vehículo.	82
Figura 43. Bloque de la dinámica del modelo complejo.	82
Figura 44. Modelo complejo en subsistema.	83
Figura 45. Componentes del sistema de freno.	84
Figura 46. Condiciones de frenado.	85
Figura 47. Modelo de sistema de frenos con ABS integrado.	87
Figura 48. Controlador del ABS.	88

Figura 49. Modelo del sistema de frenos en cada llanta.	89
Figura 50. Movimiento angular de la llanta frontal 1.	89
Figura 51. Movimiento angular de la llanta posterior 4.	89
Figura 52. Suspensión para neumáticos típica.	90
Figura 53. Estructura del modelo complejo.	91
Figura 54. Uso de ABC en vehículos comerciales.	91
Figura 55. Diagrama de bloques del espacio de estados.	96
Figura 56. Comprobación de controlabilidad y observabilidad.	103
Figura 57. Mapa polos y ceros del Controlador.	104
Figura 58. Mapa polos y ceros del Observador.	105
Figura 59. Diagrama de bloques del sistema completo.	106
Figura 60. Modelo del sistema de suspensión en Simulink.	106
Figura 61. Interfaz para la edición de parámetros de simulación.	107
Figura 62. Trayectorias básicas.	108
Figura 63. Trayectorias circulares y de cambio de carril.	109
Figura 64. Perturbaciones en el camino.	110
Figura 65. Definición del directorio de MATLAB.	112
Figura 66. Respuesta al comando rti1006.	113
Figura 67. Respuesta al comando rti.	113
Figura 68. Configuración de los parámetros RTI.	114
Figura 69. Configuración de parámetros de optimización.	114
Figura 70. Compilado del modelo.	115
Figura 71. Conexión con ControlDesk.	116
Figura 72. Cargar el archivo del modelo al procesador DSPACE.	117
Figura 73. Creando la interfaz.	117

Figura 74. Variables del modelo.	118
Figura 75. Variables del modelo.	118
Figura 76. Propiedades de los ejes.....	119
Figura 77. Interfaz finalizada.....	119
Figura 78. Captura de datos.	120
Figura 79. Interfaz DSPACE FINAL.	120
Figura 80. Distribución de las llantas.	121
Figura 81. Test [1] Trayectoria de la simulación.....	123
Figura 82. Test [1] Velocidad y aceleración del vehículo.	123
Figura 83. Test [1] Fuerzas en las llantas 1 y 3.....	124
Figura 84. Test [1] Variación angular del vehículo.	125
Figura 85. Test [1] Deslizamiento horizontal y lateral.....	125
Figura 86. Distribución de carga en el vehículo.	126
Figura 87. Test [1] Suspensión y ABC.	126
Figura 88. Test [2] Trayectoria de la simulación.....	127
Figura 89. Test [2] Velocidad y aceleración del vehículo.	128
Figura 90. Test [2] Fuerzas en las llantas 1 y 3.....	128
Figura 91. Test [2] Variación angular del vehículo.	129
Figura 92. Test [2] Deslizamiento horizontal y lateral.....	129
Figura 93. Test [2] Presión de frenado.....	130
Figura 94. Test [2] Suspensión y ABC.	130
Figura 95. Distribución de carga durante el frenado.....	131
Figura 96. Test [3] Trayectoria de la simulación.....	131
Figura 97. Test [3] Velocidad y aceleración del vehículo.	132
Figura 98. Test [3] Fuerzas en las llantas 1 y 3.....	132

Figura 99. Test [3] Variación angular del vehículo.	133
Figura 100. Test [3] Deslizamiento horizontal y lateral.	133
Figura 101. Test [3] Presión de frenado.	134
Figura 102. Test [3] Suspensión y ABC.	134
Figura 103. Test [4] Trayectoria de la simulación.	135
Figura 104. Test [4] Velocidad y aceleración del vehículo.	136
Figura 105. Test [4] Fuerzas en las llantas 1 y 3.	136
Figura 106. Test [4] Variación angular del vehículo.	137
Figura 107. Test [4] Deslizamiento horizontal y lateral.	137
Figura 108. Test [4] Presión de frenado.	138
Figura 109. Test [4] Suspensión y ABC.	138
Figura 110. Test [5] Trayectoria de la simulación.	139
Figura 111. Test [5] Velocidad y aceleración del vehículo.	139
Figura 112. Test [5] Fuerzas en las llantas 1 y 3.	140
Figura 113. Test [5] Variación angular del vehículo.	140
Figura 114. Test [5] Deslizamiento horizontal y lateral.	141
Figura 115. Test [5] Suspensión y ABC.	141
Figura 116. Test [6] Trayectoria de la simulación.	142
Figura 117. Test [6] Velocidad y aceleración del vehículo.	142
Figura 118. Test [6] Fuerzas en las llantas 1 y 3.	143
Figura 119. Test [6] Variación angular del vehículo.	143
Figura 120. Test [6] Deslizamiento horizontal y lateral.	144
Figura 121. Test [6] Suspensión y ABC.	144
Figura 122. Resultados obtenidos en DSPACE, modelo GO-KART.	146
Figura 123. Resultados obtenidos en DSPACE, modelo complejo.	147

Figura 124. Resultados obtenidos en DSPACE, modelo complejo.147

Figura 125. Visualización de la trayectoria del vehículo.149

RESUMEN

El proyecto consiste en la implementación de las técnicas de simulación “*Hardware-in-the-loop* y *Software-in-the-loop (HIL&SIL)*” en un modelo de vehículo para realizar pruebas en tiempo real de factores externos que afectarían el comportamiento del modelo. Este modelo de vehículo consta de subsistemas como dinámica del vehículo (Aceleración, control de giro, elevación, rotación), modelo de llanta, sistema de frenos con componentes hidráulicos junto con un sistema antibloqueo de frenos, sistema de suspensión junto con control activo del chasis, respuesta de los neumáticos del vehículo ante un rango de frecuencias de excitación causadas por el estado de la vía y diferentes maniobras de manejo que el conductor realizaría en determinadas circunstancias. Para esto se utilizará la tarjeta de procesamiento DS 1006 desarrollada por la empresa DSPACE, la cual está diseñada específicamente para realizar aplicaciones en tiempo real de alta gama y que requieran de gran potencia de cálculo, esta tarjeta será la encargada de calcular los modelos de simulación. Además, una computadora de sobremesa será utilizada como plataforma para realizar el modelado (SIMULINK) y como interfaz entre el usuario y el procesador en tiempo real anteriormente mencionado.

PALABRAS CLAVES:

- **SIMULACIÓN**
- **TIEMPO REAL**
- **MODELOS NO LINEALES**
- **DINÁMICA DE VEHÍCULOS**

ABSTRACT

This project consists in the implementation of “Hardware and Software in the Loop” simulation techniques in vehicle models for development, testing and validation of components like electronic control units (ECU) and other embedded real-time systems, this can be achieved by simulating a mathematical model of the dynamical system or the plant that will be controlled with the ECU, using the same inputs, outputs and behavior in order to verify and test in a certain way every possible situation. These vehicle models will have systems like vehicle dynamics (Acceleration, yaw-roll-pitch control), tire model, hydraulic brake system with an anti-lock braking system, suspension system with active body control and testing of vehicle’s wheel behavior to a range of excitation frequencies caused by road state and some driver maneuvers along that road. For this, the DSPACE DS1006 processing board will be used. This board was designed for high-end real time applications which require great computer power. In addition, a host PC will be used for the modelling (Simulink) and as interface between the user and the real-time processing board.

KEYWORDS:

- **SIMULATION**
- **REAL TIME**
- **NOLINEAL MODELS**
- **VEHICLE DYNAMICS**

CAPÍTULO 1

1. GENERALIDADES

1.1. ANTECEDENTES

Una de las fuentes principales del saber se encuentra en las universidades especialmente en los centros de investigación, los cuales concentran sus esfuerzos en crear conocimiento e innovación tecnológica logrando una transferencia del conocimiento rápida y provechosa, perfilando a las universidades como un pilar en la modernización y desarrollo de la industria nacional.

El Departamento de Ingeniería de la Universidad de Ciencias Aplicadas de Brandeburgo se caracteriza por la búsqueda y desarrollo de proyectos innovadores basados en las nuevas tecnologías, y que supongan un desafío para sus estudiantes tanto regulares como internacionales, desarrollando prototipos y equipos que representen los conocimientos aprendidos para nuevas tecnologías y soluciones a problemas en el sector empresarial en Alemania y a nivel mundial.

Debido a la relación Interinstitucional entre la Empresa SIEMENS y la Universidad de las Fuerzas Armadas - ESPE se genera el lazo de cooperación entre la Universidad de Ciencias Aplicadas de Brandeburgo y la ESPE. En este marco surge el interés por parte de la institución alemana para que estudiantes de las carreras de Ingeniería Electrónica y Mecatrónica desarrollen proyectos de investigación en sus instalaciones. El desarrollo de nuevos vínculos y conexiones a nivel internacional es una actividad relevante para la universidad porque amplían el campo profesional para los estudiantes, que deban demostrar sus conocimientos y capacidades que la Universidad imparte, dando renombre y categoría a la misma.

La Universidad de las Fuerzas Armadas – ESPE no ha desarrollado trabajos de titulación con temáticas afines al presente proyecto por lo que esta

investigación sería pionera en su campo, esto creará las bases necesarias para que futuros trabajos de titulación amplíen esta investigación y encuentren otras aplicaciones basándose en las líneas de investigación de la universidad y en los sectores estratégicos para el cambio de la matriz productiva impulsada por el gobierno.

1.2. OBJETIVOS

1.2.1. GENERAL

Implementar las técnicas de simulación “Hardware y Software in-the-loop” para realizar pruebas en tiempo real de un modelo de vehículo utilizando la plataforma DSPACE.

1.2.2. ESPECÍFICOS

- Modelar matemáticamente los sistemas y subsistemas presentes en el vehículo relevantes al tema de investigación.
- Implementar el modelo matemático del vehículo en Simulink.
- Realizar la comunicación entre la interfaz de usuario (PC – Control Desk) con el hardware DS1006.
- Transferir el modelo del vehículo a la plataforma DSPACE y comparar parámetros de rendimiento.
- Realizar las correcciones necesarias de acuerdo a los fallos encontrados en el sistema para obtener el óptimo funcionamiento del proyecto.

1.3. JUSTIFICACIÓN E IMPORTANCIA

Este proyecto se convertirá en una gran herramienta para el desarrollo de nuevos prototipos e investigaciones concernientes al campo de la mecatrónica ya que está directamente relacionado con los objetivos del Departamento de la Energía y Mecánica y, además, está incluido en varias líneas de

investigación que la Universidad de las Fuerzas Armadas - ESPE cuenta dentro de su sistema de investigación; Entre las más relevantes se puede mencionar: **Automática y Control, Sistemas Eléctricos, Electrónicos y computacionales.**

Otros resultados que se obtendrían con esta investigación son:

- Generar conocimiento específico sobre la utilización de técnicas de simulación y modelamiento matemático de plantas para realizar ensayos y pruebas de tarjetas controladoras.
- Crear bases tecnológicas para el desarrollo de proyectos relacionados en las líneas de investigación definidas por la universidad.
- Contribuir a la actualización del conocimiento científico en las áreas de automática, control, sistemas eléctricos, electrónicos y computacionales.
- Crear los ambientes y las condiciones necesarias para el desarrollo y puesta en marcha de proyectos multidisciplinarios relacionados.

El desarrollo de este proyecto de investigación en la Universidad de Ciencias Aplicadas de Brandeburgo no solo permitirá compartir conocimiento entre dos instituciones académicas, también permitirá contribuir a los índices de movilidad internacional de la Universidad, así como la relación bilateral académica con Alemania, sentando un precedente para la futura incorporación de estudiantes de nuestra universidad en universidades alemanas.

CAPÍTULO 2

2. MARCO TEÓRICO

2.1. INTRODUCCIÓN A HARDWARE Y SOFTWARE IN-THE-LOOP

La búsqueda e investigación de nuevas tecnologías es el objetivo fundamental de la ciencia en la actualidad. Estas tecnologías permiten desarrollar productos o equipos que mejoren la calidad de vida de las personas asegurándose que el impacto ambiental sea el mínimo, también tienen como objetivo hallar soluciones a problemas en la producción y la industria, optimizando procesos, reduciendo costos y tiempos.

Los **sistemas embebidos** son sistemas electrónicos cuyo objetivo es realizar funciones específicas en tiempo real, diseñados para controlar desde equipos cotidianos como teléfonos celulares, electrodomésticos hasta sistemas complejos como vehículos, satélites, transbordadores espaciales, vehículos aéreos no tripulados (UAV), sistemas de defensa, vehículos marinos, motores de jet y muchas más. Estos generalmente requieren de un alto nivel de complejidad del controlador (sistema embebido) y mantener a la planta bajo control ante numerosas variables.

Una de las aplicaciones más utilizadas de sistemas embebidos es la **Unidad Electrónica de Control (ECU)** presente en la mayoría de vehículos, este dispositivo controla los sistemas o subsistemas eléctricos del vehículo. Algunos tipos de ECU son: Módulo de control de potencia, suspensión, transmisión, motor, velocidad, Módulo de electrónica, Módulo de control de suspensión, Módulo de control motor, Módulo de control de velocidad, Módulo ABS, Módulo de control electrónico de dirección. Actualmente algunos vehículos tienen hasta 80 ECUs siendo estas cada vez más complejas y sofisticadas por lo que es necesario que su desarrollo sea en menos tiempo y utilizando la menor cantidad de recursos posibles, en este punto entra la **simulación** como herramienta para optimizar el desarrollo de las ECUs (véase **Figura 1**).

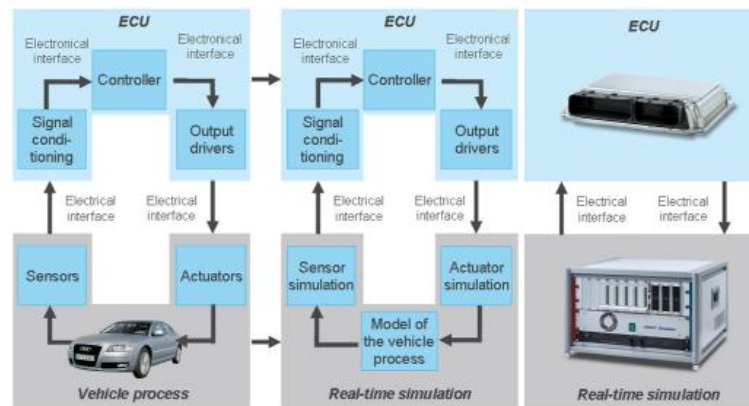


Figura 1. Diagrama de bloques de una aplicación HIL&SIL.

FUENTE: (dSPACE GmbH, 2015)

Debido al aumento en la capacidad de cálculo en los ordenadores, la simulación se está convirtiendo en una herramienta cada vez más demandada por los ingenieros que quieren evaluar y verificar sus diseños. Las herramientas de simulación que hace años requerían de potentes ordenadores, hoy pueden llevarse a cabo en computadoras personales ordinarias. En el área de diseño, desarrollo y pruebas *de sistemas embebidos* para productos existen algunos tipos de aplicaciones que sacan provecho a la capacidad de cálculo de las computadoras actuales, entre estas se destacan las técnicas de simulación **Hardware in-the-loop** y **Software in-the-loop (HIL&SIL)**.

2.2. BENEFICIOS DE HIL & SIL

El principal objetivo de las técnicas de simulación “HIL&SIL” es el de disponer de una plataforma efectiva para desarrollar y probar sistemas embebidos en tiempo real, esto se consigue al simular (*mediante un modelo matemático de los sistemas dinámicos*) la planta/equipo (*motor, transmisión, suspensión, neumáticos e inclusive todo el vehículo*) que se pretende controlar con la ECU, utilizando para ello las mismas entradas/salidas y comportamiento, de manera que el test verifique de forma segura todas las

posibles situaciones (*frente al modelo de controlador en la fase de diseño y frente al controlador final en la fase de prototipado*).

Dichas características se traducen en algunos beneficios siendo los más importantes:

- **Notable disminución en los costos de desarrollo:** Esto se logra minimizando el número de prototipos necesarios y minimizando el número de pruebas o ensayos con la planta real que usualmente son muy costosos.
- **Reducción en el tiempo de desarrollo:** Estas técnicas permiten el desarrollo en paralelo entre el producto y las pruebas o ensayos necesarios, reduciendo el tiempo de desarrollo de manera sustancial. Esto es posible debido a que las ECUs pueden ser probadas en una etapa temprana cuando todavía no se cuenta con un equipo físico (motor real) resultando en grandes avances de desarrollo.
- **Seguridad:** Al usar simulaciones para realizar las pruebas de componentes críticos de los productos se evitaría que ocurriesen fallas con pérdidas humanas o materiales, como es el caso de ensayos de prototipos de controles de vuelo que resultarían en un accidente aéreo de grandes magnitudes.
- **Visualización 3D:** Es posible generar una visualización en 3D del objeto de estudio/planta permitiendo una mejor apreciación del entorno y sus parámetros.
- **Testeo de condiciones ambientales difíciles y reproducibilidad de ensayos:** Test de lluvia, altas y bajas temperaturas y de invierno pueden ser conseguidos cambiando simplemente los parámetros en el programa.

2.3. Aplicaciones generales

Simulaciones HIL & SIL han sido utilizadas en diferentes campos de ingeniería, desde aeronaves, industria aeroespacial, industrial automotriz, aplicaciones para radares hasta en robótica y sistemas de potencia (Sarhadi & Yousefpour, 2014). La industria aeroespacial ha utilizado estas técnicas desde 1960 para simulación de cabinas de pilotos de aviones, desarrollo de software de naves (Apollo), diseño de controladores de tolerancia de fallos los cuales ayudan al piloto a experimentar condiciones de vuelo reales e inclusive para simular vehículos aéreos no tripulados (UAVs) y control de satélites.

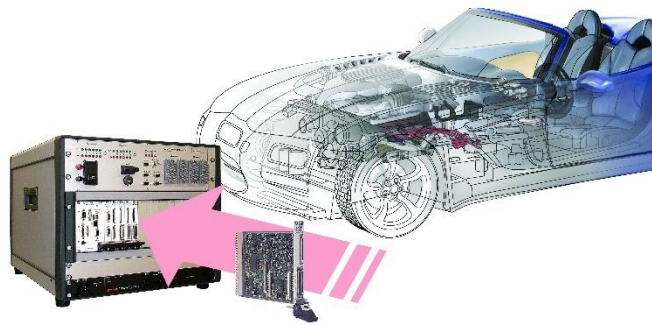


Figura 2. Aplicaciones en vehículos.

FUENTE: (National Instruments, s.f.)



Figura 3. Aplicaciones en la industria aeroespacial.

FUENTE: (Popular Science, s.f.)

2.4. PRINCIPALES CAMPOS DE APLICACIÓN DE HIL&SIL

La industria automotriz fue pionera en el desarrollo y uso de simulaciones para validar sus productos, actualmente estas técnicas están siendo utilizadas en el desarrollo de las ECUs y también en el desarrollo de software de control y asistencia para vehículos. Entre los principales campos de aplicación se tiene:

2.4.1. Desarrollo de subsistemas de vehículos.

- Control de estabilidad electrónico (ESP).
- Sistema anti-bloqueo de frenos (ABS).
- Sistema de control de tracción (TCS).
- Frenado electrónico para parqueo.
- Dirección electrónica asistida.
- Dirección activa/adaptativa.

2.4.2. Diseño y desarrollo de unidades de control electrónicas (ECU)

- Unidades de control electrónicas para motores a combustión. (Waltermann, 2009)
- Unidades de control electrónicas para subsistemas relacionados con la dinámica del vehículo.

2.4.3. Aplicaciones en vehículos eléctricos

- Vehículos híbridos y control eléctrico.
- Sistema de manejo de energía
- Estrategia de control.
- Tecnología de regeneración por frenos.

2.4.4. Manejo de potencia en vehículos

- Electrónica de potencia avanzada en vehículos eléctricos e híbridos.

- Acciones de conmutación 10KHz. Simuladores en tiempo real suficientemente rápidos, en el rango de los microsegundos.

2.4.5. Aplicaciones varias

- Análisis de accidentes
- Cinemática y dinámica del chasis y de la dirección.
- Dinámica del vehículo y confort durante el viaje.

2.5. COMPONENTES DE SISTEMAS HIL

2.5.1. Modelo de la planta

El modelo matemático del sistema es conocido como “modelo de la planta”. Estos modelos pueden ser desarrollados en base a una descripción física de los componentes o elementos que la componen (véase **Figura 4**). Esta descripción puede ser expresada por: ecuaciones basadas en leyes físicas, método llamado *Modelamiento teórico* y también por el método *Modelamiento Experimental*, el cual se basa en datos e información de pruebas y ensayos de los componentes. Actualmente los modelos de planta son considerados híbridos ya que se basan en la descripción física de los componentes, pero son validados a través de experimentos o complementados con datos de pruebas. (Zhang & Mi, 2011)

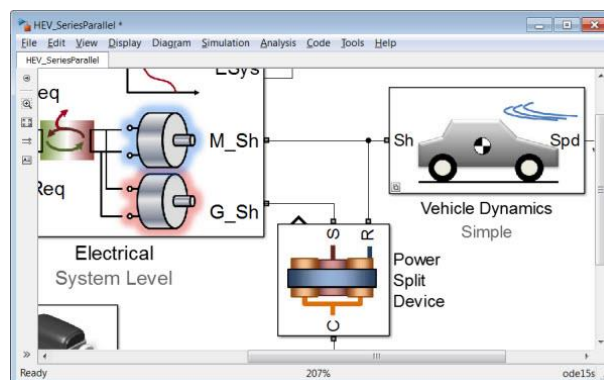


Figura 4. Modelado de subsistemas de un vehículo.

FUENTE: (Mathworks, s.f.)

2.5.2. Sensores

Son necesarios para la retroalimentación de la planta, sensores simulados y sensores reales suelen ser utilizados en simulaciones HIL ya que el modelo de la planta puede ser puramente matemático o puede incluir hardware adicional. Las salidas de los sensores tienen conexión con las entradas análogo/digitales de las ECUs. (Zhang & Mi, 2011)

2.5.3. Computadora en tiempo real

Con el fin de cumplir los requisitos de simulaciones HIL & SIL, la computadora necesita ser lo suficientemente potente desde un punto de vista de rendimiento para permitir un procesamiento en tiempo real. Una **simulación en tiempo real** se refiere a un modelo de un sistema que puede ser ejecutado y procesado en el mismo tiempo que lo haría el sistema real; se denomina **simulación offline** cuando la simulación se ejecuta en menor o mayor tiempo que el sistema.

Algunas de las características que una computadora en tiempo real puede tener son las siguientes:

- Computadoras en tiempo-real pueden estar incluidas dentro de un computador anfitrión o pueden ser independientes y estar conectadas con el computador.
- La computadora embebida puede tener puertos digitales y analógicos para la comunicación con la ECU y puede tener un enlace de conexión con el computador anfitrión.

2.5.4. Cargas

La ECU puede controlar cargas, entre los cuales se tiene: actuadores, válvulas, motores eléctricos, relés, etc. Pueden usarse en un sistema HIL cargas reales o circuitos eléctricos equivalentes. El sistema de diagnóstico de la ECU monitoriza estos actuadores en caso de que ocurra alguna falla (corto circuito, circuito abierto) y también informa al conductor (Waltermann, 2009).

Dependiendo del tipo de instalación, las cargas del sistema HIL pueden ser simulados o reales. Las cargas no solo deben simular los niveles de energía y de potencia de una carga real, sino también deben representar el mismo proceso transitorio de la carga real. (Zhang & Mi, 2011)

2.5.5. Computadora anfitrión

Un computador anfitrión con especificaciones media es generalmente utilizado para el desarrollo del modelo. Este computador tiene como tarea la: compilación, colección, almacenaje y visualización de datos mediante una interfaz gráfica de usuario (GUI) (Zhang & Mi, 2011).

Normalmente se utilizan tarjetas de interfaz optimizadas y varios protocolos como el Gigabit Ethernet para asegurar una alta velocidad de transmisión de información y una baja latencia en la comunicación con el hardware en tiempo real (Waltermann, 2009).

2.5.6. Diagnóstico de la ECU

Diagnóstico de fallas y errores inducidos por el operador son efectuados durante las pruebas o ensayos de la ECU.

2.5.7. Unidad de suministro de energía.

Dependiendo de la aplicación, componentes de la batería son simulados para proveer de energía a la ECU durante las pruebas, esto es importante para pruebas que requieran alto o bajo voltaje y también para simular caídas de voltaje que suscitan en diversas condiciones, como por ejemplo al momento de arrancar el motor.

2.5.8. Unidad de inserción de fallas (FIU)

Posibilidad de insertar fallas eléctricas para diferentes escenarios.

2.6. TARJETA PROCESADORA DSPACE DS1006

Como ya fue mencionado, es necesario hardware potente para cumplir los requisitos de procesamiento necesarios de simulaciones HIL&SIL en tiempo real. **La tarjeta procesadora DS1006** elaborada por la empresa DSPACE es usada frecuentemente para aplicaciones de gama alta en tiempo real debido a su capacidad de calcular complejos y detallados modelos que requieren de una capacidad de cálculo elevada. El poder de procesamiento y la capacidad de puertos de entrada y salida (I/O) puede ser mejorada aún más al poder enlazar hasta 20 tarjetas procesadoras formando un sistema multiprocesador (dSpace, s.f.)



Figura 5. DS1006 Vista frontal.



Figura 6. DS1006 Vista posterior.

En dichas simulaciones, la tarjeta DS1006 provee la capacidad de cálculo requerida por la simulación en tiempo real y también sirve como interface con tarjetas de adquisición de datos y con el computador anfitrión, se pueden utilizar módulos especiales para entradas y salidas como: modulo digital, conversor digital-análogo, conversor análogo-digital e inclusive interfaz CAN y FLEXRAY. Este hardware también ofrece la posibilidad de realizar todos los modelos de la planta en el software MATLAB/SIMULINK de manera que aproveche los beneficios de un entorno robusto de programación gráfica (véase **Figura 7**) (Isermann, Schaffnit, & Sinsel, 1999).

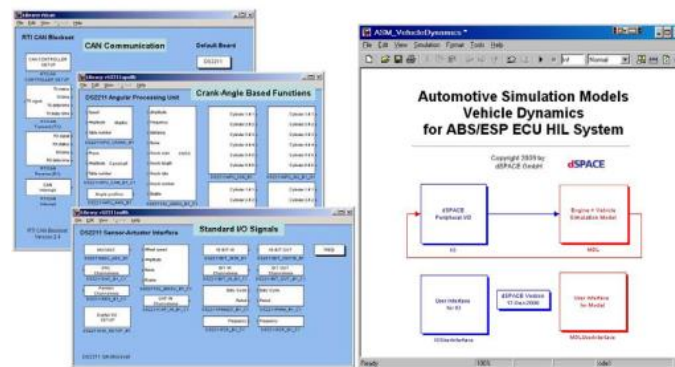


Figura 7. Librerías en Simulink para aplicaciones HIL&SIL.

FUENTE: (Waltermann, 2009)

2.6.1. Propiedades

- Quad-Core AMD Opteron Processor 2.8 GHz, tecnología x86. (dSPACE GmbH, 2015)
- Memoria local de 1GB.
- Interfaz RS-232 con estándar UART, permite transferencias de hasta 115.2 Kilo baudios.
- Conexión de alta velocidad con otros equipos con el bus periférico de alta velocidad (PHS). Este bus es una interfaz bidireccional de DSPACE para transferencias paralelas de 3 bits y tiene una velocidad de transmisión de 100Mbps. Es utilizado para establecer comunicación

entre las tarjetas procesadoras DS1005/1006 y las tarjetas de entradas y salidas DS2210/DS2211.

- Posibilidad de armar un sistema multiprocesador con hasta 20 tarjetas procesadoras conectadas entre sí. (dSpace, s.f.)
- Programación de la tarjeta desde Simulink utilizando la librería **Real Time Interface** (RTI).

2.6.2. Aplicaciones típicas

- Poder computacional para simulación de modelos en tiempo real con señales de entrada y salida. (dSPACE GmbH, 2015)
- Simulación de motores a combustión.
- Simulación de dinámica de vehículos.
- Simulación del tren motriz.

2.6.3. DSPACE ControlDesk software

La plataforma DSPACE complementa su suite de desarrollo con un software universal, modular y con herramientas de instrumentación para facilitar la simulación y el testeo. Este software va a ser utilizado en el [capítulo 5](#) como interfaz entre la computadora y el equipo DSPACE.

2.7. OBJETIVOS DESEADOS EN EL DESARROLLO DE VEHÍCULOS

Las técnicas de simulación HIL & SIL son utilizadas en el desarrollo de vehículos por los fabricantes de equipos originales (OEM) para realizar pruebas de sus productos. Existen una amplia variedad de pruebas que pueden ser realizadas en el desarrollo de vehículos. Algunos de los subsistemas de vehículo más relevantes que requieren de estas técnicas serán detallados a continuación:

2.7.1. Sistema de dirección asistida eléctricamente (EPS)

Este sistema mejora el control del vehículo al momento de cambiar de dirección en condiciones de manejo de alta y baja velocidad. Por ejemplo: el sistema de dirección asistido reduce la fuerza necesaria por el conductor para girar el volante de un vehículo que circula a velocidad baja y en velocidades altas, la sensación de estabilidad del vehículo mejora al tomar una curva (Lee, Lee, K.S., S.K., & J.I, 2011).

Actualmente, sistemas de dirección hidráulicos están equipados con tuberías y que boquillas las cuales usan energía del motor para su funcionamiento. Por otra parte, los sistemas de dirección asistidos eléctricamente operan independientemente de la energía del motor ya que cuenta con su propio motor eléctrico y no necesitan de un complejo sistema. (Lee, Lee, K.S., S.K., & J.I, 2011)

Las ventajas de este sistema son:

- Fáciles de mantener
- Reducción en un 3 al 5 % de consumo de combustible lo que reduce la contaminación del aire y aumenta la eficiencia del vehículo.
- Fácil implementación.

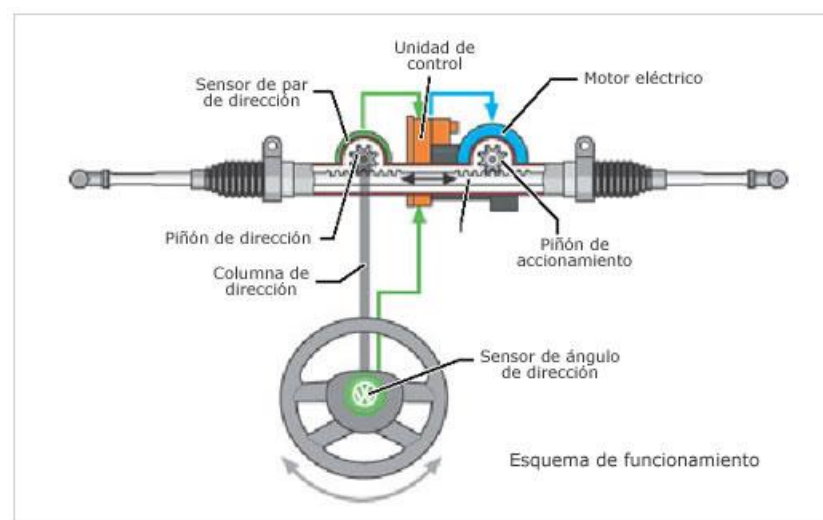


Figura 8. Sistema asistido de dirección.

FUENTE: (Aficionados a la mecánica, 2014)

En la referencia (Lee, Lee, K.S., S.K., & J.I, 2011), los investigadores desarrollaron un equipo mecánico, el cual consta de: un volante de mando, una barra de dirección y para la simulación un modelo de vehículo de 17 grados de libertad que simula la fuerza lateral causada por los neumáticos en una curva. Además de una ECU para el sistema de dirección con su respectivo algoritmo de control, el cual fue diseñado e implementado. La ECU también es un sistema de diagnóstico ya que la fuerza lateral presente en el neumático puede ser medida mediante sensores, esto detecta el esfuerzo al momento de girar el vehículo y un posible malfuncionamiento. La simulación HIL aplicada en el desarrollo de este sistema facilitó simulaciones repetitivas y redujo tiempo y costo.

2.7.2. Sistema de control de suspensión

El sistema de suspensión es un componente importante para un vehículo. Este sistema absorbe irregularidades y perturbaciones del terreno mientras que a la vez mejora el control del vehículo y el manejo. Por esta razón su desarrollo y testeado debe ser realizado con alta precisión.

Un ejemplo de una instalación HIL para testeado de sistemas de suspensión puede ser visto en la referencia (Misselhorn, Theron, & Els, 2006). En dicho estudio, una suspensión trasera de motocicleta fue comparada con un software de simulación. Esta instalación constó de un actuador hidrodinámico que causa una elongación o compresión según se requiera, una computadora conectada a un equipo DSPACE que contiene un modelo matemático y conectores entrada y salidas para sensores y actuadores.

Los investigadores a cargo de este estudio tienen pensado expandir el sistema HIL al implementar un banco de pruebas con mayor capacidad y expandir el modelo permitiendo que se realicen pruebas de cuatro suspensiones semi-activas al mismo tiempo.

2.7.3. Control de climatización de un vehículo

El sistema automático de aire acondicionado es uno de los múltiples sistemas mecatrónicos presentes en vehículos modernos de gama media-alta. Este moderno sistema incluye un número elevado de actuadores mecatrónicos distribuidos con sensores y un controlador central (ECU). El objetivo del sistema automático de aire acondicionado es, lograr una condición climática deseada en el interior del vehículo, esta condición está definida por la temperatura interior, la humedad y la velocidad del flujo de aire que circula por el vehículo y los pasajeros (Michalek, Gehsat, Trapp, & Bertram, 2005).

Los métodos de simulación HIL & SIL son usados para verificar y validar los componentes del sistema y el software. Una ventaja de usar estos métodos es que el ambiente es independiente del lugar de trabajo, esto significa que se pueden obtener diferentes condiciones climáticas logrando pruebas de manejo reproducibles. Es necesario que los modelos puedan simular: el flujo de aire, calentamiento, ventilación y aire acondicionado. Componentes del aire acondicionado y algoritmos de control complejos son necesarios para el ajuste de las condiciones internas del vehículo las cuales son realizadas por el control climático, este controlador puede ser testeado y optimizado en conjunto usando una computadora en tiempo real que está conectada al controlador vía las interfaces apropiadas.

En la referencia (Michalek, Gehsat, Trapp, & Bertram, 2005), los investigadores utilizaron el equipo DSPACE para la simulación HIL. Los modelos del aire acondicionado fueron diseñados en Simulink y luego transferidos a la computadora en tiempo real. Este experimento tiene como variables de control: la temperatura del evaporador, la temperatura de la boquilla de salida y finalmente la temperatura interior.

2.7.4. Sistema activo de control de roll (ARC)

Este sistema está basado en la manipulación hidráulica de las barras antivuelco del automóvil. El principal objetivo de este sistema es el de reducir el ángulo de roll del vehículo durante maniobras semi estacionarias. Esto lleva a una mejora en la sensación de confort transmitida hacia los pasajeros. Con este sistema es posible obtener una reducción en el ángulo de deslizamiento lateral del chasis y una reducción en las oscilaciones del chasis por el movimiento de yaw, durante maniobras de manejo como cambio de dirección o de carril (Sorniotti, Morgando, & Velardocchia, 2006).

Estas características pueden ser obtenidas mediante la variación dinámica de la rigidez de roll entre los dos ejes del vehículo. Por ejemplo, el sistema ARC puede ser implementado usando un actuador rotacional en medio de la barra antivuelco o mediante la sustitución de la barra que conecta al sistema de suspensión por un actuador lineal. Es importante mencionar que la actuación puede ser hidráulica o electro-mecánica.

En la referencia (Sorniotti, Morgando, & Velardocchia, 2006), los investigadores se enfocaron en el desarrollo de un sistema hidráulico ARC en base a un actuador lineal, en el cual fueron usados: una unidad de desplazamiento por bombeo de bajo volumen y un acumulador hidráulico como componentes hidráulicos del banco de pruebas. Este test reproduce las dinámicas de vuelco de un vehículo para verificar el rendimiento del sistema. Una idea de este concepto puede ser observada en la Figura 9.

Un modelo matemático del vehículo simula la dinámica, este calcula un ángulo de roll de referencia para el banco de pruebas, lo deriva de las fuerzas medidas por sensores y luego es comparado con el ángulo de roll efectivo medido por un potenciómetro. Un sistema de control basado componentes hidráulicos permite cambiar a un ángulo de roll deseado. Mientras tanto, el modelo de vehículo envía señales de entrada como el ángulo del volante, la aceleración lateral, la variación del ángulo de yaw del chasis al controlador, el cual calcula un torque antivuelco y envía esta señal de control a los actuadores.

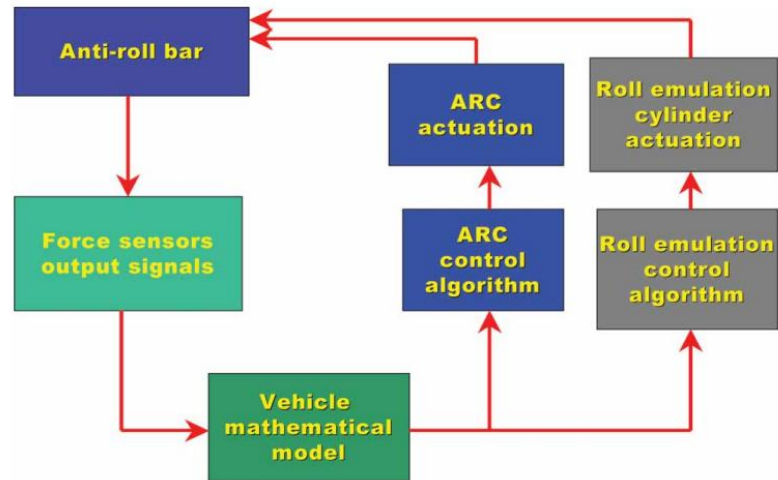


Figura 9. Esquema de un concepto de sistema HIL.

FUENTE: (Sorniotti, Morgando, & Velardocchia, 2006)

2.7.5. Sistema de control de estabilidad.

Situaciones de manejo inesperadas o una carga asimétrica en el vehículo como fuerzas laterales debido al viento y pérdida unilateral de presión en el neumático pueden llevar al vehículo a perturbaciones que necesitarían de una rápida compensación y estabilización. En condiciones en tiempo real, los conductores no podrían responder a tiempo a estos cambios, esto hace necesario que el vehículo tenga un control automático de estabilidad de yaw y lateral. Este control de estabilidad compensa las acciones del conductor durante una reacción de pánico generando una corrección instantánea.

Actualmente soluciones existentes para resolver este problema involucran el uso de tecnologías “**brake-by-wire**” las cuales proveen mejoras en, la estabilidad del vehículo, la distribución de fuerzas de frenado y también mejoran la capacidad del sistema de frenos. Un ejemplo de esto se presenta en el frenado de llantas mediante el sistema antibloqueo de frenos usando controladores de individuales por llanta.

Un simulador HIL en tiempo real fue desarrollado por (Guvenc, Guvenc, & Karaman, 2009), en este experimento un modelo completo de vehículo fue implementado para probar maniobras extremas y el comportamiento del

controlador de estabilidad. Finalmente, el controlador de estabilidad fue probado en campo en una pista de pruebas.

2.7.6. Sistema de control del motor

En la referencia (Isermann, Schaffnit, & Sinsel, 1999), se implementó un simulador en un banco de pruebas para la investigación de un sistema de manejo para el motor de un camión Mercedes-Benz. El simulador HIL probó ser una herramienta eficiente para desarrollar nuevas funciones de control, probar el software y el hardware del controlador del motor y para investigar el comportamiento de este en presencia de fallas.

Otra aplicación de HIL&SIL en motores fue implementada por la Universidad de Parma y el Centro de Desarrollo de Fiat, en la se desarrolló un sistema HIL basado en una computadora de sobremesa para la simulación de motores turbo cargados a diésel. El modelo simula la dinámica longitudinal del vehículo (Gambarotta, Ruggiero, Sciolla, & Lucchetti)

2.7.7. Control de cruceo adaptativo (ACC)

ACC es una funcionalidad bien difundida en vehículos modernos. Su objetivo es el de mantener el vehículo en movimiento siguiendo una velocidad determinada por el conductor. El sistema automáticamente adapta la velocidad en presencia de vehículos cercanos usando el sistema de frenos y el acelerador. Un radar es usado para detectar otros vehículos midiendo distancias y la velocidad relativa entre estos (véase la Figura 10). Algunas características que este sistema ofrece al usuario son: seguridad, confort, economía del combustible, eficiencia en condiciones de tráfico.

En la referencia (Naus, Ploeg, Van de Molengraft, Heemels, & Steinbuch, 2010), se aplicaron una serie de diferentes escenarios para validar un controlador. Este controlador fue implementado en un Audi S8, el cual fue equipado con un sistema electro-hidráulico de frenos facilitando el control

“**brake-by-wire**”, un dispositivo LIDAR con un rango de 150 metros también fue embebido al vehículo.

El sistema ACC fue implementado en una DSPACE AUTOBOX que incluye el controlador de la dinámica longitudinal del vehículo y una laptop que fue usada para el monitoreo continuo de las señales. Tanto simulaciones como experimentos de campo probaron condiciones de trabajo ideales.

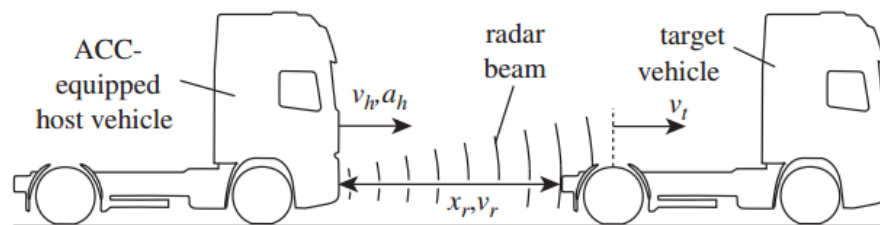


Figura 10. Aplicación de ACC.

FUENTE: (Naus, Ploeg, Van de Molengraft, Heemels, & Steinbuch, 2010)

2.7.8. Sistema avanzado de asistencia al conductor

Sistemas activos de seguridad como un ABS o un ESP son sistemas confiables, pero no suficientes si se los utiliza individualmente para asegurar a los ocupantes del vehículo en situaciones peligrosas. Debido a esto, es necesario el desarrollo de un sistema más complejo, robusto y completo.

El sistema avanzado de asistencia al conductor (ADAS) asiste al conductor mediante el reconocimiento de potenciales peligros utilizando sensores para el medir el entorno (radar, laser, visión) reduciendo significativamente el número de accidentes en las vías (Gietelink, Ploeg, DeSchutter, & Verhaegen, 2006). El sistema ADAS debe cumplir altos estándares en cuanto a rendimiento, confiabilidad y seguridad. El uso de simulaciones HIL&SIL en sistemas ADAS requiere la validez de los subsistemas del vehículo, siendo los más utilizados: ACC, stop and go, sistemas pre-choque, puntos ciegos

En la referencia (Gietelink, Ploeg, DeSchutter, & Verhaegen, 2006), un sistema ADAS en gran escala fue equipado en un vehículo para realizar pruebas y simulaciones en un entorno HIL. Un banco de potencia fue usado para emular la interacción con la carretera y vehículos robóticos para representar el tráfico. El sistema HIL mostró gran efectividad en la verificación del modelo, sintonización de los algoritmos de control y preparación de pruebas de manejo.

2.8. TIPOS DE ELEMENTOS DISPONIBLES EN SIMULINK

Simulink es un entorno gráfico programable desarrollado por MathWorks y está basado en MATLAB. Es una herramienta útil para modelado, simulación y análisis de sistemas dinámicos multi-dominio con aplicaciones en campos como: control automático., procesamiento de señales digitales y diseño basado en modelos (Mathworks, s.f.)

Una de la característica más conocida de Simulink es su compatibilidad con complementos y toolbox como: *Simulink Stateflow*, *Sim Event*, *Simulink Real-Time* que ofrece la posibilidad de trabajar con simulaciones y ensayos en tiempo real mientras que *Simulink Verification and Validation* ayudan al usuario a validar modelos.

2.8.1. Funciones S

Una función S es una descripción de un bloque de Simulink previamente escrito en MATLAB, permitiendo al usuario añadir código customizado al entorno. Puede ser escrito en lenguaje MATLAB, C, C++ o Fortran (Mathworks, s.f.). La función S puede ser compilada en archivos de tipo MEX y cargada en un bloque de Simulink, para lo cual es necesario que sean definas las siguientes funciones: (MathWorks, 1997)

Durante la inicialización

- Nombre de la función S
- mdlInitializeSizes. - Especifica el número de parámetros.
- mdlSampleTimes. - Obtiene el período de muestreo del equipo.
- mdlInitializeConditions. - Lee los valores de la interfaz.

Calculo de la salida

- mdlOutputs. - Permite al programa comunicarse con las entradas y salidas del hardware.

Resetear el hardware cuando el programa se termina

- mdlTerminate

2.8.2. Funciones de código de MATLAB

El bloque de función de MATLAB permite al usuario añadir funciones de MATLAB en el entorno Simulink. Esta capacidad es importante para codificar algoritmos que son mejor expresados en lenguaje textual de MATLAB que en lenguaje gráfico de Simulink. Con el bloque de función de MATLAB es posible generar un código C, C++ legible, eficiente y compacto (Mathworks, s.f.).

2.9. MODELAMIENTO DE SISTEMAS

El modelamiento de un sistema obtiene una descripción matemática-analítica de la planta permitiendo el estudio de sus aspectos relevantes y su influencia con otros componentes sin basarse en la experimentación, que en algunos casos no podría ser viable debido a factores externos como riesgos y costos. En otras palabras, un modelo de un sistema es una representación simplificada de un sistema físico real.

2.9.1. Definición de un sistema

Un sistema es una combinación de componentes que interaccionan e influyen entre sí para lograr un objetivo determinado. Los sistemas mecatrónicos más comunes usan sistemas mecánicos, neumáticos e hidráulicos. Un modelo matemático consiste en expresiones matemáticas que describen relaciones existentes entre magnitudes. Existen dos métodos fundamentales para desarrollar y describir el comportamiento dinámico de un sistema real utilizando modelos matemáticos (Bardini, Schramm, & Hiller, 2014).

- **Modelamiento teórico.** - Esta basado en leyes físicas expresadas por ecuaciones.
- **Modelamiento Experimental.** - Este método es utilizado cuando algunos parámetros de modelos teóricos con desconocidos o difíciles de medir (parámetros para modelamiento de llantas). Un modelo matemático puede ser usado como base y los parámetros individuales son identificados usando mediciones y pruebas.

Un modelo puede ser desarrollado de acuerdo a una complejidad y validez deseado, un modelo más complejo genera las siguientes suposiciones.

- Mayor precisión del comportamiento del sistema en las simulaciones.
- Las ecuaciones del modelo son más complejas.
- Mayor capacidad de cálculo del procesador.

Dicho esto, es importante evaluar si es necesario incrementar la complejidad de un modelo para obtener una mejor descripción del sistema.

2.10. DINÁMICA DE VEHÍCULOS

La dinámica de vehículos es un subtema de la mecánica que estudia el comportamiento de vehículos de carretera como: automóviles, camiones, buses con respecto a las causas que provocan o inducen una variación en el

movimiento del vehículo, algunas de estas causas son conocidas como aceleración, frenado, curvas y manejo.

Es importante conocer la parte analítica de un vehículo, especialmente en cómo y por qué fuerzas externas e internas son producidas; esto permite al investigador entender como pequeñas variaciones o cambios en el diseño o en las propiedades de un vehículo afectan su rendimiento bajo la consideración de las leyes físicas.

El uso de modelos de vehículos complejos para simular y desarrollar sistemas de vehículos y sus aplicaciones han ganado importancia en los últimos años (Bardini, Schramm, & Hiller, 2014).

El estudio de la dinámica de un vehículo abarca tópicos en ciencias, física, mecánica clásica, aerodinámica, etc. Algunos de los aspectos más importantes que se trabajarán y discutirán en este proyecto son:

- Diseño del automóvil.
- Control de estabilidad electrónico.
- Sistemas de dirección.
- Sistemas de suspensión.
- Sistemas de control de tracción.
- Dirección con geometría de Ackermann.
- Momentos de yaw.
- Roll y pitch.
- Distribución de carga.
- Masas con y sin amortiguamiento.
- Sobre viraje y sub viraje.
- Deslizamiento longitudinal y lateral.
- Variación de dirección.
- Carga en las llantas.

2.11. CONCEPTOS COMUNES EN DINÁMICA DE VEHÍCULOS

2.11.1. Ángulo de roll.

Es el movimiento de rotación lado a lado del vehículo con respecto al eje X debido a fuerzas centrífugas (véase **Figura 11**).

2.11.2. Ángulo de pitch.

Es el movimiento de rotación de adelante hacia atrás del vehículo con respecto al eje Y debido a fuerzas de aceleración o desaceleración (véase **Figura 11**).

2.11.3. Ángulo de yaw.

Es el movimiento de rotación del vehículo con respecto al eje Z debido a virajes (véase **Figura 11**).

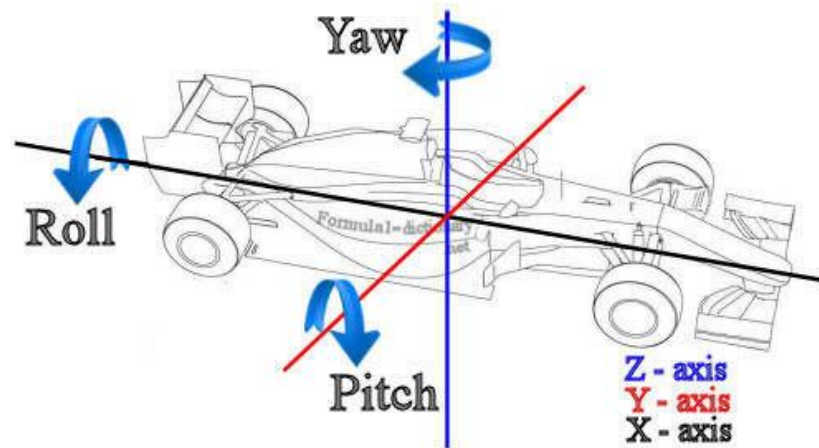


Figura 11. Movimiento del vehículo.

FUENTE: (Formula 1 Dictionary, s.f.)

2.11.4. Fuerza de tracción [F_u].

La fuerza de tracción o longitudinal es la fuerza que ejerce la calzada sobre la llanta a lo largo del eje X, es la encargada de acelerar o desacelerar el vehículo (véase **Figura 12**).

2.11.5. Fuerza lateral [F_s].

La fuerza de viraje o lateral es la fuerza que ejerce la calzada sobre la llanta a lo largo del eje Y debido a fuerzas centrífugas que ocurren en virajes (véase **Figura 12**). Esta fuerza también es considerada como la capacidad de la llanta a resistir el deslizamiento lateral en curvas (Racing Car Dynamics, 2014).

2.11.6. Torque de alineación [M_z].

Es el momento de la llanta con respecto al eje Z (véase **Figura 12**).

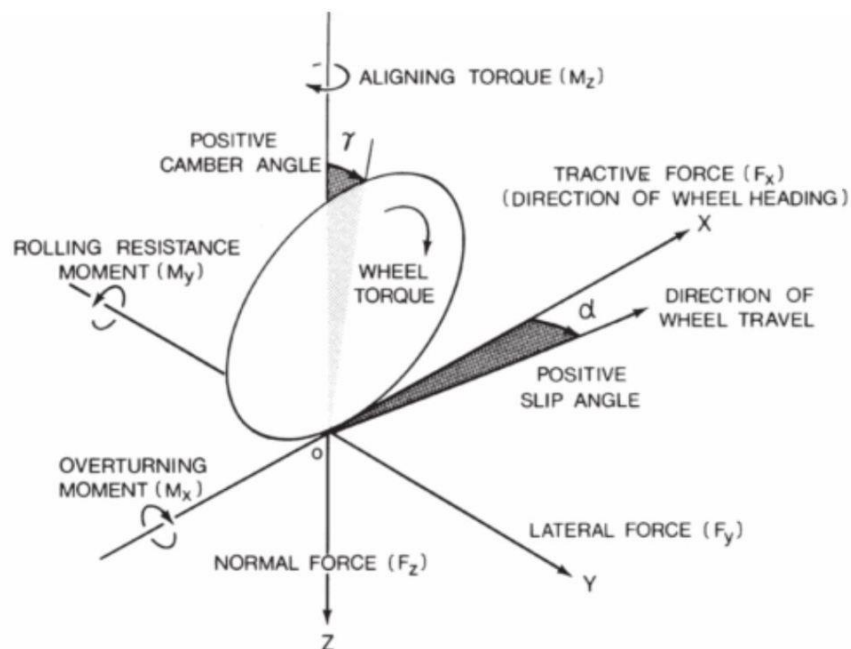


Figura 12. Fuerzas y momentos presentes en la llanta.

FUENTE: (Goldwing Facts, 2013)

2.11.7. Deslizamiento longitudinal.

Es la razón entre la velocidad rotacional de la llanta y la velocidad longitudinal del vehículo, su ecuación es:

$$slip = \frac{\omega \cdot R_d - v}{v} \quad (2.1)$$

Donde

ω : Velocidad angular de la llanta.

R_d : Radio efectivo de la llanta.

v : Velocidad longitudinal del vehículo.

Si el deslizamiento longitudinal es cero significa que la velocidad angular de la llanta es menor a la velocidad longitudinal del vehículo y la llanta se bloqueó.

2.11.8. Deslizamiento lateral.

Es el movimiento lateral de la llanta que ocurre cuando sus fuerzas laterales son mayores a la resistencia por fricción, por ejemplo, en curvas. Su ecuación es:

$$\alpha = \arctan\left(\frac{v_y}{v_x}\right) \quad (2.2)$$

2.12. MODELOS DE VEHÍCULOS

Algunos de los criterios para una lograr representación realística del comportamiento dinámico de vehículos son los siguientes:

- Cinemática y cinética espacial de todo el vehículo y subsistemas.
- Cinemática no lineal de la suspensión.
- Representaciones no lineales y dinámicas de las fuerzas.
- Fuerzas dinámicas de las llantas.

Tabla 1.

Grados de libertad de algunos modelos de vehículo

Tipo de modelo de vehículo	Grados de Libertad
"Single-track" lineal	2
"Single-track" no lineal	3 a 7
"Twin-track"	14 a 30
Sistema multicuerpo complejo	>20

Componentes mecánicos de un vehículo pueden ser modelados y simulados usando los siguientes enfoques:

- **Sistemas multicuerpo (MBS):** Este enfoque es adecuado para modelos complejos que describen la dinámica del vehículo.
- **Método de elementos finitos (FEM) y Sistemas continuos (COS):** Este enfoque es adecuado para sistemas mecánicos o cuerpos con una elasticidad distribuida equitativamente.

Un modelo completo de vehículo puede consistir de los subsistemas, entradas y condiciones externas que se observan en la Tabla 2.

Tabla 2.

Subsistemas, entradas y condiciones de un modelo

Subsistemas	Entradas	Condiciones Externas
Chasis	Pedal de freno	Ambientales
Suspension	Ángulo del volante	Inclinación de la vía
Llantas	Posición de la palanca de cambios	Perturbaciones en la vía
Freno	Control de climatización	Viento frontal y lateral
Dirección		

2.12.1. Modelo “Single-track”

El modelo single-track o de bicicleta es una simplificación de un modelo completo de un vehículo, el cual tiene de 14 a 30 grados de libertad mientras que el modelo de bicicleta tiene 2 grados de libertad en el caso de un modelo lineal y de 3 a 7 grados de libertad si el modelo es no lineal (véase **Tabla 1**).

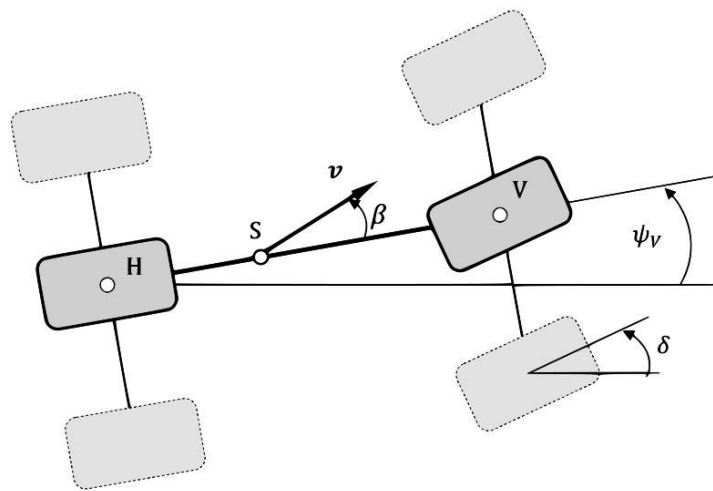


Figura 13. Modelo bicicleta.

FUENTE: (Bardini, Schramm, & Hiller, 2014)

El modelado está basado en las siguientes suposiciones (Bardini, Schramm, & Hiller, 2014):

- Los neumáticos frontales y traseros serán representados como uno solo por cada eje.
- La velocidad del centro de gravedad del vehículo será considerada constante a lo largo de su trayectoria.
- Levantamiento, roll y pitch no serán considerados. Lo que significa que el único movimiento angular posible sea el del ángulo de yaw.
- La masa del vehículo estará concentrada en el centro de gravedad S.
- La distribución de cargas en las llantas será constante entre el eje delantero y trasero.
- Fuerzas longitudinales en la llanta no serán consideradas.

Estas consideraciones permiten una descripción decente de la dinámica lateral del vehículo siempre y cuando hagan simulaciones en pista seca con una aceleración lateral de:

$$a_y \leq 0.4g \approx 4 \frac{m}{s^2}$$

2.12.2. Modelo “Twin-track”

El modelo twin track o de doble vía consiste usualmente de 14 grados de libertad sin contar con la cinemática de la suspensión, las llantas simplemente están conectadas al chasis del vehículo por un muelle y fuerzas de amortiguamiento. Este modelo es suficiente para una investigación inicial y para una simulación de un modelo de vehículo básico (Bardini, Schramm, & Hiller, 2014). La Figura 14 representa los grados de libertad de un modelo de doble vía en 3D.

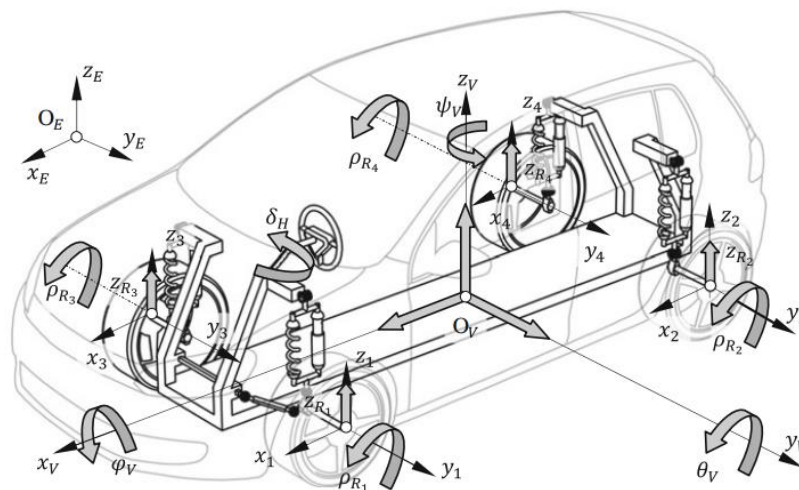


Figura 14. Modelo de doble vía.

FUENTE: (Bardini, Schramm, & Hiller, 2014)

2.12.3. Modelo completo de vehículo y sistemas multicuerpo

Puede ser difícil trabajar con modelos tridimensionales o con modelos complejos de más de 20 grados de libertad, por lo tanto, es necesario usar métodos que faciliten el modelado; uno de estos métodos es la *modularización del proceso de modelado de un vehículo completo*, esto significa que subsistemas mecánicos (suspensiones, ejes motrices, llantas) y subsistemas no mecánicos (elementos electrónicos, ECUs, controladores) deben ser modelados y preparados para que puedan ser vinculados en conjunto en un modelo de vehículo, por lo tanto, ecuaciones usuales de movimiento deben ser reemplazadas por un sistema mecánico que contenga cuerpos rígidos, uniones ideales y fuerzas externas, este método es llamado *sistema multicuerpo (MBS)*.



Figura 15. BMV serie 5.

FUENTE: (Bardini, Schramm, & Hiller, 2014)

Este método es utilizado en análisis estáticos, dinámicos y cinemáticos de sistemas mecánicos. El chasis del vehículo será modelado como un cuerpo rígido con masa, los subsistemas serán representados por cuerpos rígidos interconectados por uniones y elementos de fuerza. En la Figura 16 se puede observar este proceso de modelado.

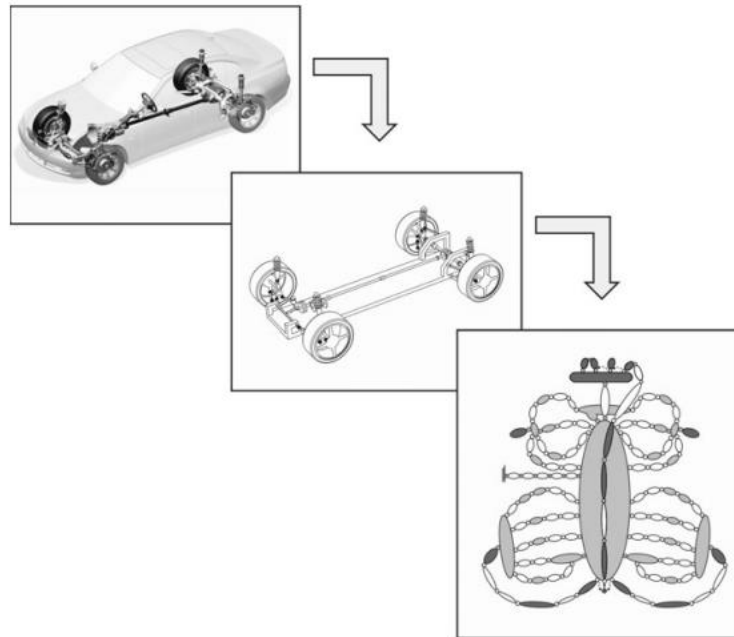


Figura 16. Proceso de modelado.

FUENTE: (Bardini, Schramm, & Hiller, 2014)

2.13. HERRAMIENTAS PARA CONSTRUIR MODELOS OFFLINE

Debido a la importancia del diseño basado en modelos (MBD) en la industria automotriz existe una amplia variedad de soluciones disponibles para construir modelos de sistemas reales. A continuación, se nombrarán las herramientas más comunes para la construcción de dichos modelos:

2.13.1. MSC ADAMS

MSC ADAMS es un software comercial para el modelado, simulación y verificación de sistemas multicuerpo. Fue desarrollado por la empresa MSC Software Corporation y es compatible con algunos lenguajes como Fortran y C++. MSC ADAMS admite Microsoft Windows y Linux. Esta herramienta es utilizada en el desarrollo de prototipos virtuales (VPD) y en la industria automotriz por sus avances en el estudio del rendimiento de suspensiones, manejo y conducción de modelos de vehículos complejos (MSC ADAMS, s.f.).

Este software tiene compatibilidad con CATIA, Inventor, Solidworks, IGES. Los complementos que ayudan al usuario en campos de investigación específicos son: ADAMS/Car, ADAMS/Control, ADAMS/Tire, ADAMS/View, ADAMS/Solver, ADAMS/Post-processor.

Modelos en ADAMS pueden ser enlazados a programas de control como MATLAB/Simulink para crear un sistema mecatrónico más sofisticado. Esto se logra conectando las variables de estado del modelo en ADAMS al sistema de control Simulink, ejemplos de conexiones entre ADAMS y Simulink pueden ser encontrados en las siguientes referencias (Anyu, Xiliang, & Shugen, 2011), (Yi, Min-min, Jin-yi, & Hu, 2010), (Ying, Yiming, & Hongni, 2010).

2.13.2. Simpack

SIMPACK es un programa comercial de simulación de multicuerpos para el análisis dinámico de sistemas mecánicos y mecatrónicos. Este programa genera modelos virtuales en 3D que preceden y visualizan el movimiento, fuerzas y tensiones (SIMPACK, s.f.). En un principio fue utilizado para soportar modelos no lineales complejos con cuerpos flexibles, pero actualmente puede soportar análisis de alta frecuencia y ser aplicados a cualquier rama de la ingeniería mecánica siendo los campos más importantes los siguientes:

- **Automóviles**

SIMPACK para automóviles es bien conocido en la industria automotriz. Esta herramienta ayuda a ingenieros y desarrolladores en el análisis y diseño de cualquier tipo de vehículo al tener una flexibilidad de modelado casi ilimitada. Una de sus principales ventajas es su base de datos completamente detallada y parametrizada de automóviles.

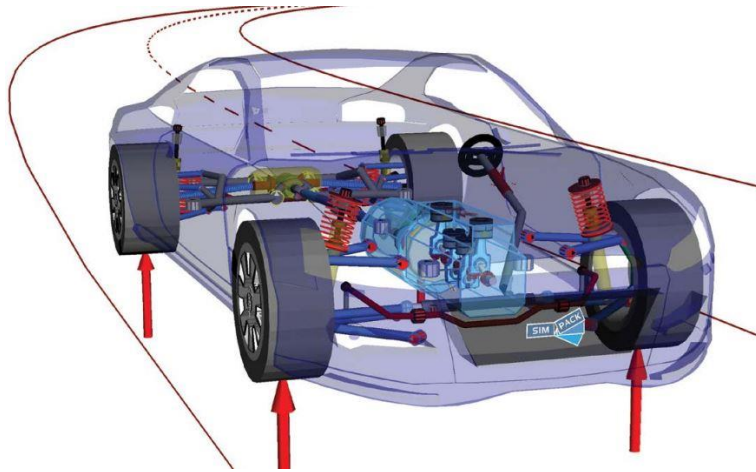


Figura 17. SIMPACK para automóviles.

FUENTE: (SIMPACK, s.f.)

- **HIL & SIL**

SIMPACK en tiempo real habilita el uso de complejos modelos para una amplia variedad de aplicaciones críticas en tiempo real en escenarios HIL & SIL. Estas aplicaciones incluyen simulaciones de manejo y testeo de ECU y componentes de bancos de pruebas (SIMPACK, s.f.).

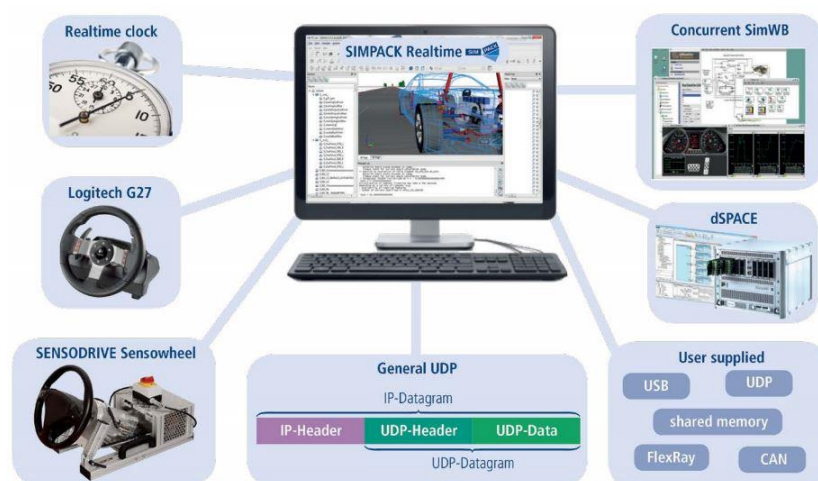


Figura 18. SIMPACK para HIL&SIL.

FUENTE: (SIMPACK, s.f.)

CAPÍTULO 3

3. IMPLEMENTACIÓN DE UN MODELO DE PRUEBA OFFLINE

3.1. INTRODUCCIÓN AL MODELO

Un modelo off-line será implementado en Simulink en un computador estándar, este modelo representará un **GO-KART** con control de dirección basado en un sistema de dos motores eléctricos ubicados en la parte trasera (véase **Figura 19**). El modelo será dividido en subsistemas, un bloque que contenga el modelo del vehículo y otro bloque que contenga el modelo del controlador; esto es para separar los subsistemas de un vehículo como se lo haría en un proceso HIL&SIL de producción real (ECU - sistema de frenos - sistema de suspensión - conductor - dinámica de chasis).



Figura 19. GO-KART real (Technische Fachschule Brandenburg).

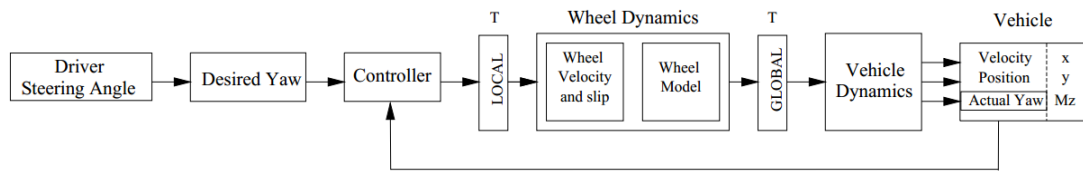


Figura 20. Diagrama del modelo de vehículo.

3.2. ESPECIFICACIONES DEL MODELO

El modelo de vehículo seguirá las siguientes consideraciones:

- Controlador de dirección.
- Modelo de los neumáticos.
- Siete grados de libertad¹.
- Tres trayectorias para pruebas.

3.3. BLOQUES DE FUNCIÓN EN SIMULINK

Una de las dificultades en la implementación de modelos en Simulink es la selección del **Bloque de Función**, este bloque permite al usuario ingresar funciones customizadas en Simulink con varias alternativas (funciones-S, funciones de Matlab, funciones-S en C) y cada una de estas alternativas tiene sus beneficios y contras. A continuación, se detallarán las diferentes versiones realizadas del modelo con respecto a los tipos de bloques de función de Simulink, con el objetivo de identificar el mejor tipo para esta aplicación.

3.3.1. Modelo versión 1 – Usando bloques de función de MATLAB

La manera más fácil de implementar funciones complejas con múltiples entradas y salidas es usando el bloque de **Función de MATLAB**. Este bloque está ubicado en *Librería de Simulink >> Simulink >> Funciones definidas por el usuario*, como se observa en la **Figura 21**.

¹ Este modelo de vehículo tiene 3 grados de libertad en el chasis y 4 grados de libertad de las ruedas.

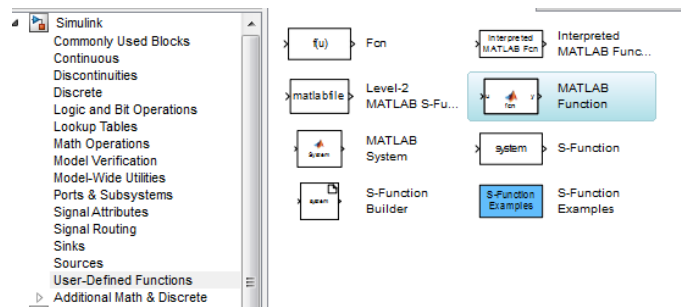


Figura 21. Bloque de función de MATLAB.

Con este bloque es posible escribir funciones y código de MATLAB para modelos en Simulink, esto es útil para aplicaciones que sean mejor descritas a través del lenguaje textual de MATLAB que el lenguaje gráfico de Simulink debido a su complejidad. Algunas ventajas de este bloque son:

- El manejo de información es más fácil.
- Permite la generación de código.
- Permite flexibilidad simplificando el proceso de modelado.
- Es posible llamar a funciones extrínsecas, las cuales no generan código y se ejecutan solo en el espacio de trabajo de MATLAB durante la simulación.

```
function y = fcn(t, t1, t2, SteeringDelta, SteeringStep )
%#codegen                                     EXTERNAL CONDITIONS

y = [0,0]; INITIAL CONDITIONS

y(1) = 0.0;
y(2) = 0.0;

if(SteeringStep == 1)
% Ramp with discontinuous derivative

if (( t > t1 ) && ( t < t1+1.0 ))
y(1) = SteeringDelta*(t-t1); OUTPUT
y(2) = SteeringDelta;
```

Figura 22. Parámetros de la función.

En versiones anteriores de MATLAB (menores a v2011) este bloque fue conocido como **Función de MATLAB embebida**; esto podría causar incompatibilidades de plataformas entre versiones diferentes de MATLAB (en este caso MATLAB 2014 y 2010b) llegando a tener errores en el proceso de compilación del modelo (véase **Figura 23**).

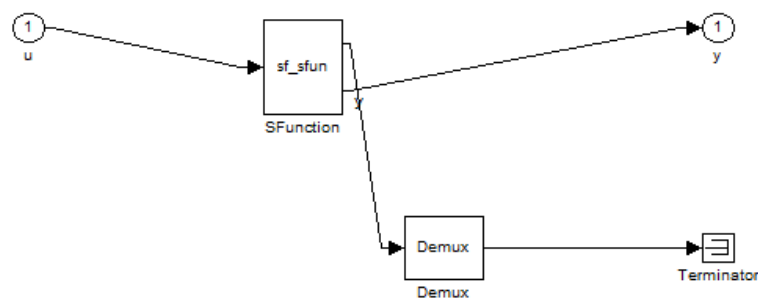


Figura 23. Incompatibilidad entre versiones de MATLAB.

3.3.2. Modelo versión 2 – Usando bloques de funciones-S nivel 2

Para solucionar el inconveniente de la primera versión del modelo se usará el bloque **Función-S nivel 2 de MATLAB**. Este bloque está ubicado en *Librería de Simulink >> Simulink >> Funciones definidas por el usuario* como se observa en la **Figura 24** y tiene las siguientes propiedades.

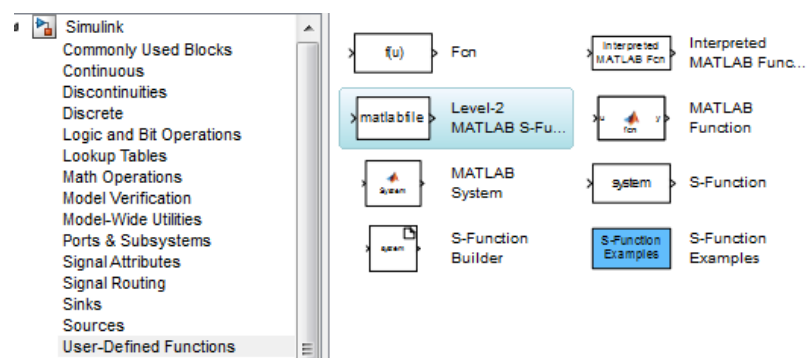


Figura 24. Bloque de función-S nivel 2 de MATLAB.

- Permite el control del tamaño de las entradas, salidas, estados y vectores del programa.
- Permite usar cualquier función, toolbox o estructura de datos, propios del lenguaje de MATLAB.
- Es código interpretado.
- No permite la generación de código.

Es posible especificar el nombre de la función-S al hacer doble clic y editar el bloque en la *ventana de edición*. También es posible definir parámetros externos en el campo de *parámetros* asegurándose que estén separados por comas como se observa en la **Figura 25**.

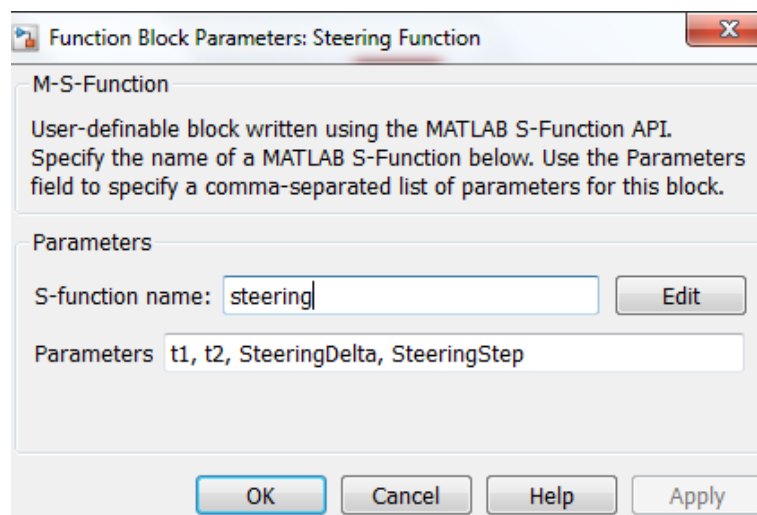


Figura 25. Ventana de edición del bloque.

Como se mencionó anteriormente este bloque no usa un lenguaje compilado sino un lenguaje interpretado, el cual ejecuta instrucciones directamente y no requiere de una compilación previa, esto resulta en una ejecución mucho más lenta de la simulación.

3.3.3. Modelo versión 3 – Usando bloques de funciones-S en C

Una de los métodos más eficientes para implementar funciones en Simulink es usando los bloques **Función-S en C**, este método compila el programa en instrucciones de lenguaje de máquina y debe ser escrito en C (esto puede ser difícil para algoritmos que requieran de algebra lineal o funciones toolbox de MATLAB) (Campa, 2014). Este bloque está ubicado en *Librería de Simulink* >> *Simulink* >> *Funciones definidas por el usuario* como se observa en la **Figura 26**.

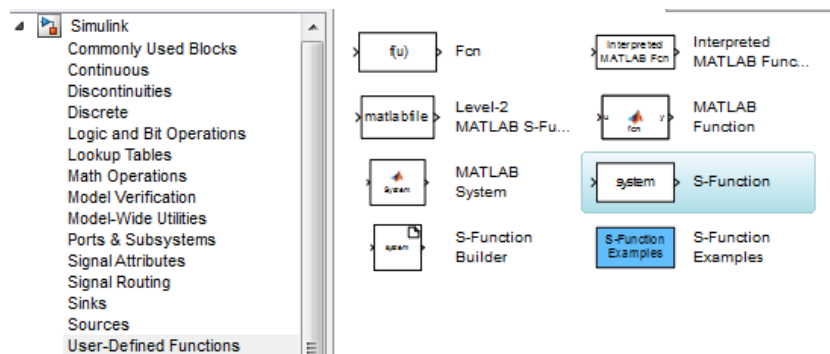


Figura 26. Bloque de función-S en C.

Es posible especificar el nombre de la función-S al hacer doble clic y editar el bloque en la *ventana de edición*. También es posible definir parámetros externos en el campo de *parámetros*, asegurándose que estén separados por comas como se observa en la **Figura 27**.

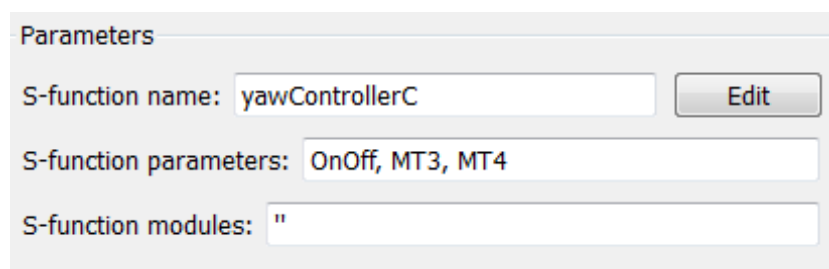


Figura 27. Ventana de edición del bloque.

La estructura para un programa básico está compuesta por los siguientes elementos. (figura)

- Nombre de la función-S
- Llamado de librerías
- Definición de constantes y funciones
- mdlInitializeSizes
- mdlSampleTimes
- mdlInitializeConditions
- mdlOutputs

```
3 #define S_FUNCTION_NAME yawControllerC
4 #define S_FUNCTION_LEVEL 2
5
6 #include "simstruc.h"
7 #include "stdio.h"
8 #include "stdlib.h"
9 #include "math.h"
10
11 #define NPARAMS      3
12 #define NINPUTS      1
13 #define NOUTPUTS     2
14
15 #define PARAM_1      ssGetSFcnParam(S,0)
16 #define PARAM_2      ssGetSFcnParam(S,1)
17 #define PARAM_3      ssGetSFcnParam(S,2)
```

Figura 28. Estructura de programación la función-S en C.

3.4. IMPLEMENTACIÓN DEL MODELO EN SIMULINK

3.4.1. Entradas y salidas del modelo

Versiones tempranas del modelo utilizaron 36 variables iniciales para sus operaciones y calcula aproximadamente 33 salidas. La **Tabla 3** indica los valores y variables iniciales con su respectiva descripción.

3.4.2. Sistema de dirección

Este sub-modelo es responsable de calcular el ángulo de dirección y su derivada utilizando el bloque de la función-S en C llamado “**steeringC**” ([ANEXO B](#)) como se ve en la **Figura 29**.

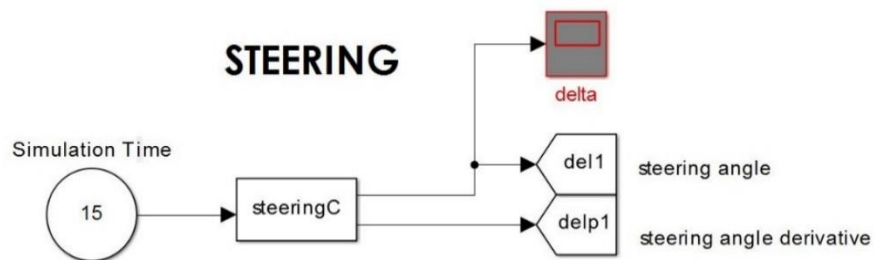


Figura 29. Bloque del sistema de dirección en Simulink.

Este bloque utiliza las variables $t1$, $t2$, $SteeringDelta$ y $SteeringStep$ de la **Tabla 3** junto con expresiones matemáticas para calcular el ángulo de dirección (el código puede ser observado en el **Error! Reference source not found.**). La evolución de este valor durante la simulación representa la trayectoria que sigue el vehículo, esta muestra la variación del ángulo de dirección supuestamente efectuado por el conductor del vehículo. (véase la Figura 30).

Tabla 3.

Nomenclatura de los parámetros utilizados en el modelo

Parámetros del Vehículo		
<i>Variables</i>	<i>Valor</i>	<i>Descripción</i>
m	150	Masa del vehículo. [kg]
g	9.8	Aceleración de la gravedad. [m/s ²]
mg	1470	Peso del vehículo. [N]
a	0.7	Distancia entre el eje frontal de las llantas y el centro de gravedad. [m]
b	0.3	Distancia entre el eje posterior de las llantas y el centro de gravedad. [m]
s	0.4	Distancia entre el eje lateral de las llantas y el centro de gravedad. [m]
Jz	35	Momento de inercia con respecto al eje Z. [kg/m ²]
Parámetros de las llantas		
Rd	0.2	Radio efectivo. [m]
Jh	0.3	Momento de inercia de las llantas frontales. [kg/m ²]
Jv	0.3	Momento de inercia de las llantas posteriores. [kg/m ²]
Parámetros del controlador del ángulo de guiñada		
OnOff	1	Activador del controlador
P_factor	100	Constante proporcional del controlador PID.
I_factor	10	Constante integral del controlador PID.
D_factor	30	Constante derivativa del controlador PID.
Parámetros para la velocidad característica		
lf	0.5	Distancia teórica entre el eje frontal de las llantas y el centro de gravedad. [m]
lr	0.5	Distancia teórica entre el eje posterior de las llantas y el centro de gravedad. [m]
l	1	Distancia teórica entre el eje frontal y el eje posterior de las llantas. [m]
c.v	1000	Rigidez de las llantas frontales.
c.h	1200	Rigidez de las llantas posteriores.
v.h	–	Velocidad característica
SteeringDelta	0.145	Ángulo de la dirección para la trayectoria.
SteeringStep	2	Trayectoria pre programada.
t1	1	Punto de inflexión para las trayectorias.
t2	8	
Vx	20	Velocidad inicial del vehículo.

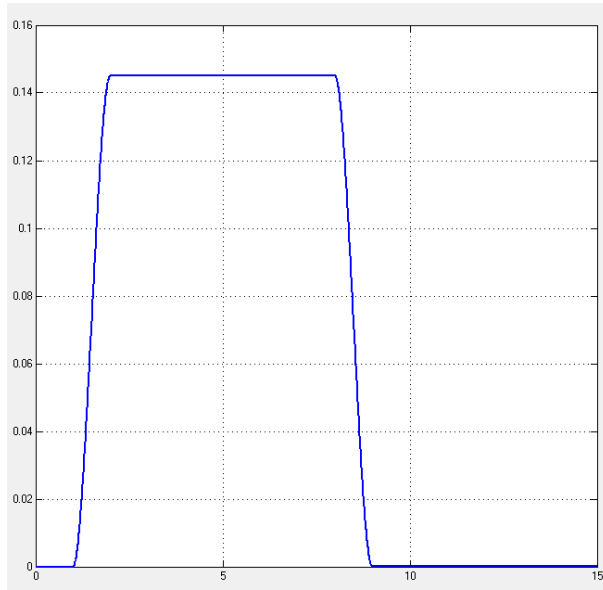


Figura 30. Grafica del ángulo de dirección vs tiempo de simulación.

3.4.3. Variación del yaw

Una vez que el ángulo de la dirección haya sido calculado es posible calcular la variación del ángulo de yaw. Para esta operación se utilizará la ecuación obtenida de SIMPACK User Meeting 2014, (Dorsch, 2014).

$$\dot{\psi} = \frac{V_x \delta_w}{(l_v + l_h) \left(1 + \frac{V_x^2}{V_{ch}^2}\right)} \quad (3.1)$$

Dónde:

δ_w es el ángulo de dirección.

V_x es la velocidad longitudinal del vehículo.

V_{ch} es la velocidad característica del vehículo dada por la ecuación (3.2) y es representativo de la dinámica del vehículo durante el estado estacionario.

$$v_{ch} = \frac{c_v * c_h * l^2}{m(c_h * l_h - c_v * l_v)} \quad (3.2)$$

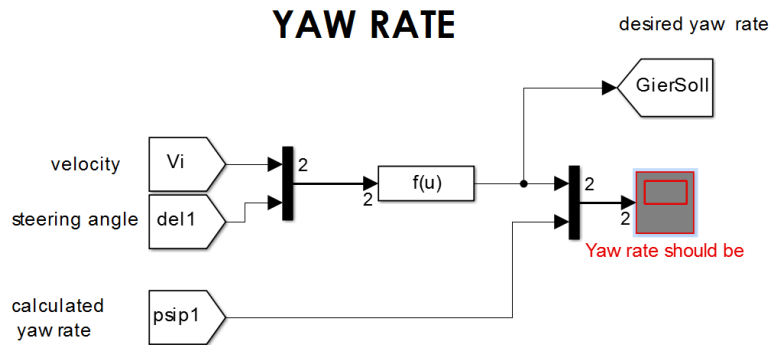


Figura 31. Bloque de la variación de ángulo de yaw en Simulink.

En la **Figura 32** se puede observar la diferencia entre el valor deseado o teórico (en blue) y el valor calculado (en rojo), esto es importante para el siguiente paso.

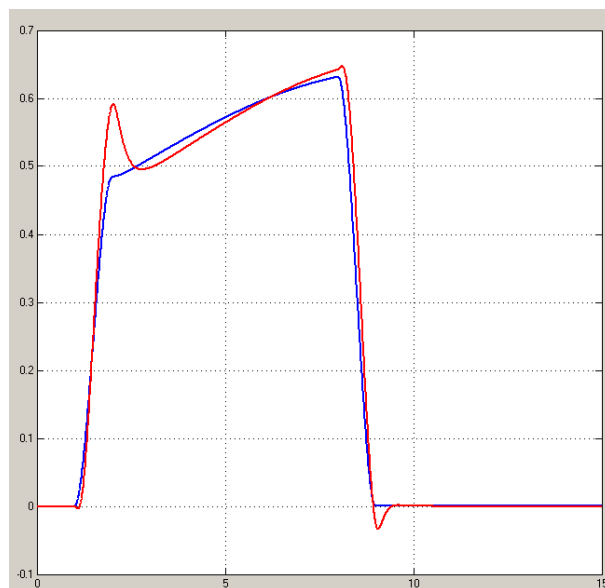


Figura 32. Comparación del valor real con el teórico.

3.4.4. Controlador de yaw

La diferencia entre el valor deseado y el obtenido de la variación del ángulo de yaw es necesario para el controlador (véase **Figura 33**), el cual identifica una variación de este valor y según esto genera una señal de control que consta de dos valores de salida. Estas salidas controlan el torque dirigido a

los motores traseros que son responsables del movimiento del vehículo. Como se observa en la **Figura 34** este torque cambia de acuerdo a la variación en la trayectoria lo que significa una variación de ángulo de yaw.

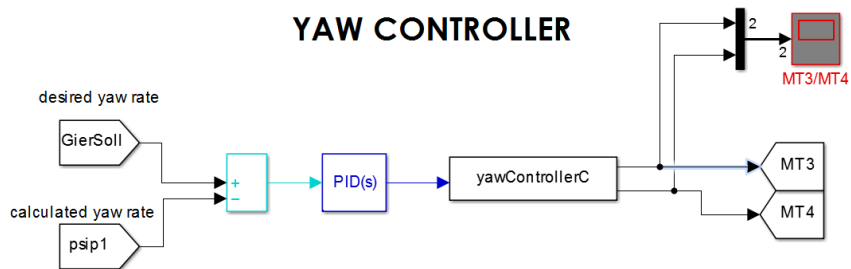


Figura 33. Bloque del controlador en Simulink.

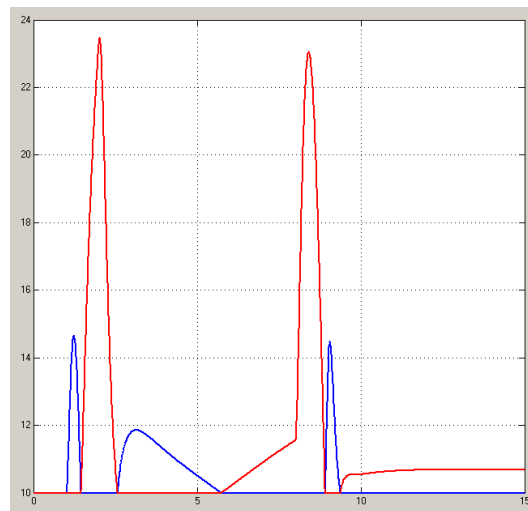


Figura 34. Gráfica de las salidas del controlador.

3.4.5. Modelo de vehículo

Como se mencionó anteriormente, el modelo del vehículo tendrá siete grados de libertad y el sistema de suspensión no será considerado. Las dimensiones, ecuaciones y el diseño del vehículo utilizadas en este proyecto están basadas en la referencia (Oertel, Kart Kursmodell).

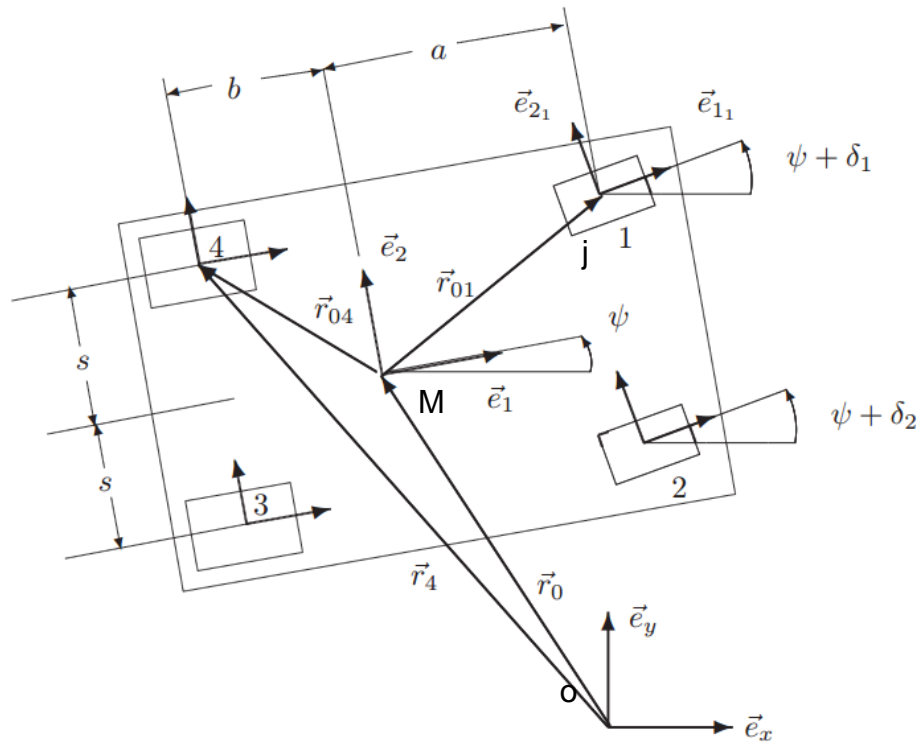


Figura 35. Esquema del vehículo

FUENTE: (Oertel, Kart Kursmodell)

En la **Figura 35** se muestra el esquema del vehículo con sus respectivos vectores, dimensiones y ángulos. Para obtener las ecuaciones de los modelos se trabajará con 3 sistemas de referencia: sistema coordinado global, sistema coordinado fijo en el vehículo y sistema coordinado fijo en los neumáticos. Los vectores desde el centro de gravedad del vehículo a cada neumático son:

$$\begin{aligned}
 \vec{r}_{01} &= a \cdot \vec{e}_1 + s \cdot \vec{e}_2 \\
 \vec{r}_{02} &= a \cdot \vec{e}_1 - s \cdot \vec{e}_2 \\
 \vec{r}_{03} &= -b \cdot \vec{e}_1 - s \cdot \vec{e}_2 \\
 \vec{r}_{04} &= -b \cdot \vec{e}_1 + s \cdot \vec{e}_2
 \end{aligned} \tag{3.3}$$

El vector desde el origen del sistema coordinado global al centro de gravedad del vehículo está dado por la siguiente ecuación.

$$\vec{r}_o = x(t) \cdot \vec{e}_x + y(t) \cdot \vec{e}_y \quad (3.4)$$

La matriz de transformación entre el sistema global y el fijo en el vehículo es:

$$\underline{T}_{OM} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \quad (3.5)$$

La matriz de transformación entre el sistema global y el fijo en cada neumático ($j = 1,2,3,4$) es:

$$\underline{T}_{Oj} = \begin{bmatrix} \cos(\varphi_j) & -\sin(\varphi_j) \\ \sin(\varphi_j) & \cos(\varphi_j) \end{bmatrix} \quad (3.6)$$

Siendo:

$$\begin{aligned} \varphi_1 &= \psi + \delta_1 & \varphi_3 &= \psi \\ \varphi_2 &= \psi + \delta_2 & \varphi_4 &= \psi \end{aligned}$$

Luego, al derivar la matriz (3.6) se obtiene:

$$\dot{\underline{T}}_{Oj} = \begin{bmatrix} -\dot{\varphi}_j \sin(\varphi_j) & -\dot{\varphi}_j \cos(\varphi_j) \\ \dot{\varphi}_j \cos(\varphi_j) & -\dot{\varphi}_j \sin(\varphi_j) \end{bmatrix} = \dot{\varphi}_j \underbrace{\begin{bmatrix} -\sin(\varphi_j) & -\cos(\varphi_j) \\ \cos(\varphi_j) & -\sin(\varphi_j) \end{bmatrix}}_{\underline{B}_{Oj}} = \dot{\varphi}_j \cdot \underline{B}_{Oj} \quad (3.7)$$

El vector desde el origen del sistema global hacia cada neumático es:

$$\underline{r}_j = \underline{r}_o + \underline{T}_{Oj} \cdot \underline{r}_{Oj} \quad (3.8)$$

Al derivar la ecuación (3.8) se obtendrán los vectores de velocidad:

$$\begin{aligned}\underline{v}_j^{(0)} &= \dot{\underline{r}}_o^{(0)} + \dot{\underline{T}}_{oj} \cdot \underline{r}_{oj}^{(i)} = \dot{\underline{r}}_o^{(0)} + \dot{\varphi}_j \cdot \underline{B}_{oj} \cdot \underline{r}_{oj}^{(j)} \\ \underline{v}_j^{(j)} &= \underline{T}_{j0} (\dot{\underline{r}}_o^{(0)} + \dot{\varphi}_j \cdot \underline{B}_{oj} \cdot \underline{r}_{oj}^{(j)})\end{aligned}\quad (3.9)$$

Siendo

$$\underline{T}_{j0} = \begin{bmatrix} \cos(\varphi_j) & \sin(\varphi_j) \\ -\sin(\varphi_j) & \cos(\varphi_j) \end{bmatrix}$$

Finalmente uniendo las ecuaciones en una sola se obtiene:

$$\underline{B}_{oj} = \underline{T}_{oj} \cdot \underline{R} = \begin{bmatrix} -\sin(\varphi_j) & -\cos(\varphi_j) \\ \cos(\varphi_j) & -\sin(\varphi_j) \end{bmatrix} = \begin{bmatrix} \cos(\varphi_j) & -\sin(\varphi_j) \\ \sin(\varphi_j) & \cos(\varphi_j) \end{bmatrix} \underbrace{\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}}_{\underline{K}} \quad (3.10)$$

$$\underline{v}_j^{(j)} = \underline{T}_{j0} (\dot{\underline{r}}_o^{(0)} + \dot{\varphi}_j \cdot \underline{B}_{oj} \cdot \underline{r}_{oj}^{(j)}) = \underline{T}_{j0} \cdot \dot{\underline{r}}_o^{(0)} + \dot{\varphi}_j \cdot \underline{R} \cdot \underline{r}_{oj}^{(j)}$$

$$\underline{v}_j^{(j)} = \begin{bmatrix} \dot{x}(t) \cos(\varphi_j(t)) + \dot{y}(t) \sin(\varphi_j(t)) - c_{j2} \cdot \dot{\varphi}(t) \\ -\dot{x}(t) \sin(\varphi_j(t)) + \dot{y}(t) \cos(\varphi_j(t)) - c_{j1} \cdot \dot{\varphi}(t) \end{bmatrix}^{(j)} \quad (3.11)$$

Tomando en cuenta que:

c_{j1}, c_{j2} son las distancias desde el centro de gravedad hacia cada llanta (a, b, s como se observa en la **Figura 35**).

La matriz de la expresión (3.11) es la matriz de transformación para la velocidad longitudinal y lateral del vehículo desde el **sistema global fijo** hacia el **sistema coordinado en cada llanta**, con el objetivo de encontrar las fuerzas en cada llanta.

3.4.6. Ecuaciones de movimiento

Las aceleraciones longitudinales (ax) y lateral (ay) pueden ser obtenidas de la segunda ley de Newton.

$$m \cdot \ddot{r}_o^{(0)} = \sum_j F_j^{(0)} = \sum_j T_{0j} \cdot F_j^{(j)} \quad (3.12)$$

La expresión (3.12) indica que la aceleración longitudinal y lateral del vehículo es igual a la sumatoria de fuerzas longitudinales y laterales respectivamente de cada llanta dividido para la masa del vehículo.

Reemplazando la expresión (3.13) en (3.12) se obtendrá las ecuaciones (3.14) y (3.15).

$$F_j^{(j)} = \begin{bmatrix} F_u \\ F_s \end{bmatrix} \quad (3.13)$$

$$m \cdot \ddot{x}(t) = \sum_j (F_{uj} \cos(\varphi_j(t)) - F_{sj} \sin(\varphi_j(t))) \quad (3.14)$$

$$m \cdot \ddot{y}(t) = \sum_j (F_{uj} \sin(\varphi_j(t)) + F_{sj} \cos(\varphi_j(t))) \quad (3.15)$$

Para la rotación con respecto al eje vertical de vehículo (eje Z), se usará la ecuación:

$$J \cdot \ddot{\psi} = \sum_j \left(e_3^{(m)T} (\mathbf{r}_{0j}^{(m)} \times F_j^{(m)}) + Mz_j \right) = \sum_j \left(e_3^{(m)T} (\mathbf{r}_{0j}^{(m)} \times T_{mj} F_j^{(j)}) + Mz_j \right) \quad (3.16)$$

Resultando en:

$$J \cdot \ddot{\psi}(t) = \sum_j \left((c_{j1} \cdot \sin(\delta_j(t)) - c_{j2} \cdot \cos(\delta_j(t))) F_{uj} + (c_{j2} \cdot \sin(\delta_j(t)) + c_{j1} \cdot \cos(\delta_j(t))) F_{sj} + Mz_j \right) \quad (3.17)$$

Es importante mencionar que el ángulo de dirección δ_j solo abarca valores $j=1, 2$ debido a que solo las llantas delanteras tienen mecanismo de dirección.

La implementación de estas ecuaciones en Simulink se puede observar en la **Figura 36**.

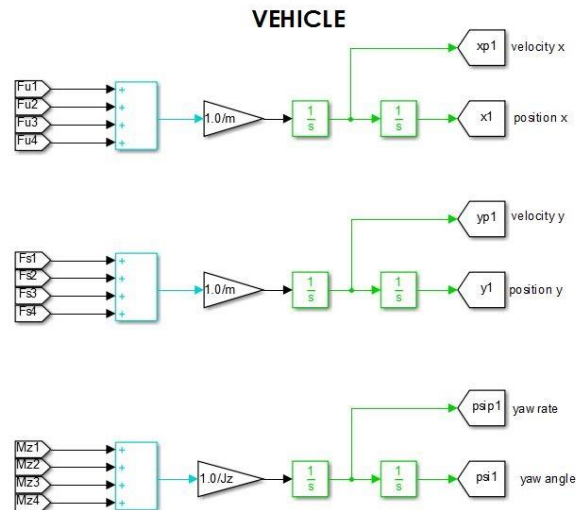


Figura 36. Bloques de las ecuaciones de movimiento del vehículo.

3.4.7. Modelo de llanta

El objetivo de este subsistema es el de calcular fuerzas que afectan o influyen a las llantas, las variables necesarias para este cálculo son: velocidades longitudinal y lateral (v_x , v_y), el ángulo de yaw y su respectiva variación (véase la **Figura 37**).

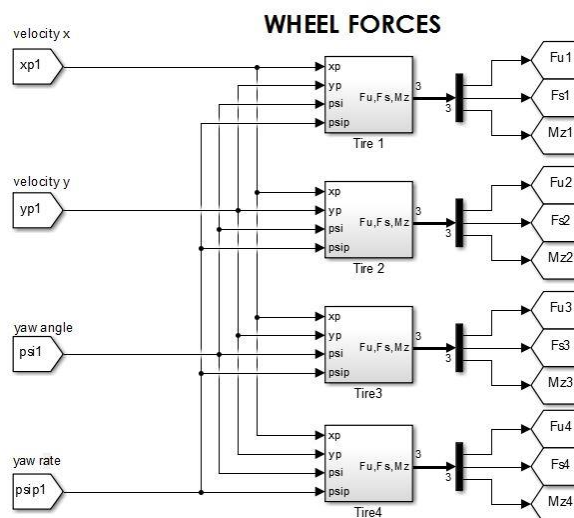


Figura 37. Bloque para el cálculo de las fuerzas de las llantas.

3.4.7.1. Transformación de sistemas

Las entradas deben ser transformadas de un sistema coordinado a otro. En la **Figura 38** se puede observar este paso, la función-S en C “**f1C**” realiza la transformación de las entradas del sistema de referencia global al fijo en el vehículo. De la misma manera, las salidas del modelo de la llanta son transformadas del sistema de referencia fijo del vehículo al sistema global utilizando la función-S en C “**f2C**”.

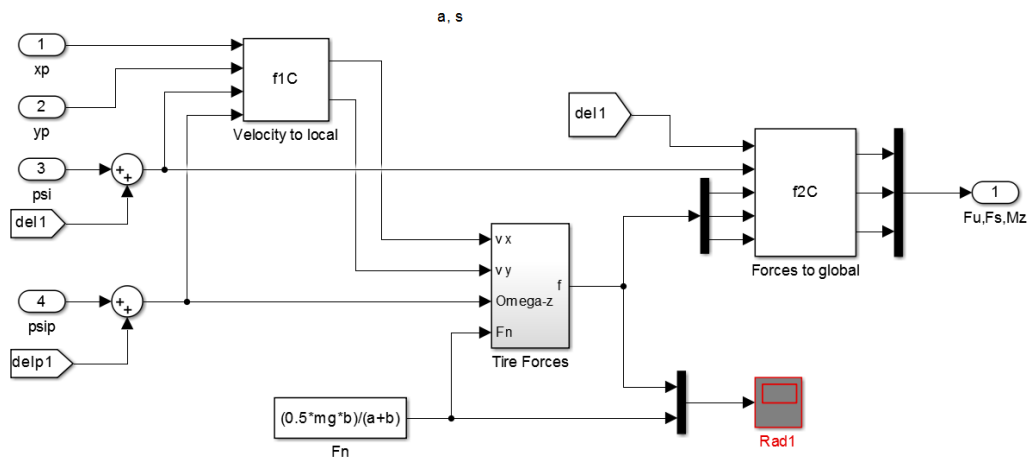


Figura 38. Transformación de coordenadas.

3.4.7.2. Ecuación dinámica para el movimiento angular de la llanta

La velocidad angular de la llanta está dada por la ecuación (3.18).

$$\dot{W}_\omega = \frac{Te - Tb - R_d F_t - R_d F_w}{J_w} \quad (3.18)$$

Donde:

Te : Torque del eje del motor.

Tb : Torque de frenado.

R_d : Radio efectivo de la llanta.

F_t : Fuerza de tracción.

F_w : Coeficiente de fricción viscoso de la llanta.

J_w : Momento de inercial de la llanta.

En este caso, el torque de frenado (T_b) no será considerado y solo se utilizarán las fuerzas y momentos de la **Figura 39**. La implementación de esta ecuación puede ser apreciada en la **Figura 40**.

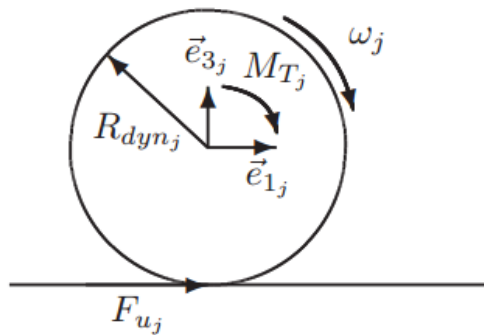


Figura 39. Fuerzas que afectan a la rueda.

FUENTE: (Oertel, Kart Kursmodell)

3.4.7.3. Fuerzas presentes en las llantas.

La velocidad angular de la llanta, la velocidad longitudinal y lateral del vehículo (V_x y V_y) serán los valores de entrada para calcular el *deslizamiento longitudinal y lateral*. Estos valores junto con la variación del ángulo de dirección y la fuerza normal en cada llanta (F_n) serán necesarios para calcular las fuerzas que las afectan.

Fuerzas normales o verticales para las llantas frontales (1, 2) y las llantas traseras (3, 4) puede ser obtenidas con las siguientes ecuaciones:

$$F_{N1} = F_{N2} = m \cdot g \frac{b}{a+b} \quad (3.19)$$

$$F_{N3} = F_{N4} = m \cdot g \frac{a}{a+b} \quad (3.20)$$

3.4.7.4. RMOD-K Formula

RMOD-K Formula es una colección abierta de dos simples modelos de llantas, herramientas de optimización e interfaces de simulación, elaborada por el Dr. Ing. Christian Oertel de la Universidad de Ciencias Aplicadas de Brandeburgo.

Se utilizará RMOD-K Formula para calcular las fuerzas que afectan a las llantas. Esta librería fue implementada en Simulink utilizando la función-S C “**tyreForcesC**” ([ANEXO A](#)) de la **Figura 40**, esta función consta de 3 entradas, 13 variables previamente definidas y tiene 3 salidas: **fuerza longitudinal, fuerza lateral, momento en Z**.

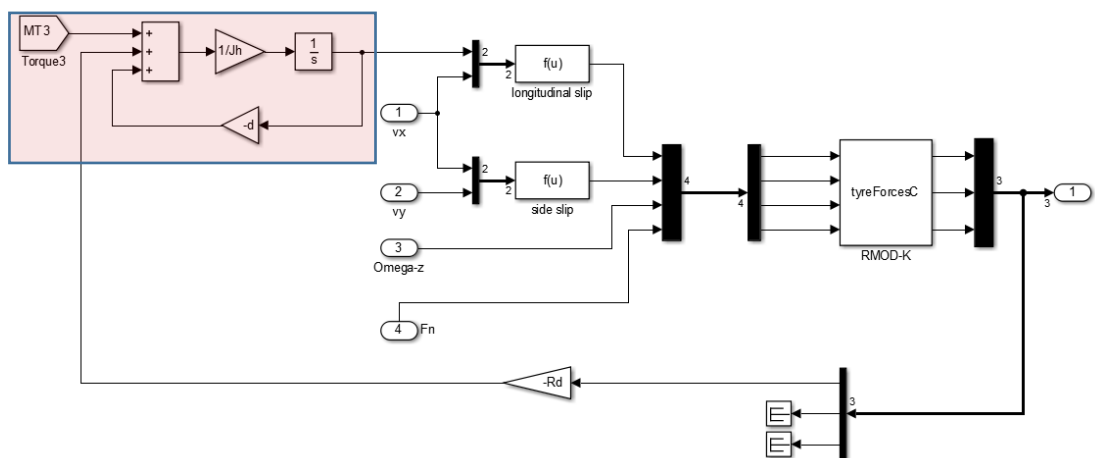


Figura 40. Modelo de la llanta en Simulink.

3.5. PARÁMETROS DE SIMULACIÓN DEL MODELO

Parámetros de simulación como tiempo inicial, tiempo final, simulación con intervalos de tiempo variables o fijos y el tipo de solucionador pueden ser configurados en *Simulación >> Configuración >> Parámetros >> Solucionador*.

El tiempo inicial y final indica el tiempo en el que correrá la simulación, este valor está escalado en segundos. El tiempo inicial es necesario para aplicaciones en tiempo real mientras que el tiempo final define la duración de la simulación, si el valor es fijado en 'inf' entonces la simulación no parará en un tiempo definido. Existen dos tipos de simulaciones con respecto al tiempo, en simulaciones con intervalos variables, el tiempo de muestreo varía dinámicamente dependiendo del modelo mientras que, en simulaciones con intervalos fijos, el tiempo de muestreo permanece constante a lo largo de la simulación, este método permite la generación de código.

Finalmente, el tipo de *solver* define el método que calculará y resolverá el modelo en la simulación. Es importante mencionar que mientras más complejo sea el solucionador más poder computacional requerirá, lo que resultaría en un tiempo de ejecución mayor. Parámetros de simulaciones que se utilizarán en este proyecto se pueden observar en la **Tabla 4** y **Figura 41**.

Tabla 4.

Parámetros de simulación para el modelo GO-KART

Opción	Parámetro	Valor
	Tiempo inicial	0
	Tiempo final	15
Solver	Tipo de simulación	Intervalo de tiempo fijo
	Tiempo de muestreo	0.01
	Tipo de solver	ode1
Optimización	Reducción de bloques	Deactivada

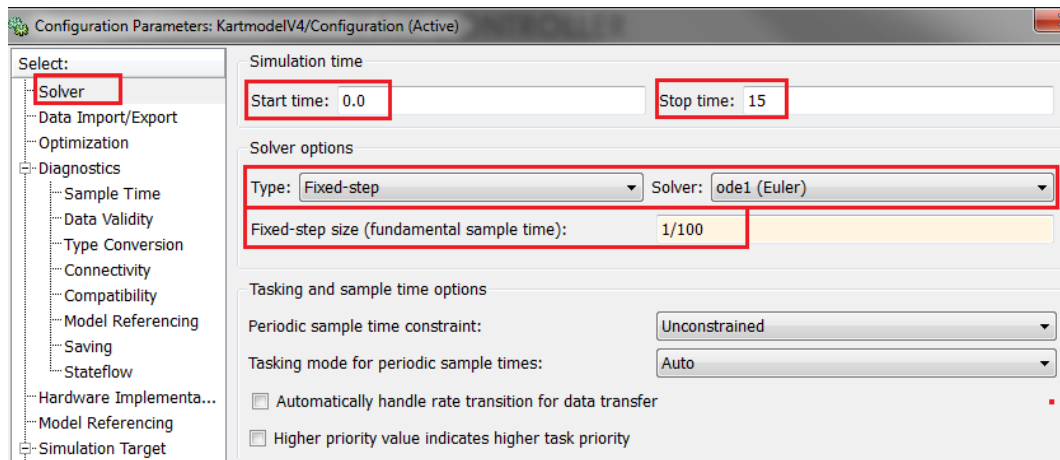


Figura 41. Configuración parámetros de la simulación.

3.5.1. Especificaciones de la computadora anfitrión

Es importante conocer las especificaciones de la computadora anfitrión para comparar los resultados con otros equipos.

Procesador: Intel (R) Core (TM) 2 Duo CPU @3.16GHz RAM 4GB.

Memoria principal: Memoria RAM 4GB

Arquitectura: x64

Sistema Operativo: Microsoft Windows 7

Versiones del software a usar:

- MATLAB/Simulink versión 2010b con toolbox RTI
- DSPACE ControlDesk 3.7.2 y ControlDesk Next Generation 4.2

CAPÍTULO 4

4. ANÁLISIS EN TIEMPO REAL DE UN MODELO COMPLEJO

Luego de haber modelado y testeado el modelo de GO-KART, se implementará un modelo de vehículo más complejo utilizando el mismo proceso que el anterior modelo (análisis matemático – modelo Simulink – transferencia a hardware DSPACE). A continuación, se expone una lista de algunos subsistemas presentes en vehículos modernos:

- Modelo de llanta diferente y más complejo.
- Implementación de condiciones ambientales y de camino.
- Control de crucero.
- Sistema de suspensión.
- Control activo de chasis (ABC).
- Sistema hidráulico de frenos.
- Sistema antibloqueo de frenos (ABS).
- Controlador electrónico de estabilidad. (ESC)

Debido a su importancia y relevancia se ha decidido modelar los subsistemas: **sistema hidráulico de frenos con ABS** y **sistema de suspensión con ABC**, los cuales serán simulados en diferentes escenarios, condiciones y pruebas.

4.1. DINÁMICA DEL MODELO COMPLEJO VEHÍCULO

La dinámica para este modelo de vehículo será la misma que el GO-KART, por lo tanto, tendrá los mismos bloques en Simulink para la dirección, variación de ángulo de yaw, controlador de yaw, movimiento del vehículo y modelo de llanta (véase **Figura 42** y **Figura 43**). Los únicos cambios importantes al modelo serán la inclusión de un modelo de freno para las cuatro llantas con sus respectivos momentos de freno y la inclusión de un modelo de suspensión con su respectivo controlador/estimador.

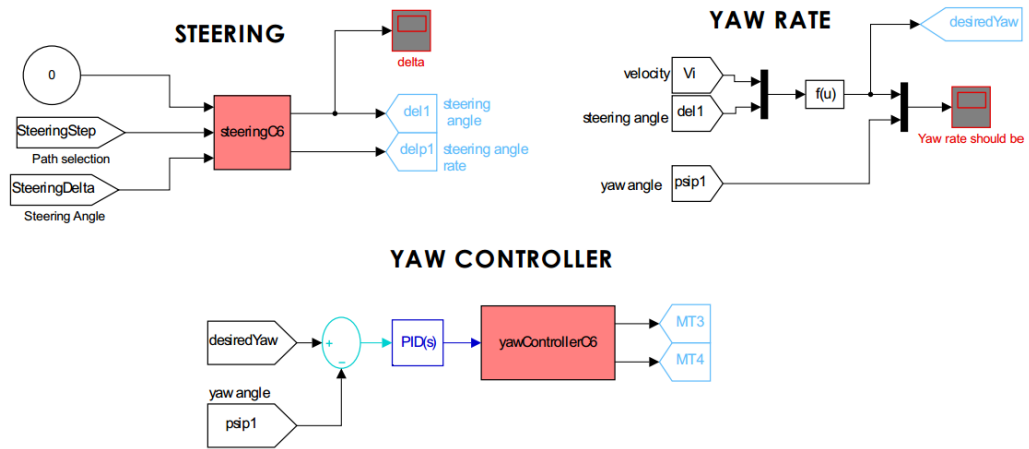


Figura 42. Bloques de la dirección y yaw del vehículo.

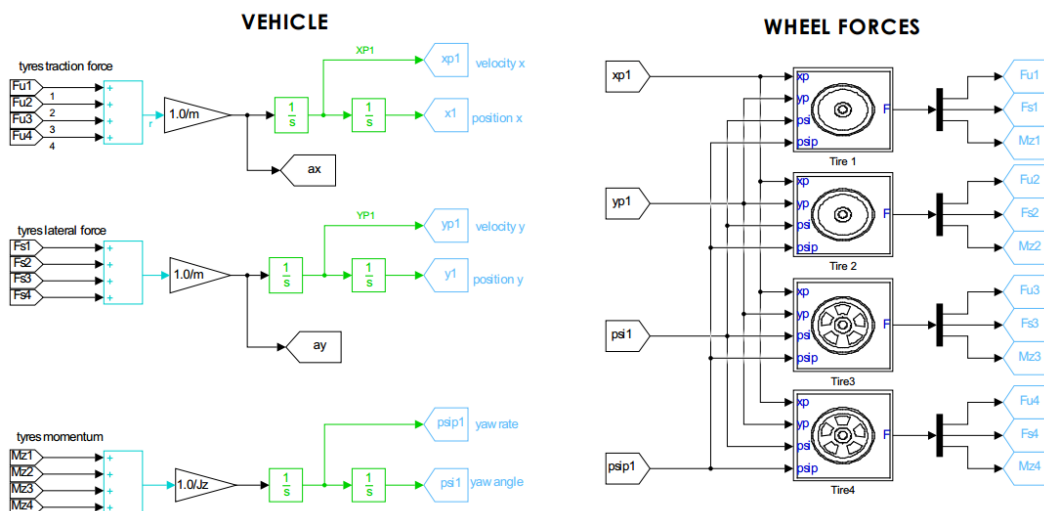


Figura 43. Bloque de la dinámica del modelo complejo.

Este modelo de vehículo contará con **14 grados de libertad**, 6 en el vehículo (longitudinal, lateral, vertical, roll, pitch y yaw) y 8 en las llantas (rotacional y vertical).

Es importante mencionar que los modelos de los subsistemas serán separados por bloques siguiendo lo establecido anteriormente (véase **Figura 44**).

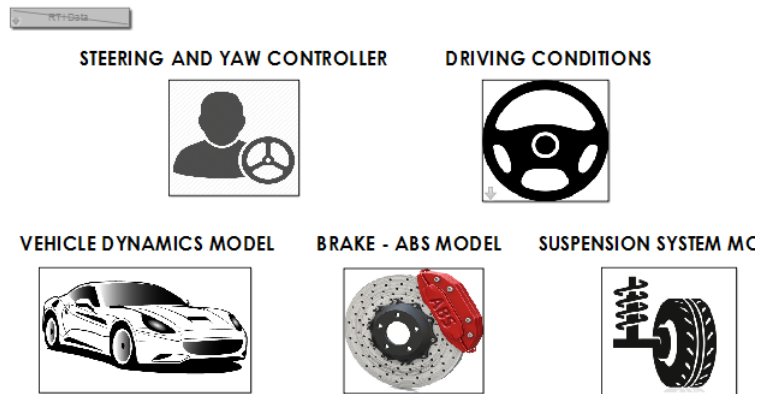


Figura 44. Modelo complejo en subsistema.

4.2. SISTEMA DE FRENOS CON CONTROL ANTIBLOQUEO (ABS)

4.2.1. Sistema de frenos

El sistema de frenos es uno de los más importantes en un vehículo siendo el encargado de reducir la velocidad del vehículo o detenerlo completamente. Los frenos pueden ser: de fricción, de bombeo o electromagnéticos siendo el friccional el más utilizado. Como se observa en la **Figura 45**, componentes de este tipo de freno son:

- Pedal de freno
- Cilindro principal
- Dispositivos de frenado. - Disco de frenos y pinza de freno que sostiene la superficie de fricción (pastilla).

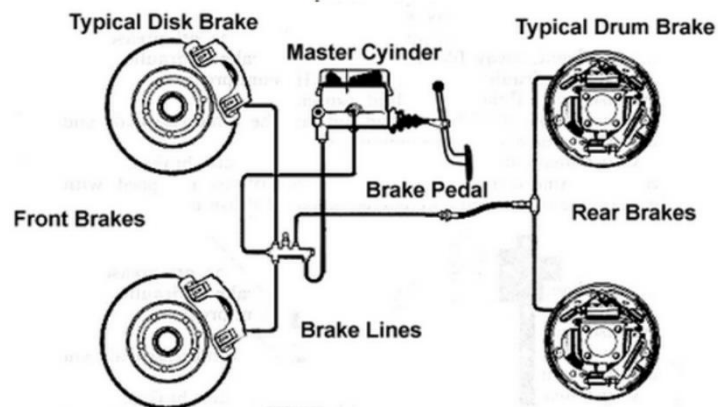


Figura 45. Componentes del sistema de freno.

FUENTE: (Fernandez, 2012)

Cuando el conductor presiona el pedal de frenos, el cilindro principal genera una presión hidráulica, usualmente el sistema de frenos viene con un sistema aumentador de presión. Esta presión luego es transmitida mediante las líneas de alimentación a las pinzas o cilindros de tambor, los cuales presionan a las superficies de fricción contra el disco de freno convirtiendo la energía cinética en calor (Fernandez, 2012).

4.2.2. Sistema antibloqueo de frenos (ABS)

ABS viene de la palabra alemana “*Antiblockiersystem*”, es un sistema de seguridad presente en vehículos modernos. Utiliza un sistema mecatrónico para prevenir el bloqueo de neumáticos (esto ocurre cuando la velocidad angular de la rueda tiende a cero $w = 0$), lo que causa inestabilidad para conducir efectivamente y un incremento en la distancia de frenado. Este sistema mejora el manejo del vehículo en condiciones difíciles de frenado (véase la **Figura 46**). El funcionamiento de este sistema está basado en las técnicas avanzadas de manejo llamadas umbral y cadencia, estas técnicas permiten al conductor controlar de mejor manera el vehículo preservando el control de dirección mientras se frena.

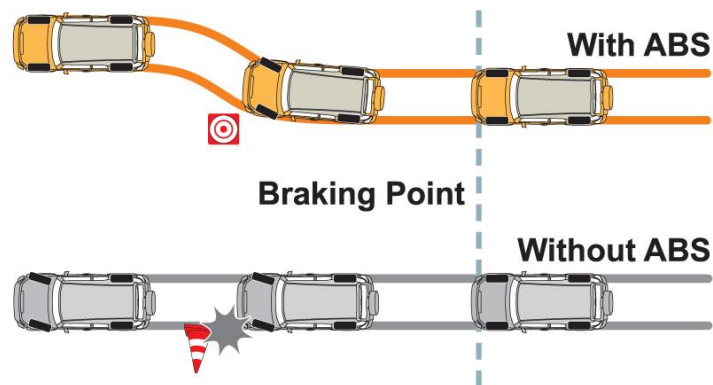


Figura 46. Condiciones de frenado.

FUENTE: (HiTech Auto Mechanical, 2016)

Componentes de este sistema son:

- **Sensores de velocidad.** - Determinan la aceleración o desaceleración de cada rueda, para medir la variación de velocidad el sensor puede ser de Efecto Hall o con discos dentados.
- **Unidad de control electrónico.** - Recibe la información de los sensores ubicados en cada rueda, detecta si una rueda gira más rápido o más despacio que otras indicando un posible bloqueo. Luego, el controlador toma acción de acuerdo al algoritmo de control y reduce o incrementa la presión del frenado en las llantas afectadas.
- **Válvulas para el control de los frenos hidráulicos.**
- **Bomba.** - Restaura la presión de los frenos hidráulicos luego de usar las válvulas.

4.2.3. Modelo matemático

De acuerdo a las referencias (Fernandez, 2012) y (Shigley, 2004), el momento de frenado Tb puede ser calculado con una ecuación que está en función de las siguientes variables:

- Diámetro interno y externo de un disco de frenos d_o, d_i .
- Presión entre la pastilla y disco de frenos P_{dp} .
- Coeficiente de fricción entre la pastilla y el disco de frenos C_{fpd} .

Estos parámetros se traducen en la ecuación (4.1).

$$Tb = P_{dp} * \pi * C_{fpd} * d_i * [d_o^2 - d_i^2]$$

$$Tb = P_{dp} * \pi * C_{fpd} * (0.58 * d_o) * [d_o^2 - (0.58 * d_o)^2] \quad (4.1)$$

Donde:

$$P_{dp} = 2 * Bp * 10 * \left[\frac{\pi * caliperD^2}{4 * pad_{area}} \right] \quad (4.2)$$

$$Bp = \min(B_{input}, 7000) \quad (4.3)$$

El resto de variables se muestran en la **Tabla 5**.

Tabla 5.

Parámetros de los frenos

Parámetros del subsistema		
Variables	Valor	Descripción
Tb	—	Momento de frenado
Pdp	—	Presión de frenado entre la pastilla y el disco de frenos.
Cfpd	0.35	Coeficiente de fricción entre la pastilla y el disco de frenos.
do	0.292	Diámetro externo de la llanta. [m]
di	0.58*do	Diámetro interno de la llanta. [m]
padArea	0.007	Area de contacto de la pastilla. [m2]
caliperD	0.06	Diámetro de la pinza de freno [m]

La transferencia de carga del tren trasero hacia el tren delantero en condiciones de freno puede causar que las llantas traseras se bloqueen debido a una pérdida de agarre con la calzada, para solucionar esto se deberá crear una condición en el controlador de freno. La ecuación (4.3) indica que la presión de freno B_p es igual a la presión inicial de freno mientras este valor sea menor a un valor límite de presión para las llantas traseras (7000 Pa). Esto asegura que las llantas delanteras se bloqueen primero en vez de las traseras evitando el **sobre-viraje**, el cual es una situación muy peligrosa ya que el movimiento es inestable. Es importante mencionar que el sistema de frenos será implementado en las **cuatro llantas**.

4.2.4. Modelo en Simulink

El modelo en Simulink visto en la **Figura 47** está basado en las ecuaciones propuestas anteriormente, este modelo consiste en un controlador ABS, un conmutador que valida la condición de ABS, un bloque de retraso hidráulico que representa el régimen transitorio del conductor al presionar el freno y los bloques que representan la ecuación del modelo de frenos.

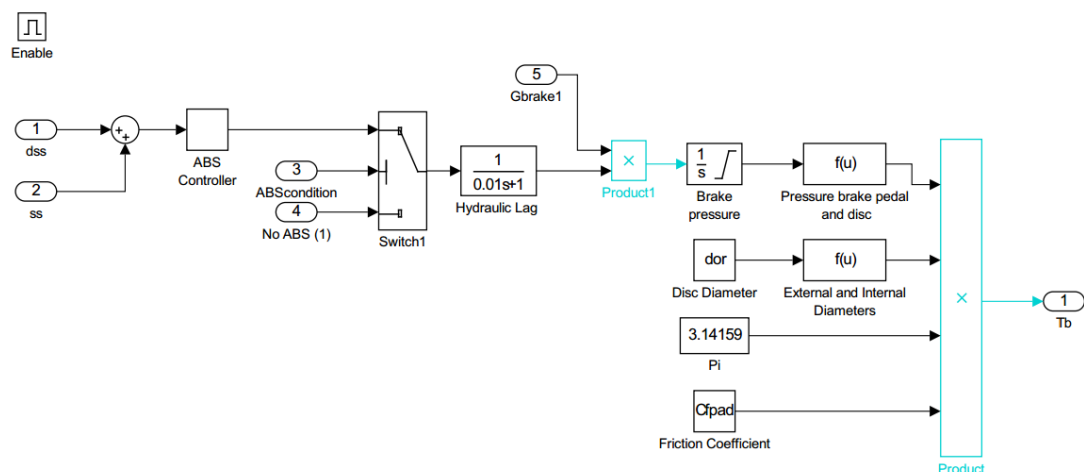


Figura 47. Modelo de sistema de frenos con ABS integrado.

En este caso un controlador ON/OFF fue implementado para controlar el deslizamiento longitudinal, el cual define si una llanta se bloquea o no (como se observó en las ecuaciones de la anterior sección, este controlador ON/OFF recibe el error entre el valor deseado de deslizamiento longitudinal y el valor actual, luego toma acción de acuerdo a este valor (véase **Figura 48**).

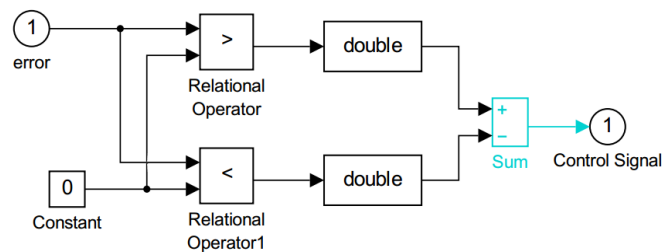


Figura 48. Controlador del ABS.

De acuerdo a (MathWorks, s.f.) un valor deseable de deslizamiento longitudinal es **0.2** lo que significa que el número de revoluciones del neumático en condiciones actuales es igual a 0.8 veces el número de revoluciones bajo condiciones sin freno con la misma velocidad de vehículo longitudinal. Esto maximiza la adherencia entre la llanta y la calzada minimizando la distancia de frenado.

Luego, este valor será multiplicado por el retraso hidráulico y luego será integrado para obtener la presión de freno (B_p). Esta presión de freno será transformada a momento de freno con la ecuación (4.1).

La **Figura 49** muestra los bloques que calculan el momento de freno para cada llanta. Estos valores serán multiplicados por el activador de freno, luego el momento de freno pasará directamente a los bloques sumatoria vistos en **Figura 50** y **Figura 51** para obtener el momento total de la llanta, este valor será dividido para el momento de inercia de la llanta y luego integrado para obtener la velocidad angular de la llanta.

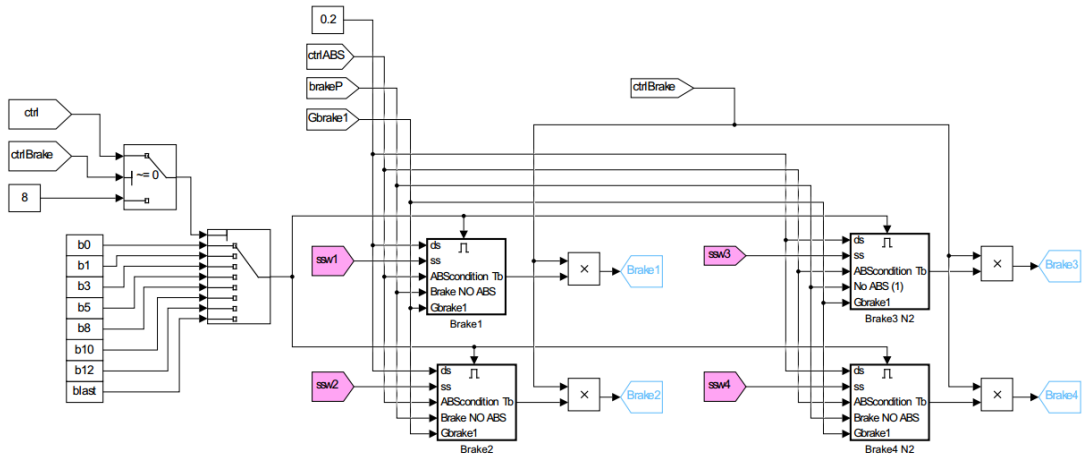


Figura 49. Modelo del sistema de frenos en cada llanta.

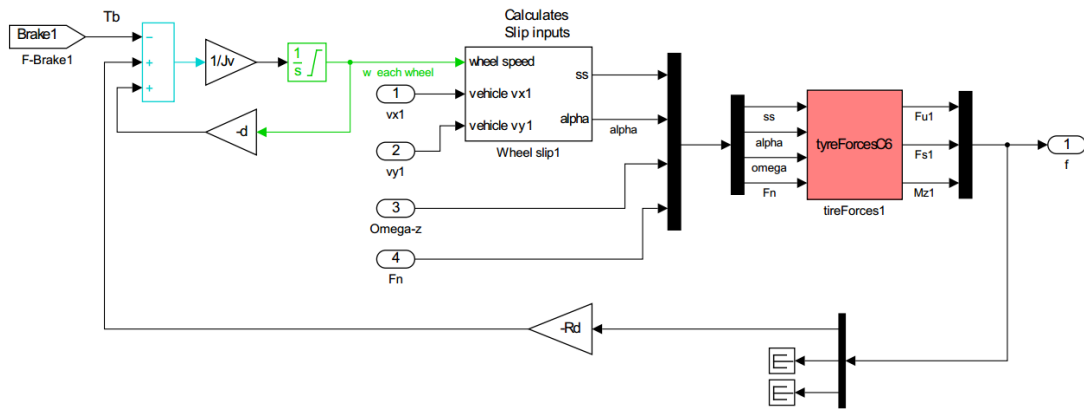


Figura 50. Movimiento angular de la llanta frontal 1.

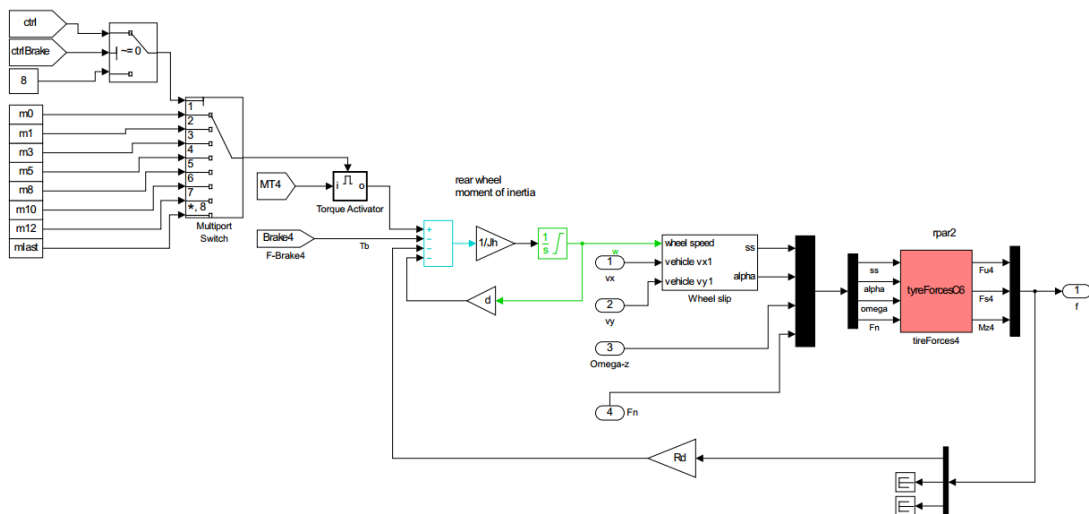


Figura 51. Movimiento angular de la llanta posterior 4.

4.3. SUSPENSIÓN CON CONTROL ACTIVO DEL CHASIS (ABC)

4.3.1. Sistema de suspensión

Un buen sistema de suspensión debe proveer de confort al conducir sobre baches y huecos en la vía. El chasis no debería tener oscilaciones cuando el vehículo experimente perturbaciones en el camino.

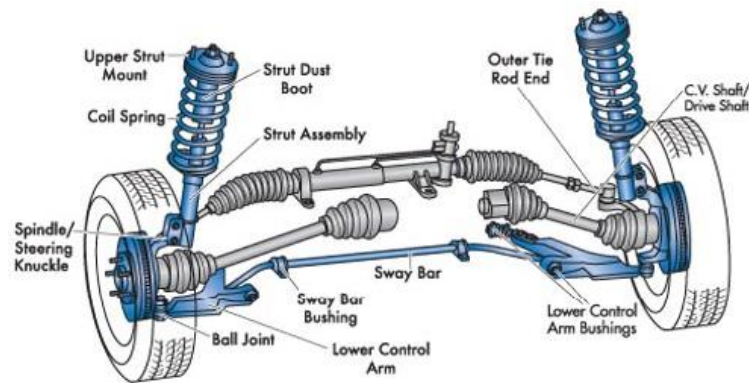


Figura 52. Suspensión para neumáticos típica.

FUENTE: (SpeedLab, s.f.)

4.3.2. Control Activo del Chasis

Este sistema mejora el confort de los pasajeros durante el viaje al detectar irregularidades en el terreno, estas deben ser disipadas o balanceadas con ayuda de un actuador lineal que estará ubicado en cada llanta y que recibe señales de control de la ECU. El uso de un actuador lineal hace que este tipo de suspensión sea más flexible que una suspensión pasiva.

4.3.3. Modelo matemático

Los parámetros del modelo de la suspensión utilizados en este estudio están dados por la **Tabla 6**, la **Figura 53** y están basados en un trabajo realizado por el Prof. Dr. Ing. Christian Oertel – Technische Hochschule Brandenburg.

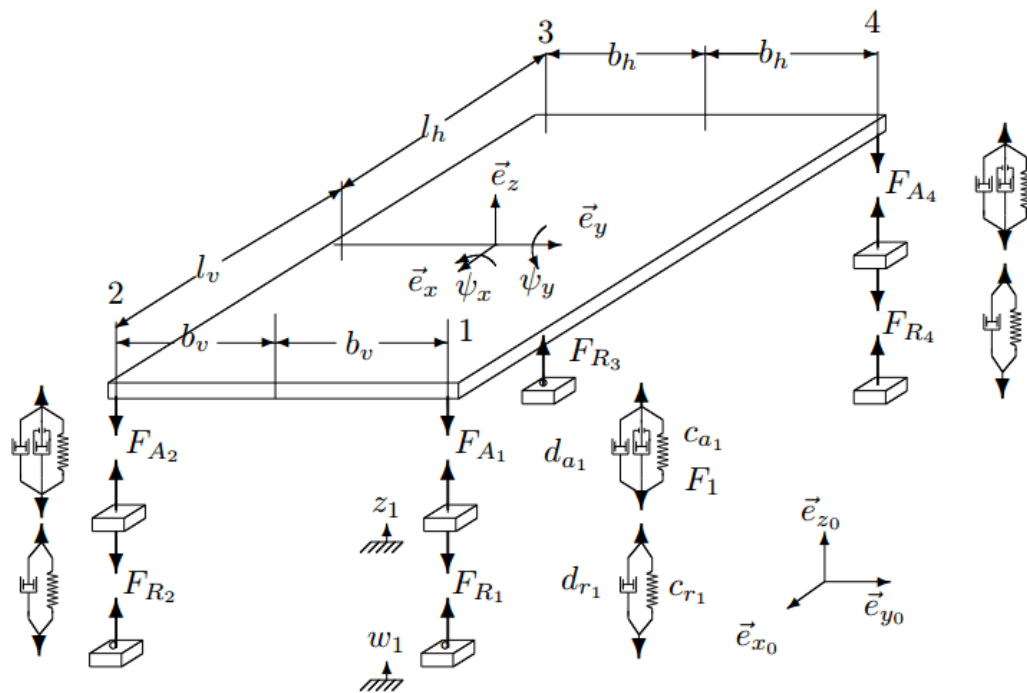


Figura 53. Estructura del modelo complejo.

FUENTE: (Oertel, Einführung in die Systemdynamik, 2008)



Figura 54. Uso de ABC en vehículos comerciales.

FUENTE: (Mercedes-Benz Active Body Control - Fully Active Suspension Technology, 2012)

Tabla 6.

Parámetros para el modelado de la suspensión.

Parámetros del subsistema			
<i>Param.</i>	<i>Valor</i>	<i>Unidades</i>	<i>Descripción</i>
ms	300	[kg]	Masa del vehículo.
mt	47.801	[kg]	Masa de la llanta.
ds	200	[N.s/m]	Coeficiente de amortiguamiento de la suspensión
dt	20	[N.s/m]	Coeficiente de amortiguamiento de la llanta.
cs	20000	[N/m]	Coeficiente de rigidez de la suspensión.
ct	200000	[N/m]	Coeficiente de rigidez de la llanta.
lv	1.3760	[m]	Distancia entre el eje delantero de las llantas y el centro de gravedad del vehículo
lh	0.9230	[m]	Distancia entre el eje posterior de las llantas y el centro de gravedad del vehículo
bv	0.7	[m]	Distancia entre el eje lateral (derecha) de las llantas y el centro de gravedad del vehículo.
bh	0.7	[m]	Distancia entre el eje lateral (izquierda) de las llantas y el centro de gravedad del vehículo.
Jx	500	[kg-m ²]	Momento de inercia del vehículo con respecto al eje de alabeo (X).
Jy	1000	[kg-m ²]	Momento de inercia del vehículo con respecto al eje de cabeceo (Y).

4.3.3.1. Fuerzas en las llantas

La suma total de las fuerzas F_T en cada elemento amortiguado y no amortiguado puede ser interpretado por la ecuación (4.4).

$$F_{T_j} = c_{T_j} (z_j - w_j) + d_{T_j} (\dot{z}_j - \dot{w}_j) \quad (4.4)$$

Al reemplazar los valores para cada neumático ($j = 1,2,3,4$) teniendo en cuenta que los valores de la rigidez y amortiguamiento de la llanta serán considerados iguales para los cuatro neumáticos.

$$\begin{aligned}
F_{T_1} &= c_T(z_1 - w_1) + d_T(\dot{z}_1 - \dot{w}_1) \\
F_{T_2} &= c_T(z_2 - w_2) + d_T(\dot{z}_2 - \dot{w}_2) \\
F_{T_3} &= c_T(z_3 - w_3) + d_T(\dot{z}_3 - \dot{w}_3) \\
F_{T_4} &= c_T(z_4 - w_4) + d_T(\dot{z}_4 - \dot{w}_4)
\end{aligned}$$

4.3.3.2. Fuerzas en la suspensión activa.

Fuerzas F_S que afectan a la suspensión en cada neumático siguen la ecuación (4.5).

$$F_{S_j} = c_{S_j}(z - z_j + \sum l\varphi) + d_{S_j}(\dot{z} - \dot{z}_j + \sum l\dot{\varphi}) + F_j \quad (4.5)$$

F_j determina la fuerza efectuada por el actuador del ABC para controlar la altura de la suspensión de cada llanta.

Reemplazando los valores para cada neumático ($j = 1,2,3,4$) teniendo en cuenta que los valores de la rigidez y amortiguamiento de la suspensión serán considerados iguales para los cuatro neumáticos.

$$\begin{aligned}
F_{S_1} &= c_S(z - z_1 - l_v\varphi_y + b_v\varphi_x) + d_S(\dot{z} - \dot{z}_1 - l_v\dot{\varphi}_y + b_v\dot{\varphi}_x) + F_1 \\
F_{S_2} &= c_S(z - z_2 - l_v\varphi_y - b_v\varphi_x) + d_S(\dot{z} - \dot{z}_2 - l_v\dot{\varphi}_y - b_v\dot{\varphi}_x) + F_2 \\
F_{S_3} &= c_S(z - z_3 + l_h\varphi_y - b_h\varphi_x) + d_S(\dot{z} - \dot{z}_3 + l_h\dot{\varphi}_y - b_h\dot{\varphi}_x) + F_3 \\
F_{S_4} &= c_S(z - z_4 + l_h\varphi_y + b_h\varphi_x) + d_S(\dot{z} - \dot{z}_4 + l_h\dot{\varphi}_y + b_h\dot{\varphi}_x) + F_4
\end{aligned}$$

Al usar el principio de conservación de fuerzas en cada llanta y en la estructura se obtendrán las siguientes ecuaciones:

$$m_{T_j}\ddot{z}_j = F_{S_j} - F_{T_j} \quad (4.6)$$

$$m_{T_1}\ddot{z}_1 = F_{S_1} - F_{T_1} \quad (4.7)$$

$$m_{T_2} \ddot{z}_2 = F_{S_2} - F_{T_2} \quad (4.8)$$

$$m_{T_3} \ddot{z}_3 = F_{S_3} - F_{T_3} \quad (4.9)$$

$$m_{T_4} \ddot{z}_4 = F_{S_4} - F_{T_4} \quad (4.10)$$

$$m_s \ddot{z} = -F_{S_1} - F_{S_2} - F_{S_3} - F_{S_4} \quad (4.11)$$

4.3.3.3. Efectos de roll y pitch en el vehículo

En este paso es importante considerar los efectos de la suspensión en un vehículo en movimiento, este movimiento causa un cambio en los ángulos de roll y pitch. Se efectuó una suma de momentos con referencia a los ejes X (roll) y Y (pitch) para obtener las ecuaciones (4.12) y (4.13)

$$\begin{aligned} J_x \ddot{\phi}_x &= \sum Mx \\ J_x \ddot{\phi}_x &= [RollMoment] + [TireForcesMoment] \\ J_x \ddot{\phi}_x &= [m_s \cdot hg \cdot g \cdot \phi_x - m_s \cdot hg \cdot (\ddot{y} + \dot{x} \cdot \dot{\psi} - \dot{z} \cdot \dot{\phi}_x)] + [-b_v F_{S_1} + b_v F_{S_2} + b_h F_{S_3} - b_h F_{S_4}] \\ J_x \ddot{\phi}_x &= m_s \cdot hg \cdot (g \cdot \phi_x - \ddot{y} - \dot{x} \cdot \dot{\psi} + \dot{z} \cdot \dot{\phi}_x) - b_v F_{S_1} + b_v F_{S_2} + b_h F_{S_3} - b_h F_{S_4} \end{aligned} \quad (4.12)$$

$$\begin{aligned} J_y \ddot{\phi}_y &= \sum My \\ J_y \ddot{\phi}_y &= [PitchMoment] + [TireForcesMoment] \\ J_y \ddot{\phi}_y &= [m_s \cdot hg \cdot g \cdot \phi_y - m_s \cdot hg \cdot (\ddot{x} - \dot{y} \cdot \dot{\psi} + \dot{z} \cdot \dot{\phi}_y)] + [l_v F_{S_1} + l_v F_{S_2} - l_h F_{S_3} - l_h F_{S_4}] \\ J_y \ddot{\phi}_y &= m_s \cdot hg \cdot (g \cdot \phi_y - \ddot{x} + \dot{y} \cdot \dot{\psi} - \dot{z} \cdot \dot{\phi}_y) + l_v F_{S_1} + l_v F_{S_2} - l_h F_{S_3} - l_h F_{S_4} \end{aligned} \quad (4.13)$$

4.3.3.4. Espacio de estados

Se utilizará la teoría de espacio de estados y diseño por ubicación de polos para el control activo de chasis. Las siete ecuaciones de movimiento (4.7), (4.8), (4.9), (4.10), (4.11), (4.12), (4.13) y sus respectivas derivadas formarán **catorce variables de estados** para el modelo, las cuales se mostrarán a continuación.

$$q(t) = \begin{matrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z \\ \varphi_x \\ \varphi_y \\ \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \\ \dot{z} \\ \dot{\varphi}_x \\ \dot{\varphi}_y \end{matrix} \quad \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix}$$

Luego de resolver las siete ecuaciones anteriormente nombradas, se las transformará en las siguientes matrices. M_m matriz de masa, B_m matriz de longitudes, X_m matriz de entradas, D_m matriz de amortiguación, C_m , matriz de rigidez.

$$M_m = \begin{bmatrix} m_T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & m_T & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & m_T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_T & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & m_s & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & J_x & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & J_y \end{bmatrix} \quad B_m = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 1 & 1 & 1 \\ b_v & -b_v & -b_h & b_h \\ -l_v & -l_v & l_h & l_h \end{bmatrix} \quad x_m = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z \\ \varphi_x \\ \varphi_y \end{bmatrix}$$

$$D_m = \begin{bmatrix} d_T + d_S & 0 & 0 & 0 & -d_S & -b_v d_S & l_v d_S \\ 0 & d_T + d_S & 0 & 0 & -d_S & b_v d_S & l_v d_S \\ 0 & 0 & d_T + d_S & 0 & -d_S & b_h d_S & -l_h d_S \\ 0 & 0 & 0 & d_T + d_S & -d_S & -b_h d_S & -l_h d_S \\ -d_S & -d_S & -d_S & -d_S & 4d_S & 0 & 2(l_h d_S - l_v d_S) \\ -b_v d_S & b_v d_S & b_h d_S & -b_h d_S & 0 & 2(b_h^2 d_S + b_v^2 d_S) & 0 \\ l_v d_S & l_v d_S & -l_h d_S & -l_h d_S & 2(l_h d_S - l_v d_S) & 0 & 2(l_h^2 d_S + l_v^2 d_S) \end{bmatrix}$$

$$C_m = \begin{bmatrix} c_T + c_S & 0 & 0 & 0 & -c_S & -b_v c_S & l_v c_S \\ 0 & c_T + c_S & 0 & 0 & -c_S & b_v c_S & l_v c_S \\ 0 & 0 & c_T + c_S & 0 & -c_S & b_h c_S & -l_h c_S \\ 0 & 0 & 0 & c_T + c_S & -c_S & -b_h c_S & -l_h c_S \\ -c_S & -c_S & -c_S & -c_S & 4c_S & 0 & 2(l_h c_S - l_v c_S) \\ -b_v c_S & b_v c_S & b_h c_S & -b_h c_S & 0 & 2(b_h^2 c_S + b_v^2 c_S) & 0 \\ l_v c_S & l_v c_S & -l_h c_S & -l_h c_S & 2(l_h c_S - l_v c_S) & 0 & 2(l_h^2 c_S + l_v^2 c_S) \end{bmatrix}$$

Un esquema básico del modelo en espacio de estados se muestre en la **Figura 55**. Las ecuaciones de espacio de estados están dadas por (4.14) y (4.15).

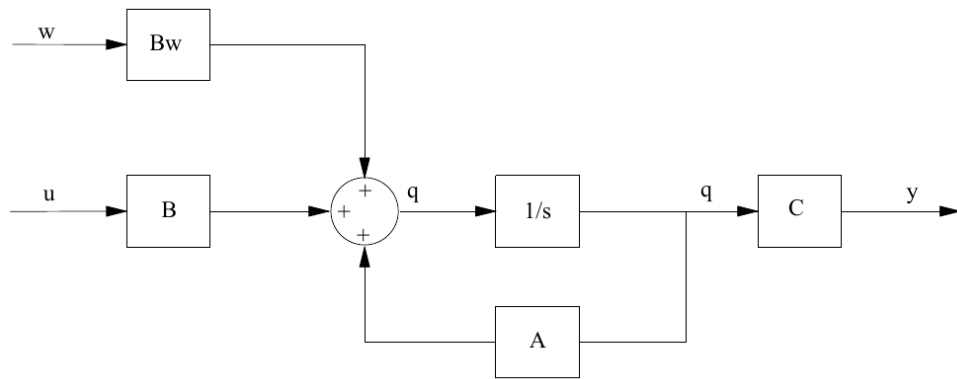


Figura 55. Diagrama de bloques del espacio de estados.

$$\dot{q}(t) = \underset{[14 \times 1]}{A} \cdot \underset{[14 \times 1]}{q}(t) + \underset{[14 \times 1]}{B} \cdot \underset{[4 \times 1]}{u}(t) + \underset{[14 \times 1]}{B_w} \cdot \underset{[4 \times 1]}{w}(t) + \underset{[14 \times 1]}{[Mw]} \tag{4.14}$$

$$y(t) = \underset{[4 \times 14]}{C} \cdot \underset{[14 \times 1]}{q}(t) + \underset{[4 \times 14]}{D} \cdot \underset{[4 \times 1]}{u}(t) \tag{4.15}$$

Nomenclatura las estas ecuaciones es:

$q(t)$: Vector de estados	A: Matriz de estados
$u(t)$: Vector de entradas	B: Matriz de entradas
$y(t)$: Vector de salidas	C: Matriz de salidas
$w(t)$: Vector de perturbaciones	B_w : Matriz de perturbaciones

Como se mencionó con anterioridad, $u(t)$ consiste de los valores de entrada del controlador los cuales son las fuerzas del actuador para cada llanta F_j , $y(t)$ consiste de los valores de salida los cuales son las alturas de la suspensión activa en cada llanta Z y $w(t)$ representa las perturbaciones en el camino.

El elemento $[Mw]$ de la ecuación (4.14) representa la matriz que contiene el valor de roll y pitch de la [sección](#). especialmente las variables $[RollMoment]$ y $[PitchMoment]$ las cuales no fueron consideradas en las matrices previamente descritas. Este proceso se puede observar en el bloque Mxy de la **Figura 63**.

Finalmente, al reemplazar los valores de matrices se obtendrán las siguientes ecuaciones de estado las cuales están descritas en la forma canónica controlable.

$$\dot{q}(t) = \begin{bmatrix} 0 & E \\ -M_m^{-1}C_m & -M_m^{-1}D_m \end{bmatrix} q(t) + \begin{bmatrix} 0 \\ -M_m^{-1}B_m \end{bmatrix} u(t) + B_w w(t) + [Mw] \quad (4.16)$$

$[14 \times 1]$ $[14 \times 14]$ $[14 \times 1]$ $[14 \times 1]$ $[4 \times 1]$ $[14 \times 1]$ $[4 \times 1]$ $[14 \times 1]$

$$\dot{q}(t) = \begin{bmatrix} 0 & E \\ -M_m^{-1}C_m & -M_m^{-1}D_m \end{bmatrix} q(t) + \begin{bmatrix} 0 \\ -M_m^{-1}B_m \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} + B_w \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} + [Mw] \quad (4.17)$$

4.3.4. Controlabilidad en espacio de estados

Si existe una entrada que permite llevar x_0 a x_f , el sistema es controlable. El sistema de dimensión n cuya ecuación de estado es la ecuación (4.14), es **controlable** si y solo si, la matriz de controlabilidad Q es de rango n .

$$Q = [B \quad AB \quad A^2B \quad \dots \quad A^{n-1}B]$$

4.3.5. Observabilidad en espacio de estados

Si de los valores de entrada y salida (u , y), puedo obtener un estado x_0 dentro de un rango de tiempo definido, el sistema es **observable**. Esto sucede cuando la matriz de observabilidad P es de rango n .

$$P = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

4.3.6. Diseño por ubicación de polos

En un sistema de la forma:

$$\dot{q} = A \cdot q + B \cdot u \quad (4.18)$$

Se implementará un **regulador** mediante la ley de control:

$$u = -K \cdot q \quad (4.19)$$

Esta ley de control indica que la señal de control es la realimentación negativa de la multiplicación entre una matriz de ganancia de realimentación del estado K y el vector de estados. Es importante mencionar que el valor de referencia r es cero ya que se desea que las salidas y variables de estado tiendan a cero.

Reemplazando la ley de control (4.19) en la ecuación (4.18) se obtiene:

$$\begin{aligned}\dot{q} &= A \cdot q - B \cdot K \cdot q \\ \dot{q} &= (A - B \cdot K_R) q\end{aligned}\tag{4.20}$$

La ecuación (4.20) representa el sistema ya compensado por la matriz K_R .

Los valores propios de $(A - B \cdot K_R)$ coinciden con los polos en lazo cerrado del sistema, por lo tanto, se desea obtener una matriz de ganancias K_R de tal manera que sea posible colocar los valores propios de $(A - B \cdot K_R)$ en los polos deseados para el sistema, los cuales deben cumplir las especificaciones del sistema de control requerido. Para que se cumpla esto, el sistema debe ser controlable. Un método para calcular la matriz de ganancias K_R es utilizando la fórmula de Ackermann, usando el comando en MATLAB **place()** o el comando **acker()**.

4.3.7. Observadores del estado

En ocasiones, el estado no puede ser medido directamente por lo que es necesario implementar un **observador** o **estimador** para que reconstruya el vector de estados, basándose en los valores medidos de las salidas (y) y de la señal de control (u). El modelo del sistema y el modelo del estimador se ejecutarán paralelamente.

En un sistema de la forma:

$$\begin{aligned}\dot{q} &= A \cdot q + B \cdot u \\ y &= C \cdot q\end{aligned}\tag{4.21}$$

El estimador tiene la estructura:

$$\dot{q} = F \cdot q + H \cdot u + G \cdot y\tag{4.22}$$

Se considerará que:

$$H = B$$

$$F = A - G \cdot C$$

Donde G, es la matriz transpuesta de ganancia del estimador, luego reemplazando en (4.22) se obtiene:

$$\dot{q} = (A - G \cdot C)q + B \cdot u + G \cdot y$$

$$\dot{q} = A \cdot q - G \cdot C \cdot q + B \cdot u + G \cdot y$$

$$\dot{q} = A \cdot q + B \cdot u + G \begin{pmatrix} y - C \cdot q \\ y \end{pmatrix}$$

$$\dot{q} = A \cdot q + B \cdot u + G(y - y)$$

$$\dot{q} = A \cdot q + B \cdot u + K_B^T (y - y) \quad (4.23)$$

A la expresión $K_B^T (y - y)$ se la considera como un término corrector.

Si las condiciones iniciales $x(0)$ y $\hat{x}(0)$ son iguales el error es cero, pero si son diferentes el error tenderá a cero. El error de estimación se puede definir como:

$$e = q - \hat{q}$$

$$\dot{e} = \dot{q} - \dot{\hat{q}}$$

$$\dot{e} = A + \cancel{B \cdot u} - A \cdot \hat{q} - \cancel{B \cdot u} - K_B^T (y - \hat{y})$$

$$\dot{e} = A(q - \hat{q}) - K_B^T (C \cdot q - C \cdot \hat{q})$$

$$\dot{e} = A(q - \hat{q}) - K_B^T \cdot C(q - \hat{q})$$

$$\dot{e} = (A - K_B^T \cdot C)(q - \hat{q})$$

$$\dot{e} = (A - K_B^T \cdot C)e \quad (4.24)$$

Por lo tanto, $(A - K_B^T \cdot C)$ representa la ecuación característica del error y sus valores propios son los polos del observador, los cuales determinan la dinámica del error. La matriz K_B debe elegirse de manera que la respuesta del observador sea más rápida que la del sistema (de 2 a 4 veces más rápida).

Un método para calcular la matriz de ganancias K_B es utilizando la fórmula de Ackermann, para esto se utilizará el comando en MATLAB **place()** o el comando **acker()**.

4.3.8. Implementación del controlador y del observador

Se usará el observador para producir una estimación de q , de manera que: (Rodríguez & Bordóns, 2005)

$$u = -K_R \cdot q \quad (4.25)$$

Sea el sistema (4.21) controlado por (4.25), la ecuación de estado puede reescribirse como:

$$\begin{aligned} \dot{q} &= A \cdot q + B \cdot u \\ \dot{q} &= A \cdot q - B \cdot K_R \cdot q \\ \dot{q} &= (A - B \cdot K_R)q + B \cdot K_R(q - q) \\ \dot{q} &= (A - B \cdot K_R)q + B \cdot K_R \cdot e \end{aligned} \quad (4.26)$$

Donde e es el error de observación dado por la ecuación: (Rodríguez & Bordóns, 2005)

$$\dot{e} = (A - K_B^T \cdot C)e \quad (4.27)$$

Para describir la dinámica del sistema completo (sistema – controlador observador) se combinará la ecuación de estado y la del error.

$$\begin{bmatrix} \dot{q} \\ \dot{e} \end{bmatrix} = \begin{bmatrix} A - B \cdot K_R & B \cdot K_R \\ 0 & A - K_B^T \cdot C \end{bmatrix} \begin{bmatrix} q \\ e \end{bmatrix}$$

Donde la ecuación característica del sistema será: (Rodríguez & Bordóns, 2005).

$$\begin{vmatrix} s \cdot I - A + B \cdot K_R & -B \cdot K_R \\ 0 & s \cdot I - A + K_B^T \cdot C \end{vmatrix} = 0$$

$$|s \cdot I - A + B \cdot K_R| |s \cdot I - A + K_B^T \cdot C| = 0 \quad (4.28)$$

Las raíces de la ecuación (4.28) son los polos del sistema en lazo cerrado, tanto de los polos colocados por el controlador como de los polos del observador.

4.3.8.1. Análisis de controlabilidad y observabilidad

Para analizar la controlabilidad y la observabilidad del sistema se utilizarán los comandos **ctrb(A,B)** y **obsv(A,C)** respectivamente.

```

% ----- Controlabilidad
Qb = ctrb(A,B);
Controlador = rank(Qb);

% ----- Observabilidad
Qc = obsv(A,C);
Observador = rank(Qc);

```

Controlador =	
14	
Observador =	
14	

Figura 56. Comprobación de controlabilidad y observabilidad.

Una vez realizado el análisis previo, se concluye que el sistema es controlable y observable, por lo que es posible la implementación de un controlador y un estimador del vector de estados, que serán determinados por las matrices A, B C y la diferencia en la salida real y la salida estimada (error de estimación).

4.3.8.2. Polos deseados para el controlador

Los polos deseados para el sistema se detallan a continuación y tienen las especificaciones de la **Figura 57**.

```
% ----- Observador polos
a1 = -90.0D+00; b1 = 150.0D+00;
a2 = -90.0D+00; b2 = 150.0D+00;
a3 = -90.0D+00; b3 = 150.0D+00;
a4 = -90.0D+00; b4 = 150.0D+00;
a5 = -30.0D+00; b5 = 30.0D+00;
a6 = -30.0D+00; b6 = 30.0D+00;
a7 = -30.0D+00; b7 = 10.0D+00;
```

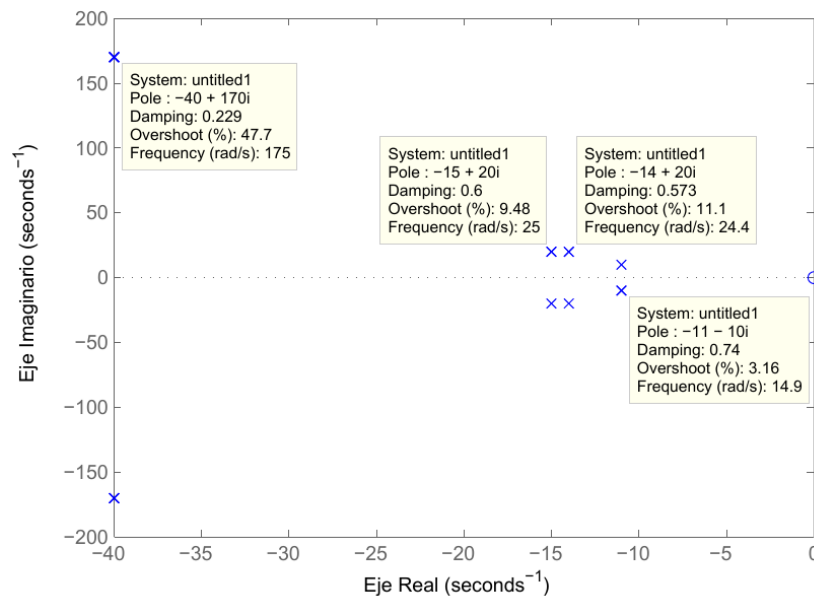


Figura 57. Mapa polos y ceros del Controlador.

4.3.8.3. Polos deseados para el observador

Los polos deseados para el observador se detallan a continuación y tienen las especificaciones de la **Figura 58**.

```
% ----- Observador polos
a1 = -90.0D+00; b1 = 150.0D+00;
a2 = -90.0D+00; b2 = 150.0D+00;
a3 = -90.0D+00; b3 = 150.0D+00;
a4 = -90.0D+00; b4 = 150.0D+00;
a5 = -30.0D+00; b5 = 30.0D+00;
a6 = -30.0D+00; b6 = 30.0D+00;
a7 = -30.0D+00; b7 = 10.0D+00;
```

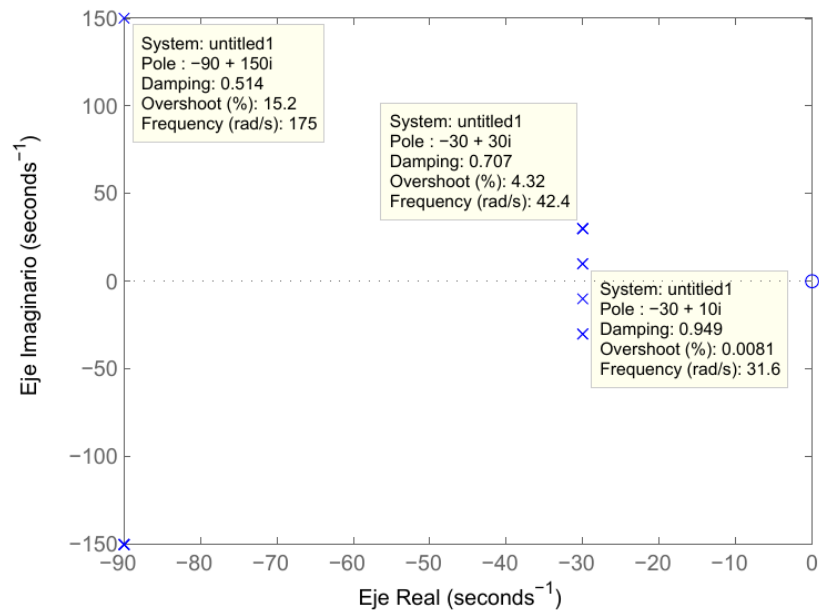


Figura 58. Mapa polos y ceros del Observador

4.3.8.4. Diagrama de Bloques

En la **Figura 59** se puede observar el diagrama de bloques del sistema completo con controlador y con observador, esto se implementará posteriormente en Simulink.

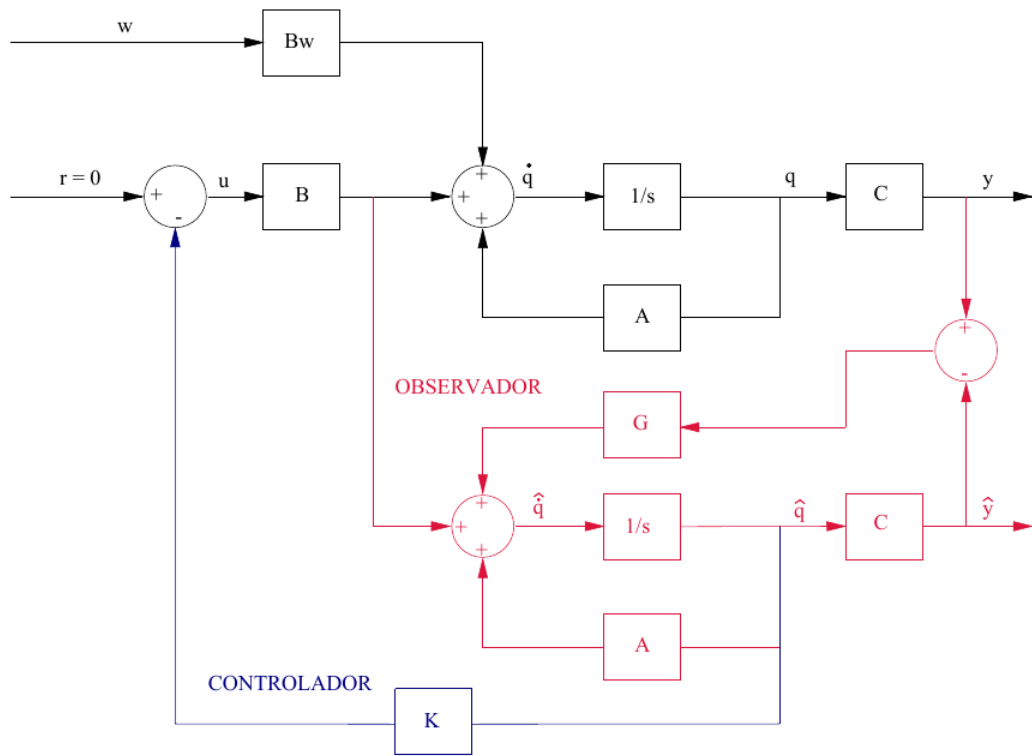


Figura 59. Diagrama de bloques del sistema completo.

4.3.9. Modelo en Simulink

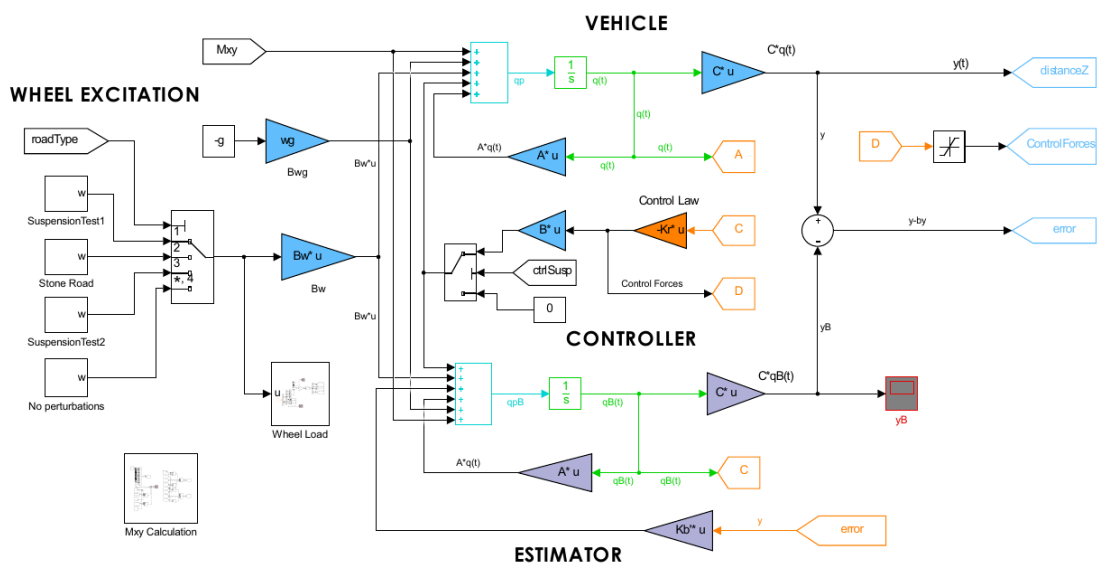


Figura 60. Modelo del sistema de suspensión en Simulink.

4.4. PARÁMETROS DE SIMULACIÓN DEL MODELO

Una interfaz gráfica fue elaborada para configurar y cambiar los parámetros del modelo de la simulación (véase **Figura 61**).

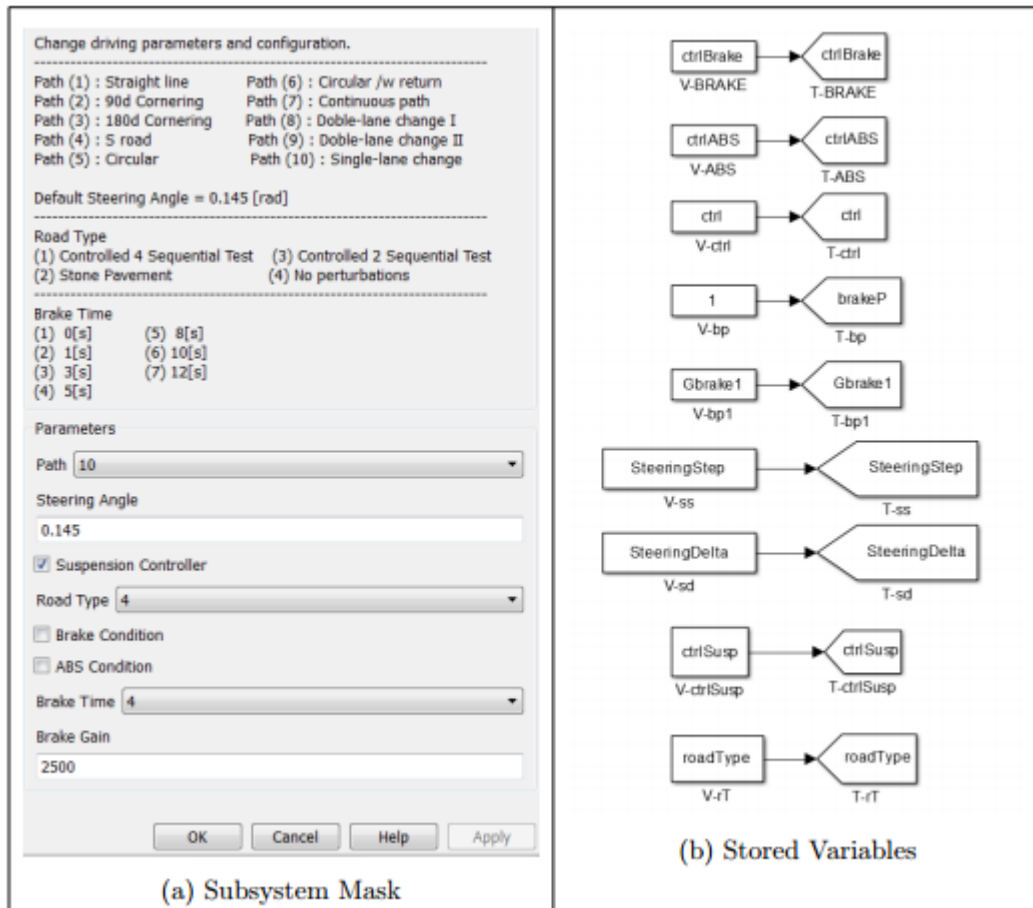


Figura 61. Interfaz para la edición de parámetros de simulación.

Esta interfaz contiene:

- **Selección de la trayectoria:** Esta opción permite al usuario elegir una trayectoria de vehículo previamente programada. Estas trayectorias pueden verse en la **Figura 62** y **Figura 63**.
- **Angulo de dirección:** Es el ángulo que el vehículo se varia rotacionalmente para cumplir la trayectoria deseada. El valor predefinido es de 0.145 radianes.

- **Condición de control de suspensión:** Habilita o deshabilita el controlador de la suspensión.
- **Tipo de camino/vía:** Esta opción permite al usuario elegir las perturbaciones previamente programadas que tendrá el camino. Este comportamiento se puede observar en la **Figura 64**.
- **Condición de frenado:** Habilita o deshabilita los frenos.
- **Condición de ABS:** Habilita o deshabilita el ABS.
- **Tiempo de frenado:** Esta opción permite al usuario elegir el tiempo en que el que el vehículo empieza a frenar. En este modelo es posible elegir los siguientes tiempos $t = [0,1,3,5,8,10,12]$ [s]
- **Ganancia para el freno:** Este valor modifica la presión de frenado ejercida en los neumáticos, deteniendo completamente al vehículo en mayor o menor tiempo. Valor predefinido = 2500.

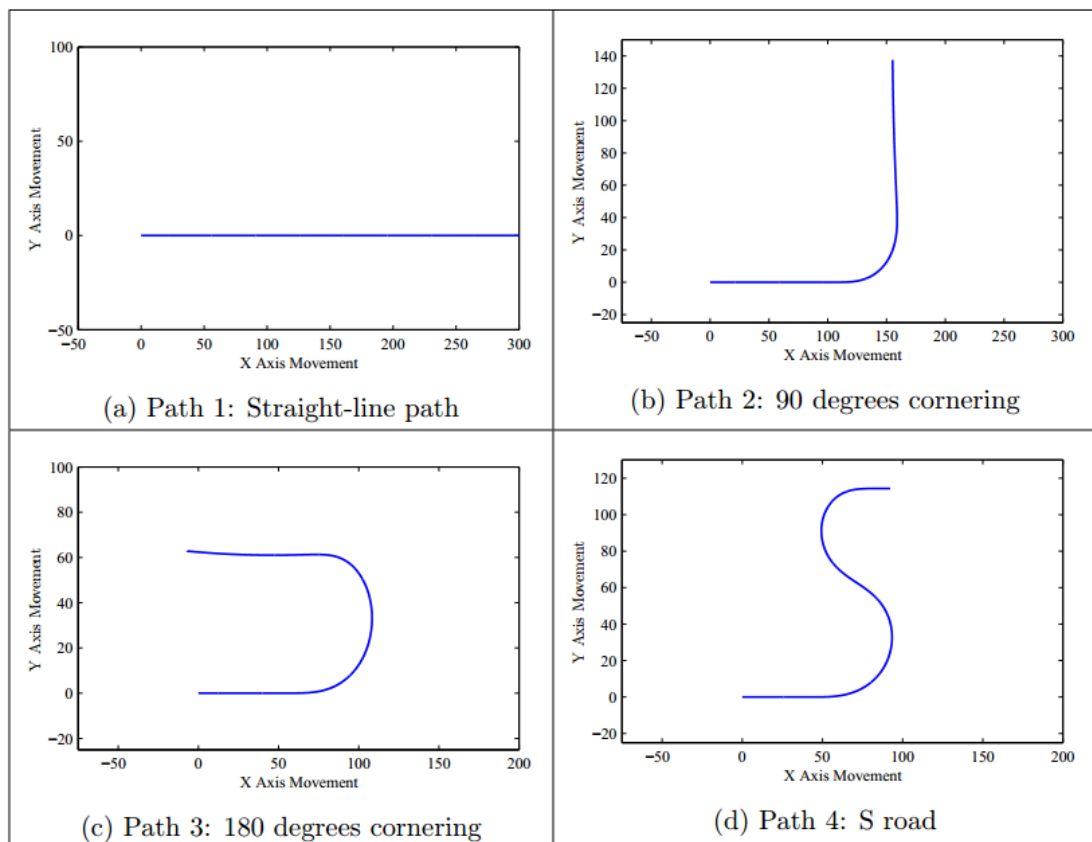


Figura 62. Trayectorias básicas.

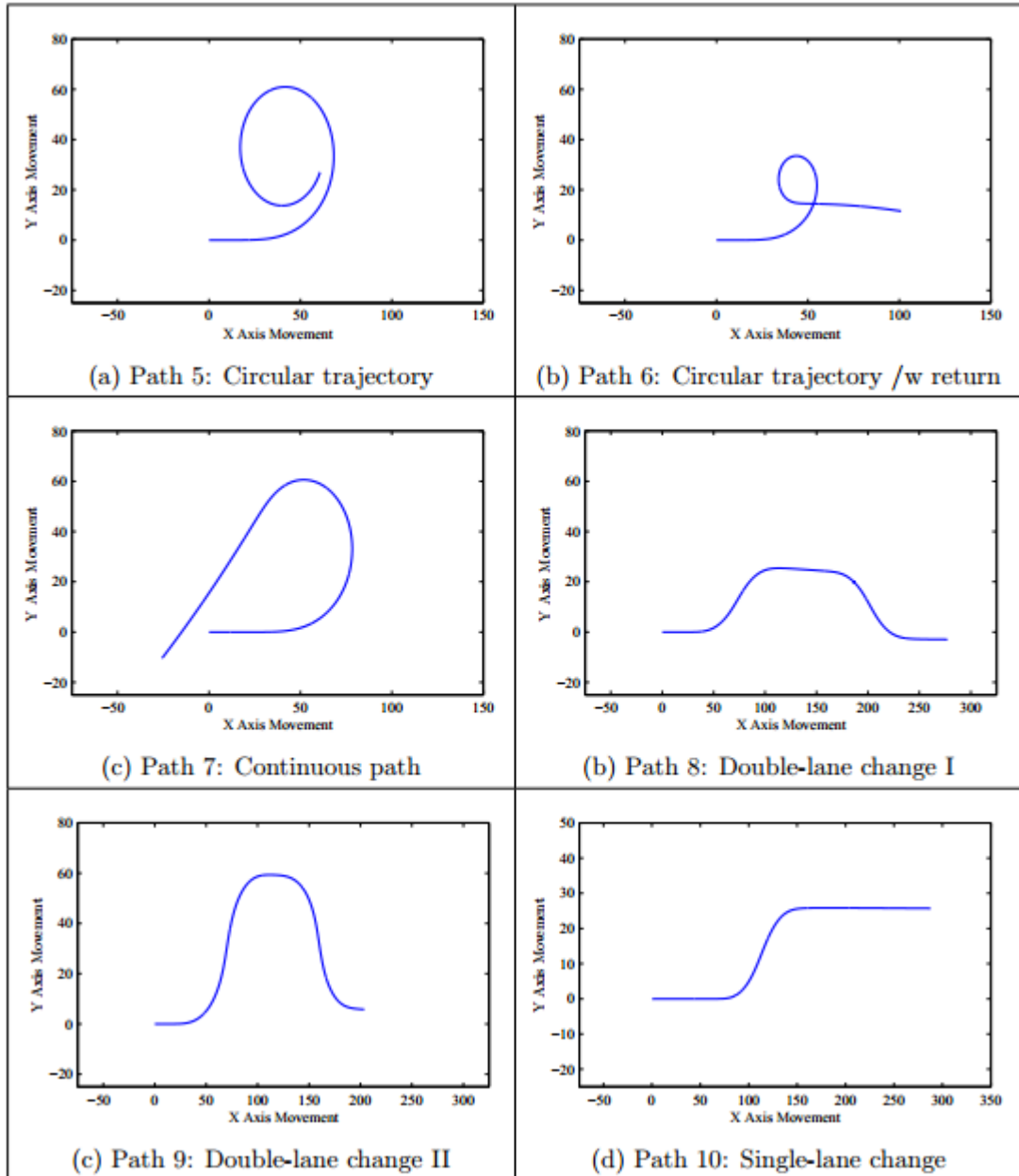


Figura 63. Trayectorias circulares y de cambio de carril.

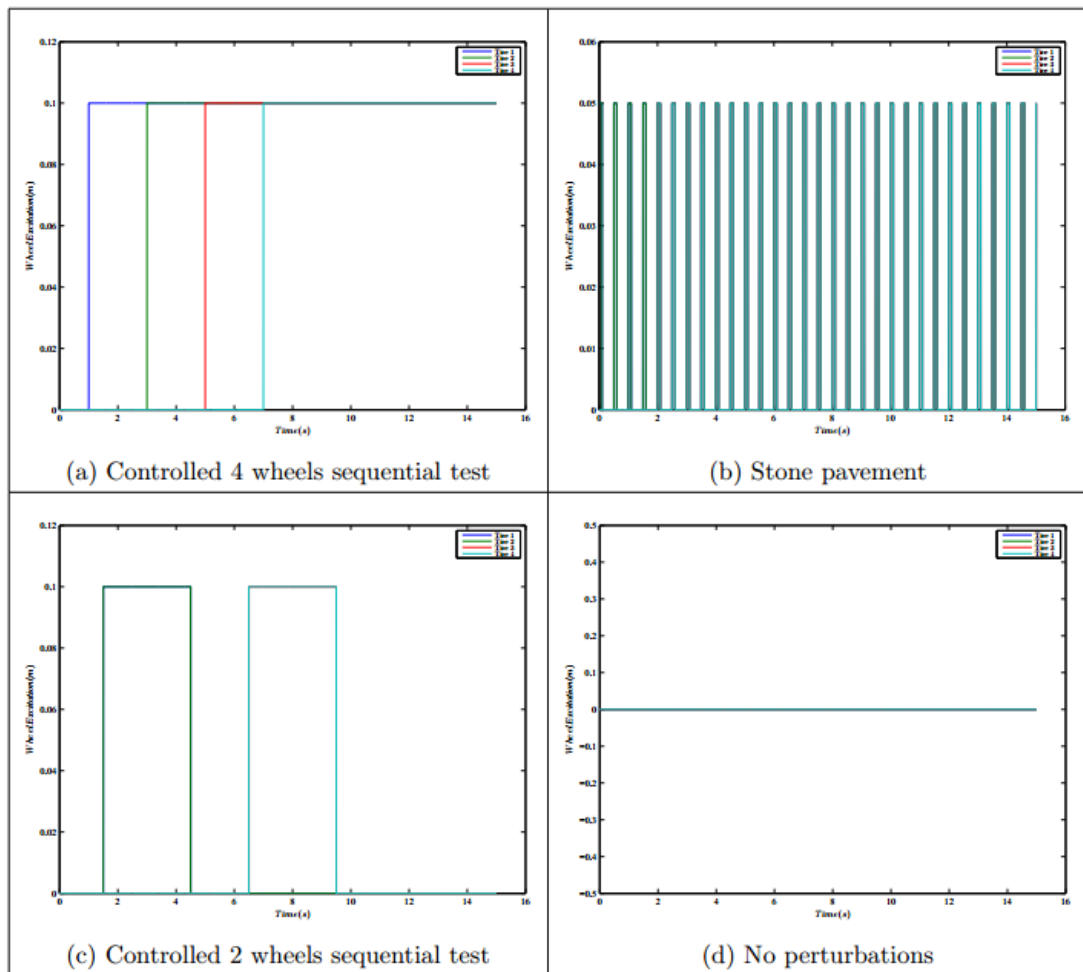


Figura 64. Perturbaciones en el camino.

CAPÍTULO 5

5. TRANSFERENCIA DEL MODELO A LA PLATAFORMA DSPACE.

5.1. DSPACE CONTROL DESK

DSPACE ControlDesk es una herramienta potente para aplicaciones en tiempo real, incluye funciones como: herramientas de visualización, edición de la interfaz de la instrumentación, medición, pos procesado y la posibilidad de enlazar modelos en Simulink mediante el uso de funciones de alto nivel de MATLAB, facilitando la lectura de señales de sensores y la generación de señales de control. DSPACE Control Desk se enlaza con Simulink usando el toolbox “Real Time Interface (RTI)” el cual provee de bloques y funciones para configurar modelos. Estos bloques permiten al usuario acceder al hardware DSPACE. Soporta las siguientes versiones de MATLAB: r2012a, r2011a, r2010a, r2010b, 2009Bsp1 (Ghaffari, 2009).

Otras funciones que este software tiene son (dSPACE GmbH, 2016):

- Módulo de simulación de fallas para controlar unidades de inserción de fallas (FIU) en tiempo real.
- Monitoreo del bus de comunicación.
- Sincronización en la captura de datos a lo largo de las plataformas ECU, RCP, HIL y sistemas de bus.

A continuación, se van a describir los pasos necesarios para implementar un sistema de control usando la tarjeta DSPACE 1006.

- Modelar la planta en Simulink utilizando el toolbox RTI.
- Configurar la simulación y los parámetros del modelo.
- Compilar y generar código C del modelo.
- Establecer la conexión entre el computador anfitrión y el equipo DSPACE.
- Crear una interfaz de control para el experimento.

- Realizar pruebas en tiempo real del experimento: ControlDesk puede ser usado para monitorear el proceso de la simulación, ajustar parámetros en línea y capturar datos.

5.2. TRANSFERENCIA DEL MODELO DE VEHÍCULO A DSPACE

Como se mencionó anteriormente, el software ControlDesk es utilizado como una interfaz entre el equipo DS1006 y el computador anfitrión. Una vez que el modelo en Simulink haya sido terminado se deberá transferir al equipo, a continuación, se mencionarán los pasos que se siguieron para este propósito.

5.2.1. Definición del directorio en MATLAB

Es importante definir el directorio en MATLAB donde estarán ubicados todos los archivos del proyecto, esto también incluye las librerías y archivos necesarios para el correcto funcionamiento del toolbox RTI. Se definieron las siguientes rutas:

- C:\dSPACE RCPHIL 7.2(2)
- C:\Program Files(x86)\dSPACE\
- C:\Program Files(x86)\Common Files\dSPACE\

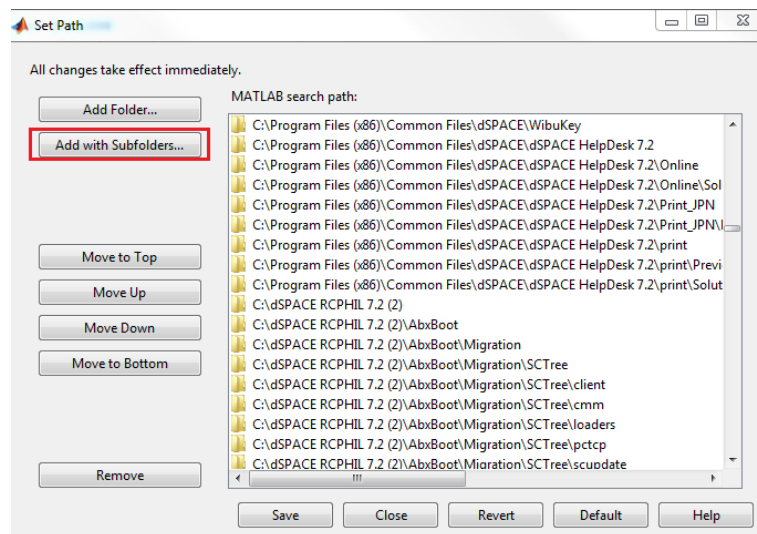


Figura 65. Definición del directorio de MATLAB.

5.2.2. Habilitación de la plataforma RTI

El mensaje de la **Figura 66** aparecerá al redactar el comando **rti1006** en MATLAB. Este comando habilita el uso de la plataforma RTI, en este caso para el equipo DS1006.

```
Switching RTI platform support from  to RTI1006, please wait ...

=====
Configuring dSPACE(R) Software for MATLAB(R) 7.11.0.584 (R2010b) ...

RTI           Real-Time Interface to Simulink (RTI1006)  6.8           02-Nov-2011 okay
RTIFPGA       RTI FPGA Programming Blockset                       2.2           02-Nov-2011 okay
MLIB/MTRACE   MATLAB-dSPACE Interface Libraries                   4.7.3        02-Nov-2011 okay
=====

*** RTI Platform Support RTI1006 activated.
*** Some Configuration Preferences are unsuitable for use with RTI.
    Details: RTIPrefMismatch.txt
    Wizard: Click here to invoke dialog based configuration now.
```

Figura 66. Respuesta al comando rti1006.

5.2.3. Habilitación del toolbox RTI para el equipo DS1006

La pantalla de la **Figura 67** aparecerá al redactar el comando **rti** en MATLAB. Esta ventana contiene todas las funciones, bloques, ejemplos y demostraciones del toolbox RTI, permitiendo la conexión con múltiples procesadores y tarjetas de adquisición de datos.

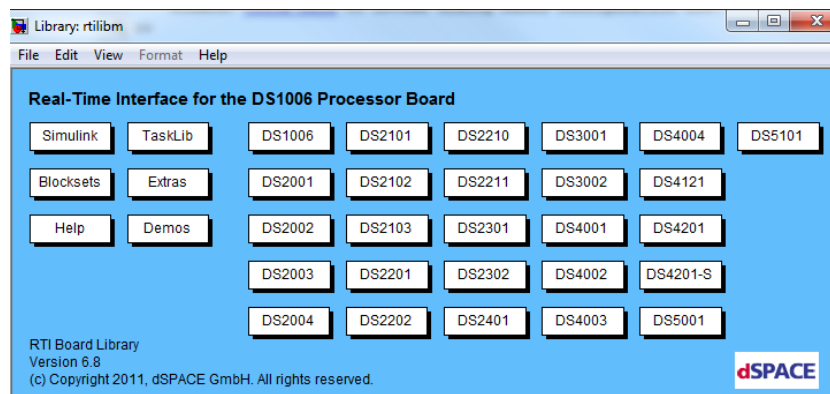


Figura 67. Respuesta al comando rti.

5.2.4. Configuración de los parámetros del toolbox y de la simulación.

En la pestaña *Real-Time Workshop* se utilizó la configuración de la **Figura 68** y en la pestaña *Optimización* se desactivó la casilla de reducción de bloque.

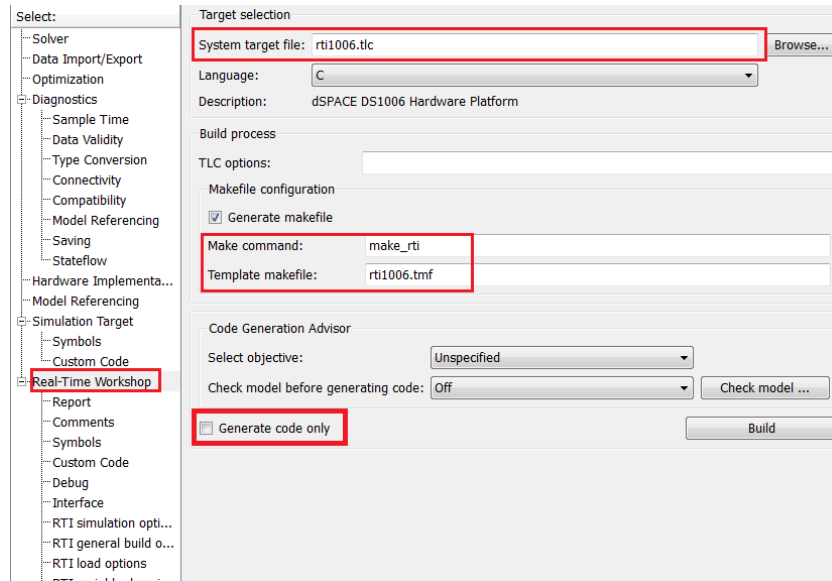


Figura 68. Configuración de los parámetros RTI.

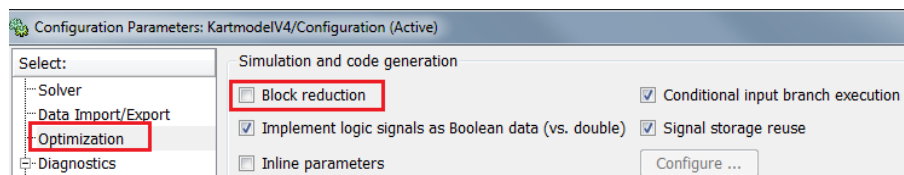


Figura 69. Configuración de parámetros de optimización.

5.2.5. Compilación y generación del código del modelo.

Una vez terminada la configuración, el modelo puede ser cargado y compilado al equipo DS1006. Se puede compilar el modelo dirigiéndose a *Herramientas > Generación de código > Compilar modelo* o simplemente al presionar CTRL + B.

Un mensaje aparecerá indicando que la compilación fue satisfactoria (véase **Figura 70**). Este paso compila el modelo en Simulink en código C admisible para el equipo DSPACE. Este proceso también genera un archivo con extensión **.sdf (VARIABLE FILE)** el cual contiene todas las variables, señales, entradas y salidas del modelo.

```

.....
*** Found User-Code File KartmodelV4_usr.c from 06-Jun-2016 16:22:18
*** Found User Makefile KartmodelV4_usr.mk from 06-Jun-2016 16:22:18
*** Optional User Variable Description File KartmodelV4_usr.trc not available
.
### Processing Template Makefile: C:\dSPACE RCPHIL 7.2 (2)\MATLAB\RTI1006\M\rti1006.tmf
### KartmodelV4.mk which is generated from C:\dSPACE RCPHIL 7.2 (2)\MATLAB\RTI1006\M\rti1006.tmf
### Building KartmodelV4: dsmake -f KartmodelV4.mk GENERATE_REPORT=0 EXTMODE_STATIC_ALLOC

BUILDING APPLICATION "KartmodelV4" (Single Timer Task Mode)

WORK DIRECTORY "F:\files\6June6\MATLAB_Simulink\v4 imp\func"
BUILD DIRECTORY "F:\files\6June6\MATLAB_Simulink\v4 imp\func\KartmodelV4_rti1006"
TARGET COMPILER "C:\dSPACE RCPHIL 7.2 (2)\Compiler\x86Tools22" (DS1006 Compiler Vs 2.2, GC

COMPILING "KartmodelV4.c"
COMPILING "KartmodelV4_data.c"
COMPILING "KartmodelV4_trc_ptr.c" |
COMPILING "rtGetInf.c"
COMPILING "rtGetNaN.c"
COMPILING "C:\PROGRA-2\MATLAB\R2010b\rtw\c\src\rt_matrx.c"
COMPILING "rt_nonfinite.c"
COMPILING "rt_pow_snf.c"
COMPILING "C:\dSPACE RCPHIL 7.2 (2)\MATLAB\RTI1006\C\rt_printf.c"
COMPILING "C:\dSPACE RCPHIL 7.2 (2)\MATLAB\RTI1006\C\rti_sim_engine.c"

LINKING APPLICATION ...
LINKING FINISHED

LOADING APPLICATION "KartmodelV4.sdf" ...
[#1] ds1006 - RTLIB: Application is located in RAM. (0)
[#2] ds1006 - RTLIB: DS1006 serial number: 239084 (0)
[#3] ds1006 - RTI: Initializing ... (720)
[#4] ds1006 - RTI: Initialization completed (721)
[#5] ds1006 - RTI: Simulation state: RUN (700)
LOADING FINISHED

MAKE PROCESS SUCCEEDED

### Successful completion of Real-Time Workshop build procedure for model: KartmodelV4
*** Finished RTI build procedure for model KartmodelV4

```

Figura 70. Compilado del modelo.

5.2.6. Conexión con el equipo DS1006 y con el software ControlDesk.

Un mensaje diciendo que la conexión con ControlDesk ha sido exitosa aparecerá una vez que la compilación haya finalizado (véase **Figura 71**).

```

Framework: Initializing AppServerManager.Component
Framework: Initializing PythonInterpreter.Component
Framework: Initializing CommonTextEditor.Component
Framework: Initializing Experiment.Component
Framework: Initializing FailureSim.Component
Framework: Initializing HierarchyBrowser.Component
Framework: Initializing Instrumentation.Component
Framework: Initializing ReferenceData.Component
Framework: Initializing SCOUT.Component
Platform: Connecting to the bus ...
Platform: ... completed.
Platform: Establishing the connection.
Framework: Initializing TrcParser.Component
Framework: Initializing BusNavigator.Component
Platform: Loaded application F:\files\6June6\MATLAB_Simulink\w4 imp\func\kartmodelv4.sdf to ds1006.
Variable Browser: Loading file F:\files\6June6\matlab_simulink\w4 imp\func\kartmodelv4.sdf' ...
SDF File Parser: Loading file F:\files\6June6\matlab_simulink\w4 imp\func\kartmodelv4.trc' ...
Trace File Parser: Loading file F:\files\6June6\matlab_simulink\w4 imp\func\kartmodelv4.map' ...
Trace File Parser: ... successful.
SDF File Parser: ... successful.
Variable Browser: ... finished.
Real-Time Processor: [#3] ds1006 - RTI: Initializing ... (720)
Real-Time Processor: [#4] ds1006 - RTI: Initialization completed (721)
Real-Time Processor: [#5] ds1006 - RTI: Simulation state: RUN (700)
Real-Time Processor: [#6] ds1006 - RTI: Simulation Stop time reached (703)
Real-Time Processor: [#7] ds1006 - RTI: Simulation state: STOP (702)

```

Figura 71. Conexión con ControlDesk.

5.2.7. Creación de un nuevo experimento en ControlDesk

Es importante grabar en el mismo directorio que el modelo en Simulink y elegir la opción DS1006 como principal procesador de la simulación.

5.2.8. Cargar el archivo compilado al equipo DS1006

Elegir la opción Plataforma en la ventana de navegación, luego arrastrar el archivo .sdf del directorio y llevarlo a ds1006 como se ve en la **Figura 72**. Aceptar cualquier mensaje posterior.

5.2.9. Creación de la interfaz grafica

Se creó una ventana nueva en *Archivo >> Nuevo >> Layout*, en la opción Adquisición de información existen algunos elementos e instrumentos para la interfaz como check-boxes, sliders, botones, etc. (véase la **Figura 73**)

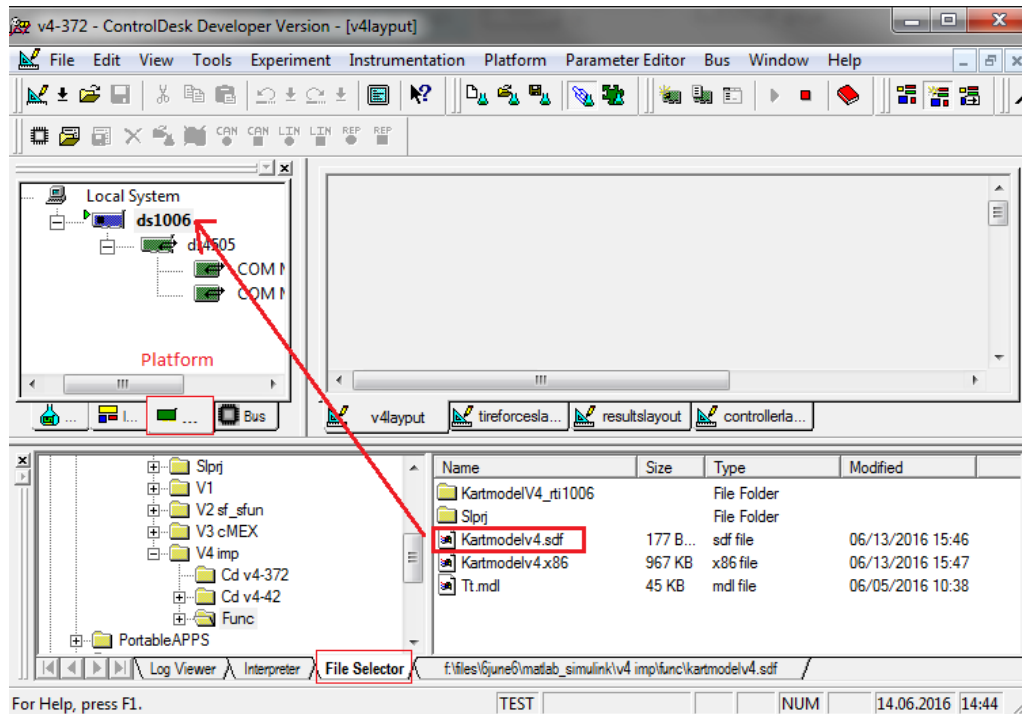


Figura 72. Cargar el archivo del modelo al procesador DSPACE.

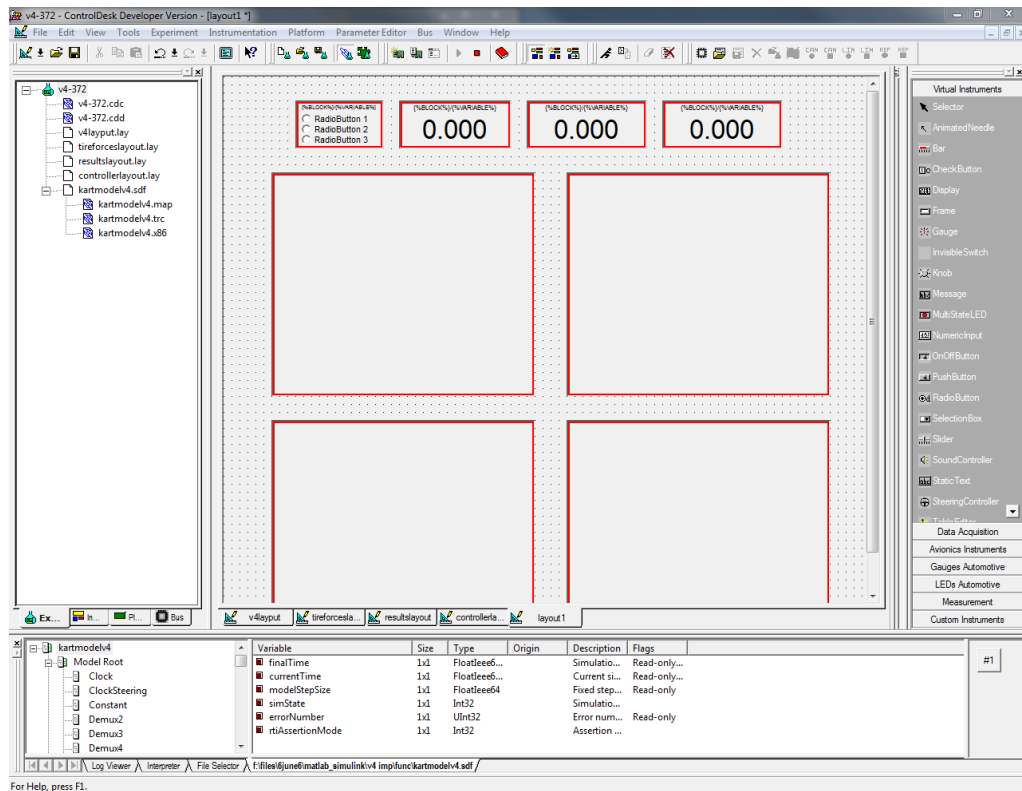
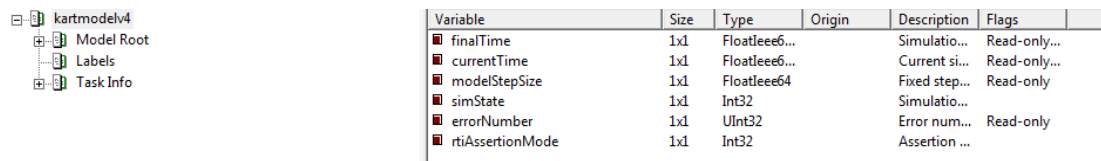


Figura 73. Creando la interfaz.

5.2.10. Enlace de las variables que se quieren medir

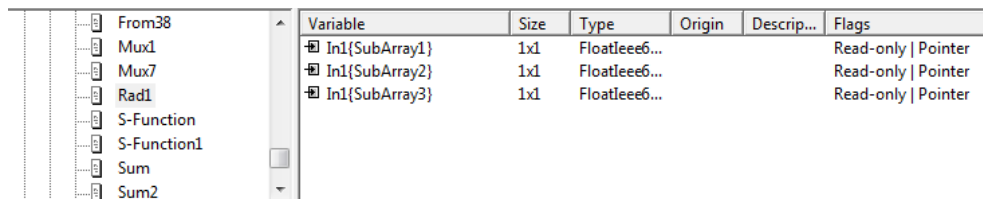
Luego de cargar el archivo **.sdf**, una nueva pestaña aparecerá en la ventana inferior llamada Administrador de variables, esta pestaña contiene todas las variables y señales del modelo en Simulink y también algunas variables para uso interno del simulador (véase **Figura 74**).



Variable	Size	Type	Origin	Description	Flags
finalTime	1x1	FloatIEEE6...		Simulatio...	Read-only...
currentTime	1x1	FloatIEEE6...		Current si...	Read-only...
modelStepSize	1x1	FloatIEEE64		Fixed step...	Read-only
simState	1x1	Int32		Simulatio...	
errorNumber	1x1	UInt32		Error num...	Read-only
rtiAssertionMode	1x1	Int32		Assertion ...	

Figura 74. Variables del modelo.

Para insertar las variables del modelo con su respectivo instrumento hacer clic en una variable y arrastrarla al instrumento deseado (véase **Figura 75**).



Variable	Size	Type	Origin	Descrip...	Flags
In1{SubArray1}	1x1	FloatIEEE6...			Read-only Pointer
In1{SubArray2}	1x1	FloatIEEE6...			Read-only Pointer
In1{SubArray3}	1x1	FloatIEEE6...			Read-only Pointer

Figura 75. Variables del modelo.

Una vez hecho esto se obtendrá como resultado la interfaz de la **Figura 77**.

5.2.11. Configuración de parámetros en la interfaz gráfica.

Cambiar los parámetros de acuerdo a los límites de los datos y el tiempo de la simulación como se observa en la **Figura 76**.

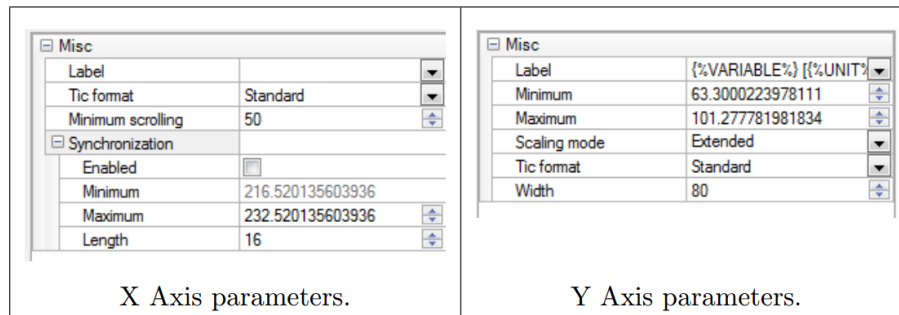


Figura 76. Propiedades de los ejes.

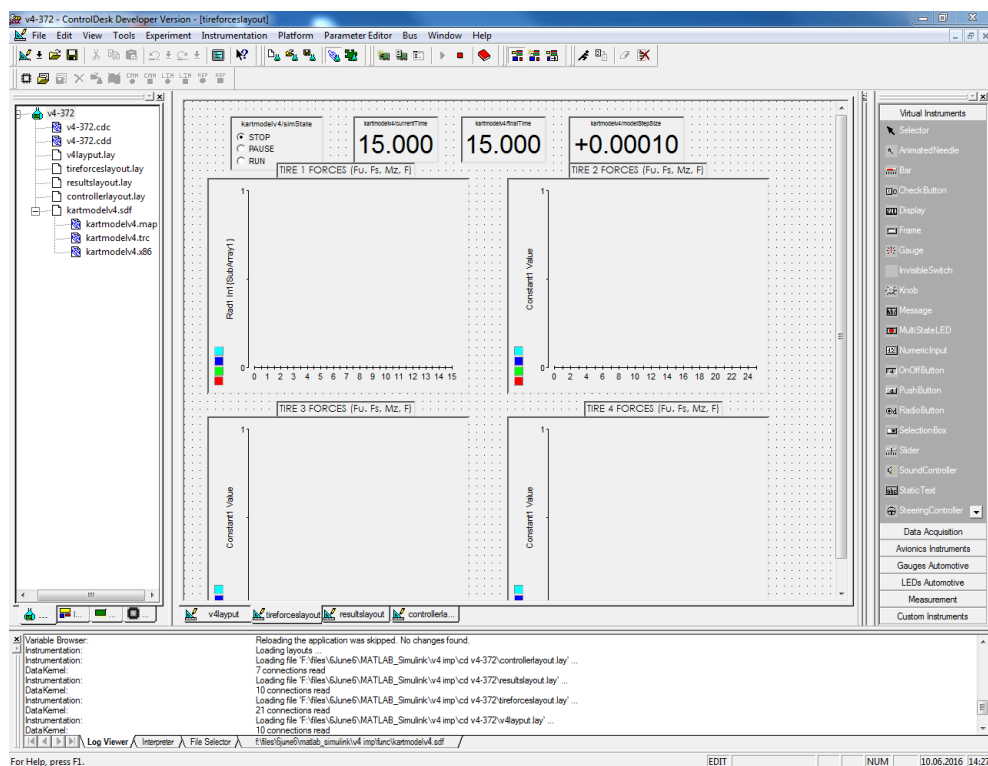


Figura 77. Interfaz finalizada.

5.2.12. Correr la simulación

El software debe estar en modo Animación para correr la simulación.

5.2.13. Capturar datos

Es posible capturar datos del experimento y guardarlos en un archivo de formato conocido (véase **Figura 78**).

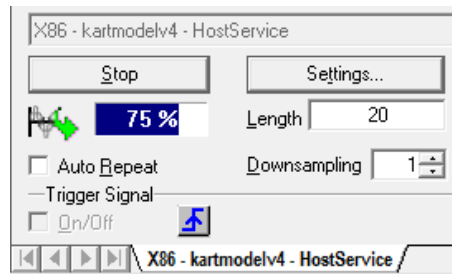


Figura 78. Captura de datos.

La interfaz gráfica del modelo final puede ser observada en la **Figura 79**, en esta interfaz es posible cambiar los parámetros de simulación igual a la interfaz de Simulink.

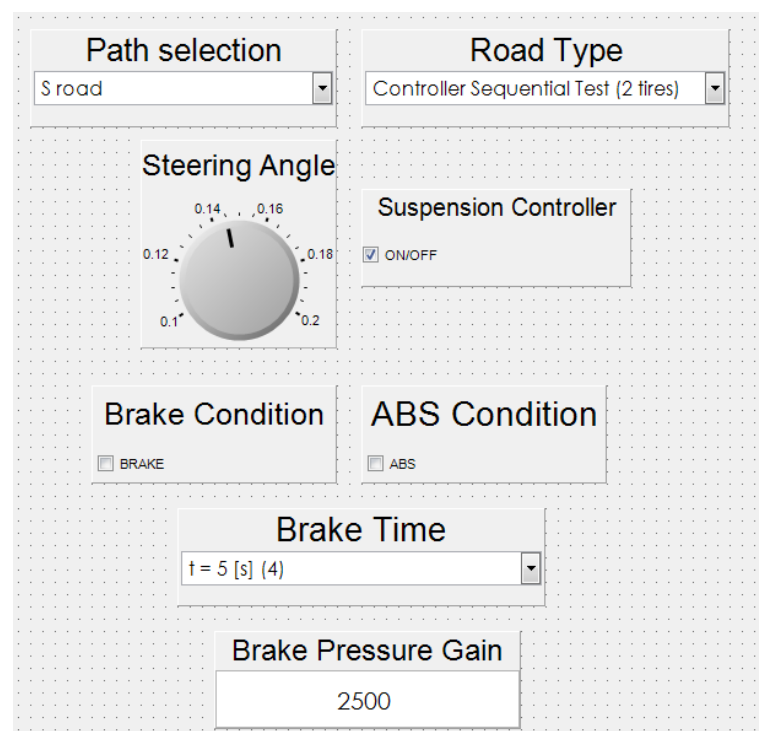


Figura 79. Interfaz DSPACE FINAL.

CAPÍTULO 6

6. PRUEBAS Y ANÁLISIS DE RESULTADOS

6.1. MODELO COMPLEJO

La distribución de cada neumático del vehículo se presenta en la **Figura 80**, con el objetivo de entender de mejor manera el análisis de las simulaciones.

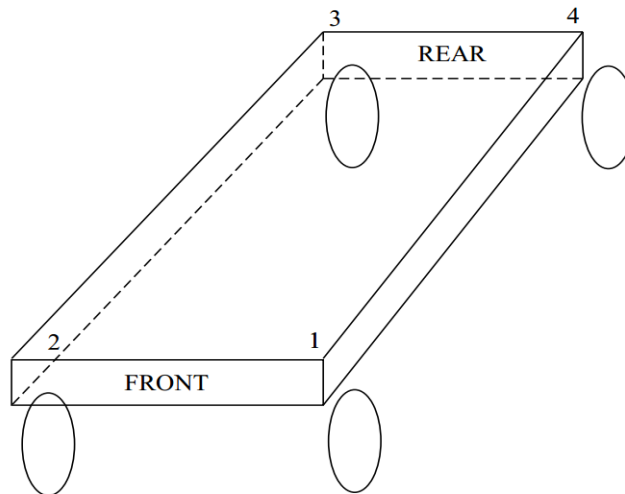


Figura 80. Distribución de las llantas.

6.1.1. Simulaciones y pruebas

Los parámetros de la **Tabla 7** fueron utilizados en las simulaciones de prueba del modelo complejo, estas pruebas pueden consistir en: el seguimiento de una trayectoria predefinida que puede tener perturbaciones o no, una condición de frenado en un tiempo definido y el uso del sistema antibloqueo de frenos. La **Tabla 8** contiene las especificaciones y condiciones para cada prueba.

Tabla 7.

Parámetros de simulación

Opción	Parámetro	Valor
	Tiempo inicial	0
	Tiempo final	15
Solver	Tipo de simulación	Intervalo de tiempo fijo
	Tiempo de muestreo	0.001
	Tipo de solver	ode1
Optimización	Reducción de bloques	Deactivada

Tabla 8.

Pruebas del freno y del ABS

Condiciones	Test [1]	Test [2]	Test [3]	Test [4]	Test [5]	Test [6]
<i>Trayectoria</i>	2	1	1	3	1	8
<i>Perturbación</i>	4	4	4	4	3	2
<i>Ángulo de dirección</i>			0.145 [rad]			
<i>Freno</i>	X	✓	✓	✓	X	X
<i>ABS</i>	X	X	✓	✓	X	X
<i>t Inicio de freno</i>			5 [s]			
<i>Ganancia del freno</i>			2500			

6.1.2. Test 1

Esta prueba consiste en la simulación del vehículo moviéndose a lo largo de la trayectoria sin perturbaciones de la **Figura 81**. Este camino consiste en una vía recta durante los primeros 2 segundos, luego el vehículo rotará 0.7 radianes de manera anti horaria durante 5 segundos y finalmente seguirá una vía recta hasta el final de la simulación.

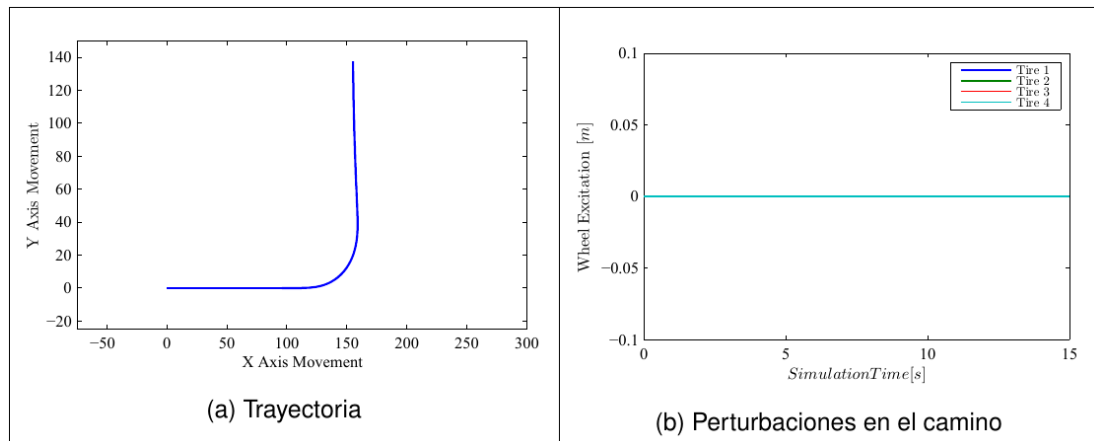


Figura 81. Test [1] Trayectoria de la simulación.

6.1.2.1. Análisis movimiento - Test 1

Al iniciar la simulación, el vehículo se desplazará con una velocidad inicial de 20 m/s (dato inicial) (véase **Figura 82** (a)). Luego, la curva causará que el vehículo disminuya de velocidad hasta que se encuentre con la vía recta, que es cuando empieza a recuperar la velocidad inicial, como se observa en el tramo 5 – 15 [s] de la **Figura 82** (b).

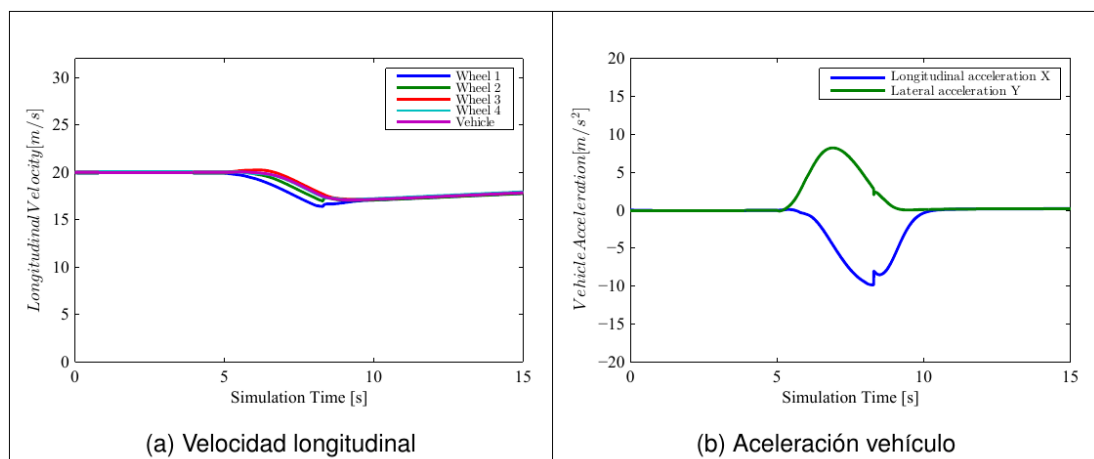


Figura 82. Test [1] Velocidad y aceleración del vehículo.

La **Figura 83** muestra las fuerzas presentes en la llanta frontal y delantera 1 y 3 respectivamente (Fuerza de tracción F_u - azul, Fuerza lateral F_s - verde, Torque de la llanta con respecto al eje Z M_z - rojo, Fuerza normal - celeste).

La fuerza lateral F_s tiene una gran variación en el intervalo 5 – 10 s debido a la curva. La fuerza longitudinal F_u permanece relativamente constante a lo largo del trayecto.

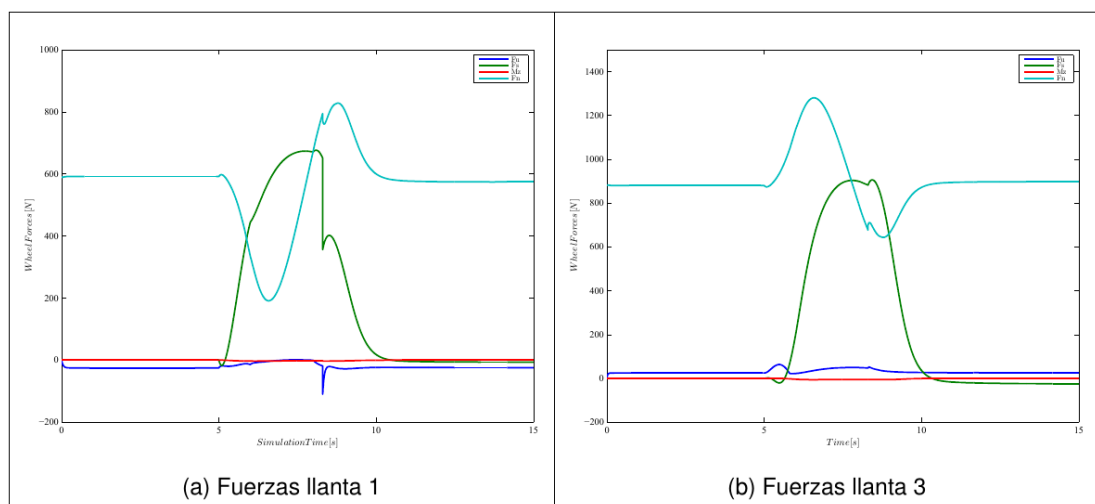


Figura 83. Test [1] Fuerzas en las llantas 1 y 3.

La **Figura 84** muestra la variación angular con respecto a los ejes X, Y, Z que el vehículo realizó durante la simulación. En este caso solo hubo una variación del yaw de aproximadamente 0.7 radianes o 90 grados coincidiendo con la trayectoria de la **Figura 81** (a).

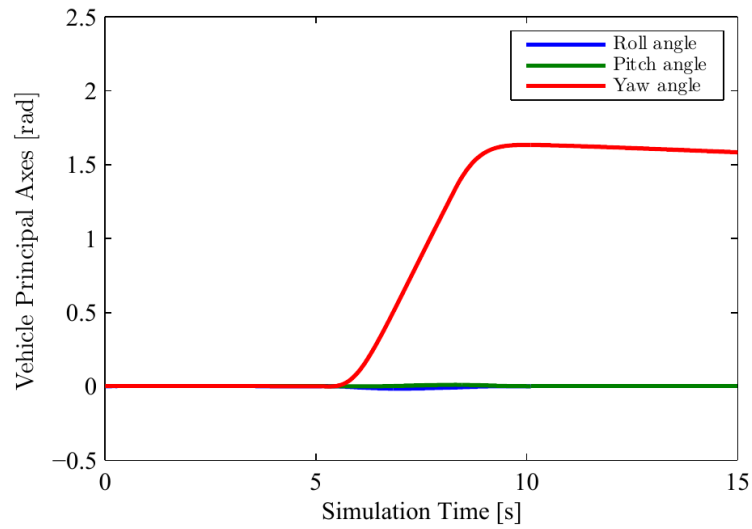


Figura 84. Test [1] Variación angular del vehículo.

Debido a que el vehículo está en movimiento el deslizamiento longitudinal de las llantas es nulo, por otra parte, existe una variación del ángulo de deslizamiento lateral negativo en las llantas causado por la curva del intervalo 5 - 10 [s] (véase **Figura 85**).

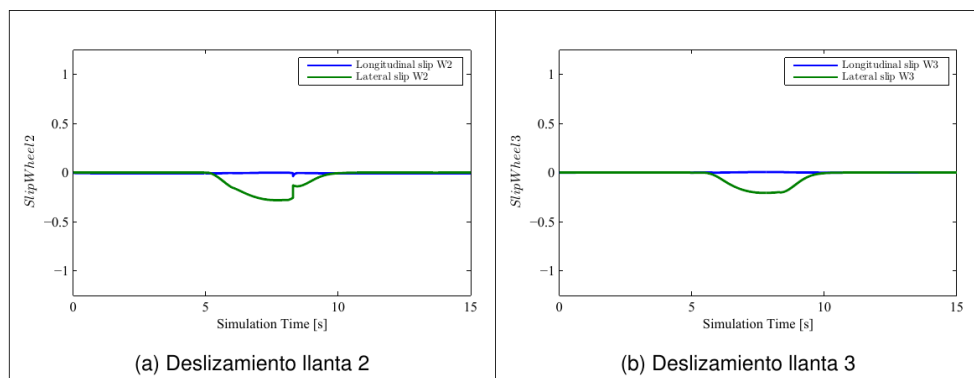


Figura 85. Test [1] Deslizamiento horizontal y lateral.

6.1.2.2. Análisis de suspensión y ABC - Test 1

Existe una variación de longitud en la suspensión debido a la distribución de carga durante la curva (variación del ángulo *roll*), cuando un lado tiene a inclinarse hacia la curva y el otro tiene a alzarse (véase **Figura 86**), es por esto que las llantas laterales derechas 2 y 3 tienen una mayor variación que las llantas laterales izquierdas 1 y 4 (véase **Figura 87 (a)**). Lo mismo se puede decir para las fuerzas ejercidas por el actuador activo que deben compensar la variación de la suspensión mencionada anteriormente (véase **Figura 87 (b)**).



Figura 86. Distribución de carga en el vehículo.

FUENTE: (Driving Fast, s.f.)

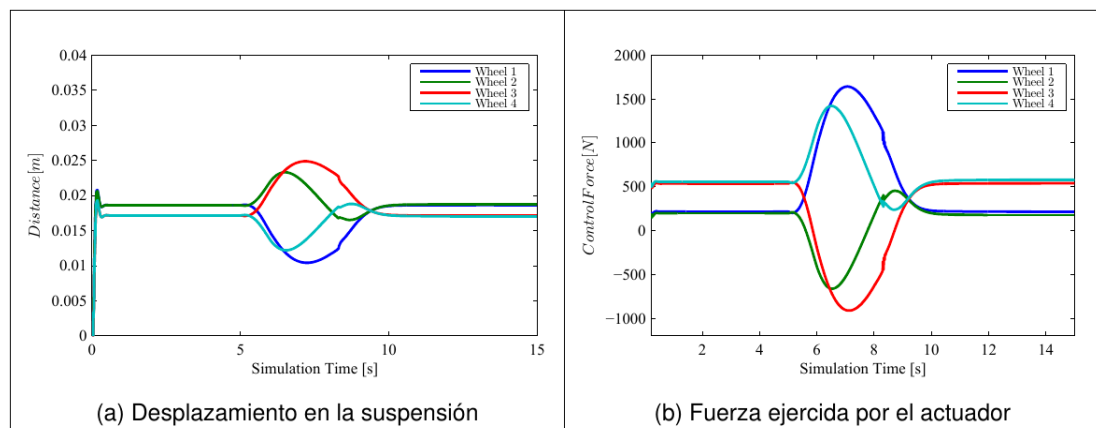


Figura 87. Test [1] Suspensión y ABC.

6.1.3. Test 2

Esta prueba consiste en la simulación del vehículo moviéndose a lo largo de la trayectoria en línea recta de la **Figura 88** (a), en este caso el camino no tiene perturbaciones (véase **Figura 88** (b)). Luego de 5 segundos de iniciada la simulación, se aplicarán los frenos en las llantas del vehículo hasta detenerlo completamente.

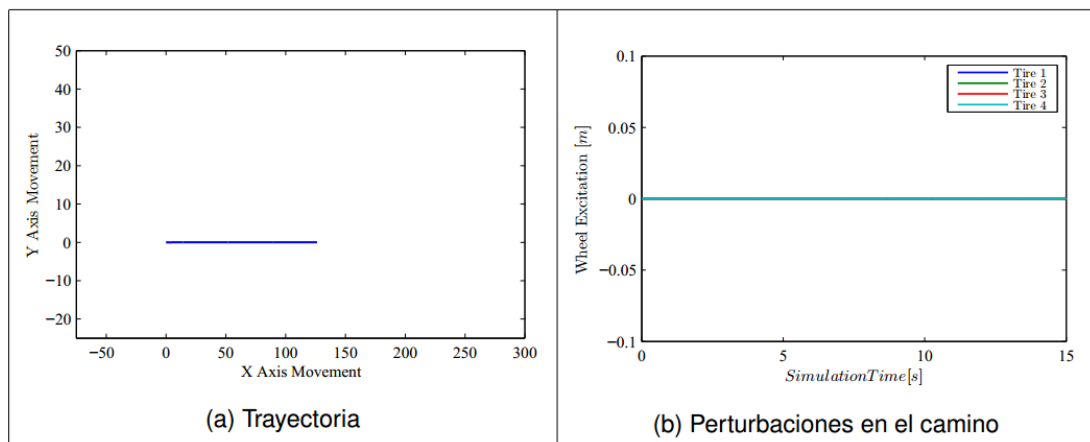


Figura 88. Test [2] Trayectoria de la simulación.

6.1.3.1. Análisis movimiento - Test 2

La velocidad tangencial de cada llanta y la velocidad longitudinal del vehículo es la misma en el tramo de 0 a 5 [s] debido a que el vehículo mantiene una velocidad constante, el vehículo empieza a frenar en el instante de tiempo $t = 5$ [s] y se detiene completamente en $t = 7$ [s].

En la **Figura 89** (a) se puede apreciar que la velocidad tangencial de las llantas varía con la velocidad del vehículo (magenta), este comportamiento muestra que las llantas dejan de girar antes de que se detenga el vehículo completamente, indicando de las llantas se han bloqueado.

La **Figura 89** (b) muestra la desaceleración longitudinal del vehículo desde el momento que se presiona el freno hasta su completa detención.

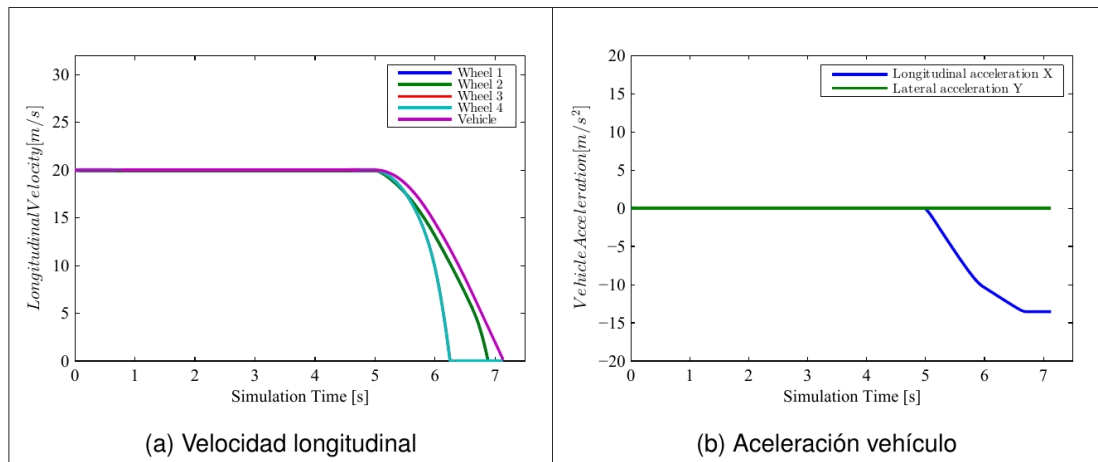


Figura 89. Test [2] Velocidad y aceleración del vehículo.

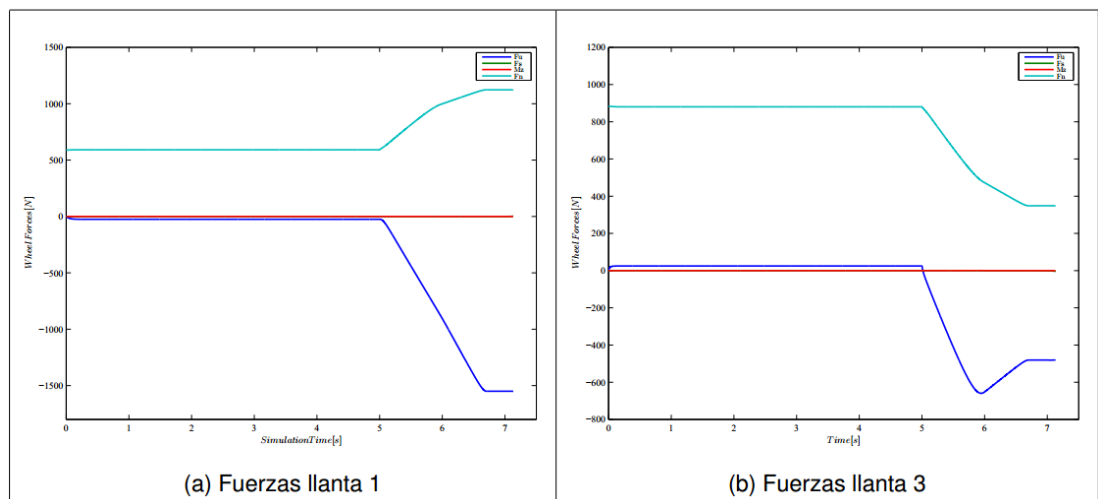


Figura 90. Test [2] Fuerzas en las llantas 1 y 3.

Existe una pequeña variación del *pitch* como se observa en la **Figura 91** debido a la distribución de peso de atrás hacia adelante causada por el frenado.

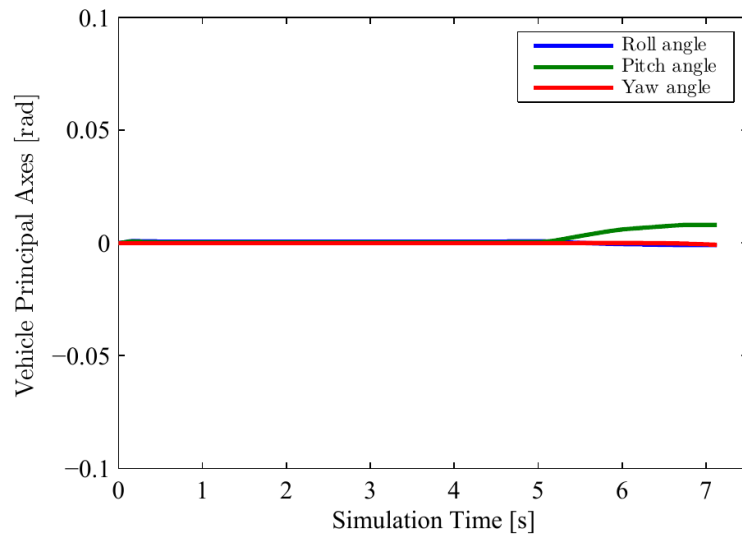


Figura 91. Test [2] Variación angular del vehículo.

6.1.3.2. Análisis de freno y ABS - Test 2

Como se mencionó anteriormente las llantas se bloquearon durante el frenado, esto se puede comprobar analizando la **Figura 92** donde se observa que el deslizamiento horizontal de la llanta frontal 2 y de la llanta posterior 3 disminuye hasta llegar a -1, un indicativo de bloqueo de llantas coincidiendo con el análisis de la sección [Análisis movimiento - Test 2](#).

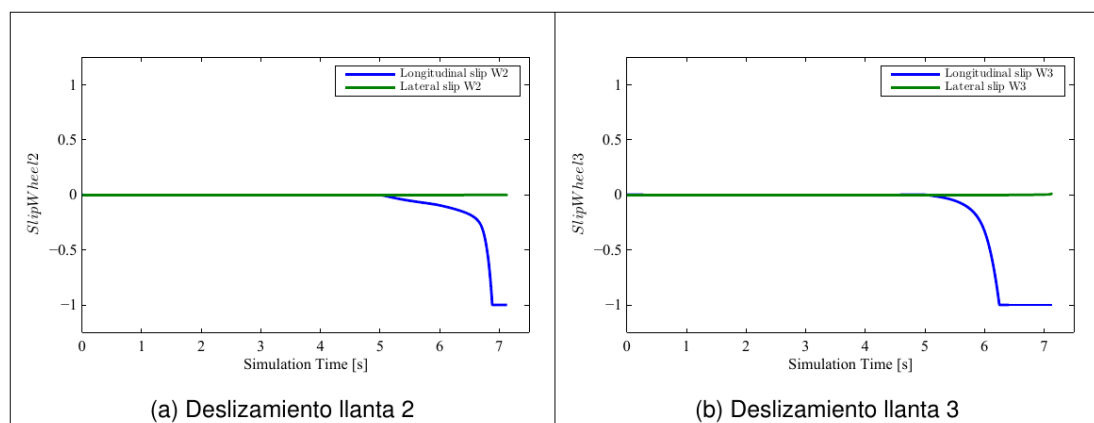


Figura 92. Test [2] Deslizamiento horizontal y lateral.

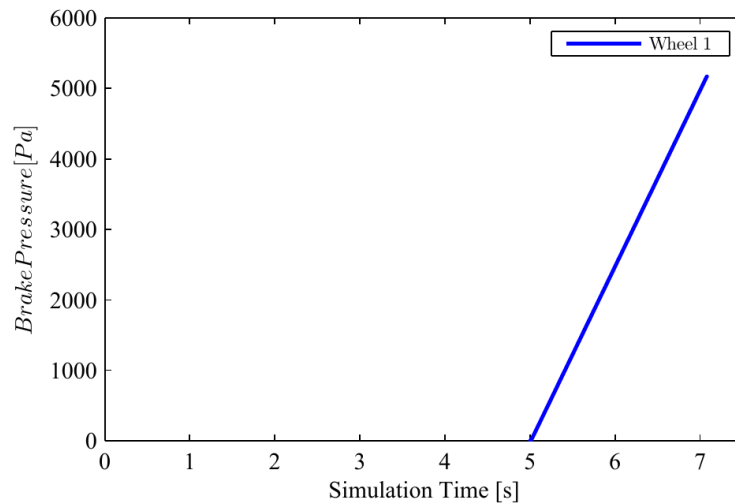


Figura 93. Test [2] Presión de frenado.

6.1.3.3. Análisis de suspensión y ABC - Test 2

Existe una variación de longitud en la suspensión debido a la distribución de carga durante el frenado (variación del ángulo *pitch*), cuando el lado posterior tiende a alzarse y el lado frontal a inclinarse (véase **Figura 95**), es por esto que las llantas frontales 1 y 2 tienen un comportamiento contrario al de las llantas posteriores 3 y 4 (véase **Figura 94 (a)**). Lo mismo se puede decir para las fuerzas ejercidas por el actuador activo que deben compensar la variación de la suspensión mencionada anteriormente (véase **Figura 94 (b)**).

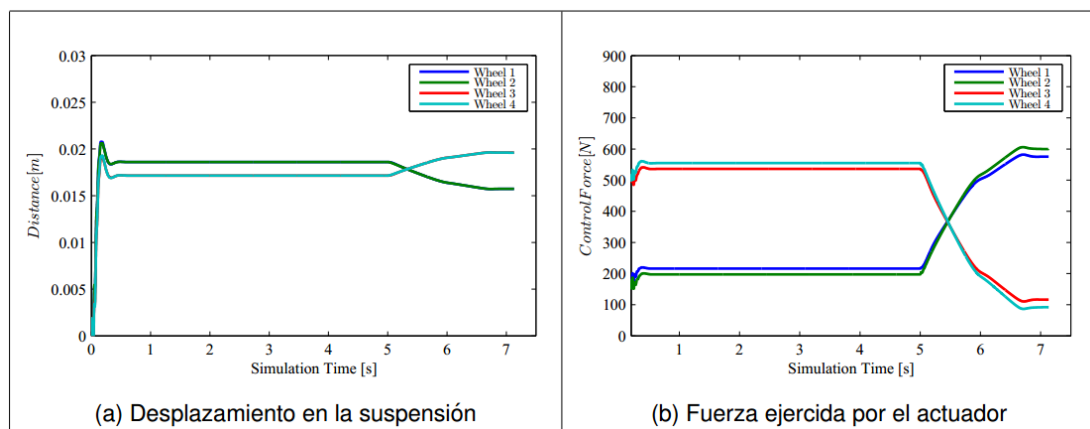


Figura 94. Test [2] Suspensión y ABC.

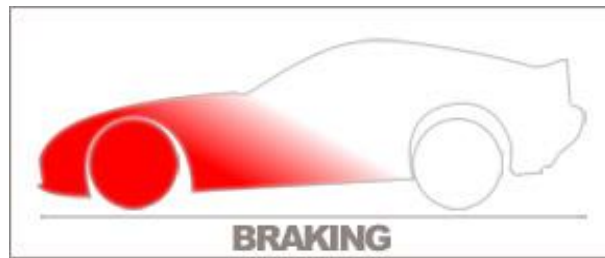


Figura 95. Distribución de carga durante el frenado.

FUENTE: (Gamespy, s.f.)

6.1.4. Test 3

Esta prueba consiste en la simulación del vehículo moviéndose a lo largo de la trayectoria en línea recta sin perturbaciones de la **Figura 96**. Luego de 5 segundos de iniciada la simulación, se aplicarán los frenos en las llantas del vehículo hasta detenerlo completamente y se utilizará el sistema antibloqueo de frenos ABS.

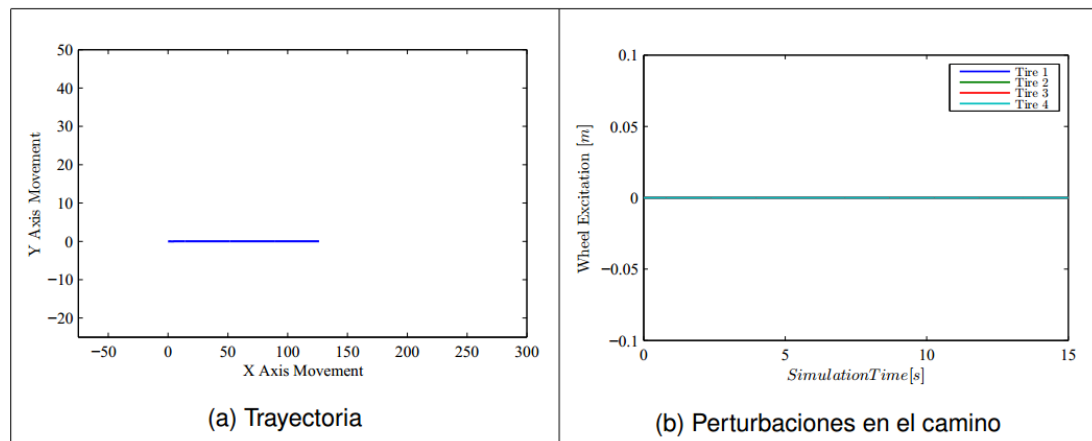


Figura 96. Test [3] Trayectoria de la simulación.

6.1.4.1. Análisis movimiento - Test 3

La velocidad tangencial de cada llanta y la velocidad longitudinal del vehículo es la misma en el tramo de 0 a 5 [s] debido a que el vehículo mantiene una velocidad constante, el vehículo empieza a frenar en el instante de tiempo $t = 5$ [s] y se detiene completamente en $t = 7$ [s].

En la **Figura 97** (a) se puede apreciar que la velocidad tangencial de las llantas tiende a igualar la velocidad longitudinal del vehículo (magenta), este comportamiento muestra el funcionamiento del sistema ABS.

La **Figura 97** (b) muestra la desaceleración longitudinal del vehículo desde el momento que se presiona el freno hasta su completa detención.

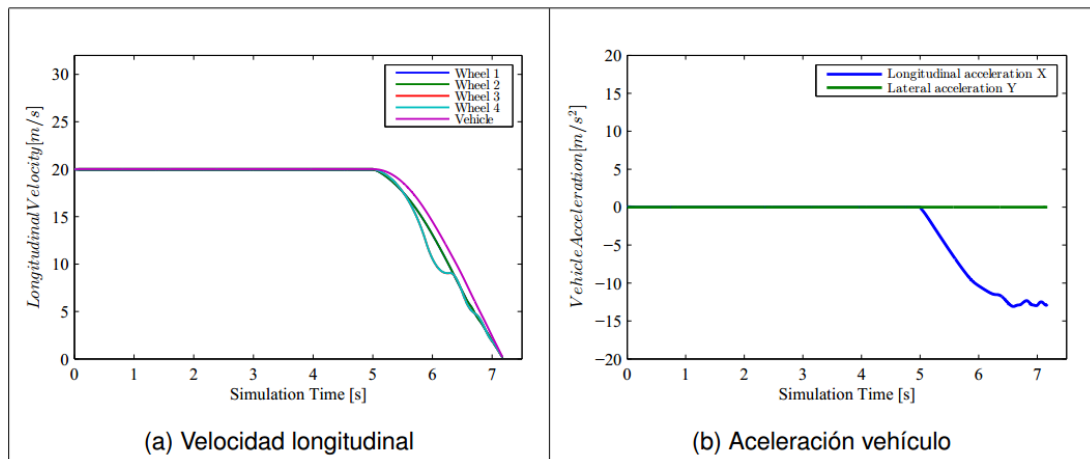


Figura 97. Test [3] Velocidad y aceleración del vehículo.

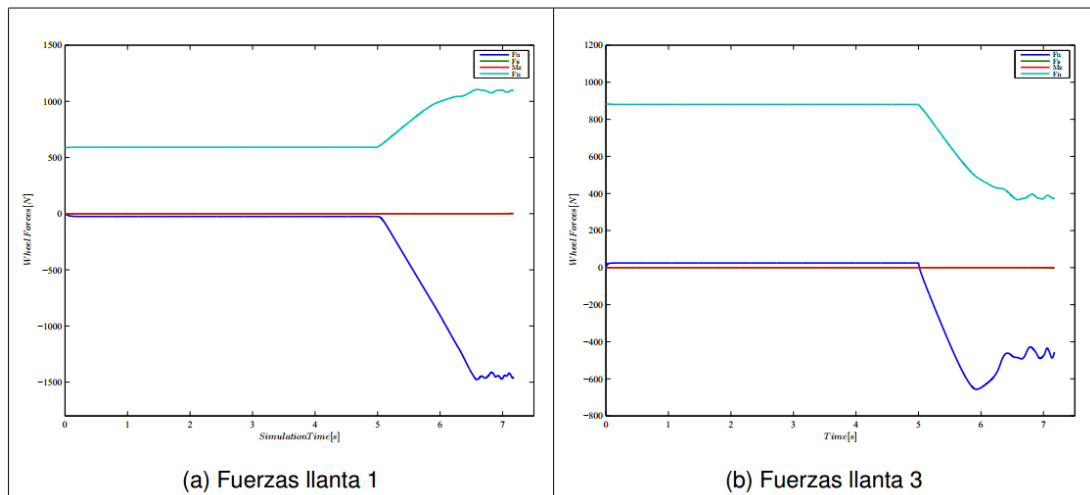


Figura 98. Test [3] Fuerzas en las llantas 1 y 3.

Existe una pequeña variación del *pitch* como se observa en la **Figura 99** debido a la distribución de peso de atrás hacia adelante causada por el frenado.

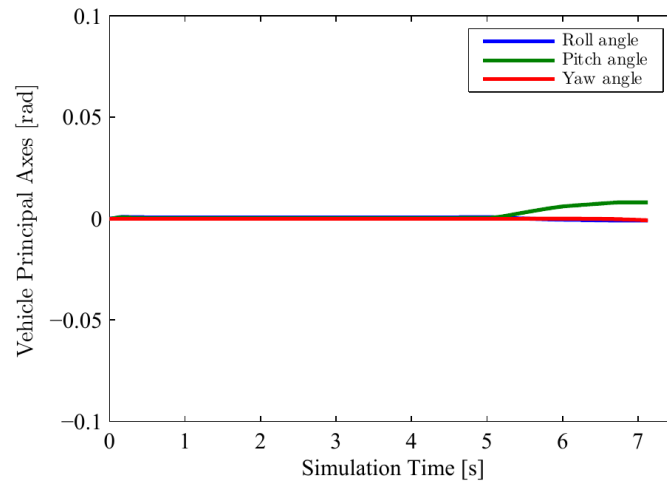


Figura 99. Test [3] Variación angular del vehículo.

6.1.4.2. Análisis de freno y ABS - Test 3

Una vez activado el sistema antibloqueo de frenos, las llantas y el vehículo se detuvieron completamente casi al mismo tiempo indicando que el ABS funcionó previendo el bloqueo de las llantas y asegurando una buena respuesta por parte del vehículo. Una vez que el deslizamiento longitudinal de las llantas haya llegado al valor deseado (0.2) el controlador tomará acción e intentará mantener este valor. Este comportamiento se puede observar en la **Figura 100**.

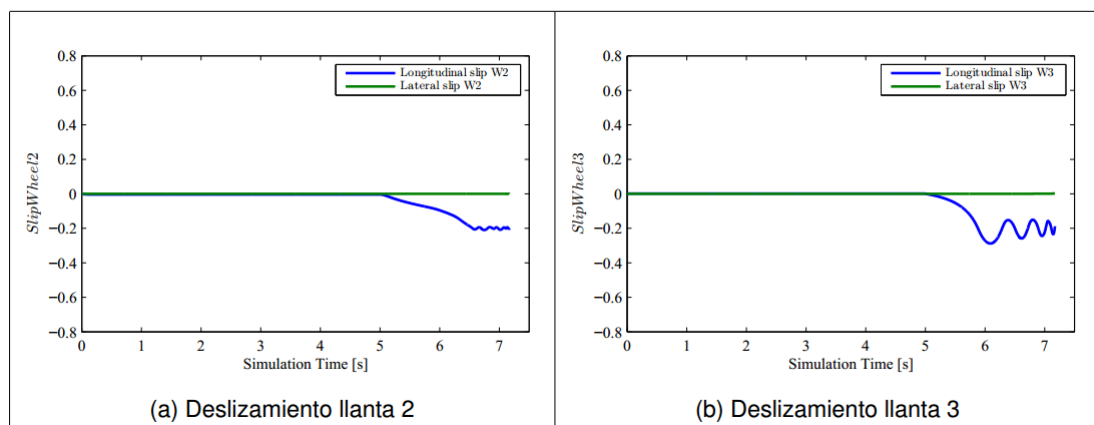


Figura 100. Test [3] Deslizamiento horizontal y lateral.

En la **Figura 101** se puede observar la acción del controlador del ABS en la presión de frenado

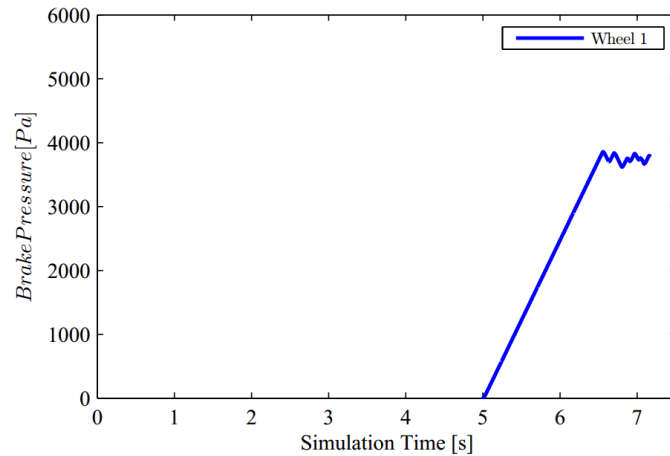


Figura 101. Test [3] Presión de frenado.

6.1.4.3. Análisis de suspensión y ABC - Test 3

Existe una variación de longitud en la suspensión debido a la distribución de carga durante el frenado (variación del ángulo *pitch*), cuando el lado posterior tiende a alzarse y el lado frontal a inclinarse, es por esto que las llantas frontales 1 y 2 es por esto que las llantas frontales 1 y 2 tienen un comportamiento contrario al de las llantas posteriores 3 y 4 (véase **Figura 102** (a)). Lo mismo se puede decir para las fuerzas ejercidas por el actuador activo que deben compensar la variación de la suspensión mencionada anteriormente (véase **Figura 102** (b)).

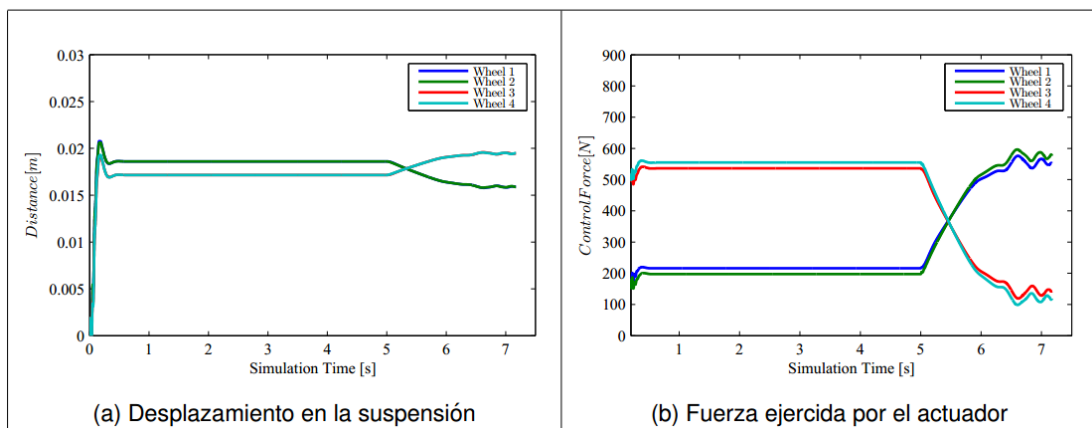


Figura 102. Test [3] Suspensión y ABC.

6.1.5. Test 4

Esta prueba consiste en la simulación del vehículo moviéndose a lo largo de la trayectoria sin perturbaciones de la **Figura 103**. Luego de 5 segundos de iniciada la simulación, se aplicarán los frenos en las llantas del vehículo hasta detenerlo completamente y se utilizará el sistema antibloqueo de frenos ABS.

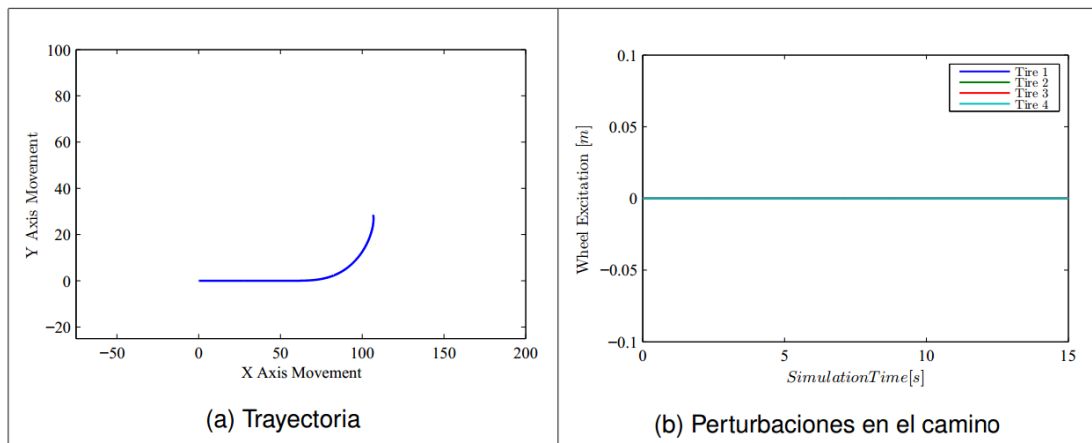


Figura 103. Test [4] Trayectoria de la simulación.

6.1.5.1. Análisis movimiento - Test 4

La velocidad tangencial de cada llanta y la velocidad longitudinal del vehículo es la misma en el tramo de 0 a 5 [s] debido a que el vehículo mantiene una velocidad constante, el vehículo empieza a frenar en el instante de tiempo $t = 5$ [s] y se detiene completamente en $t = 7$ [s].

En la **Figura 104** (a) se puede apreciar que la velocidad tangencial de las llantas tiende a igualar la velocidad longitudinal del vehículo (magenta), este comportamiento muestra el funcionamiento del sistema ABS.

La **Figura 104** (b) muestra la desaceleración longitudinal del vehículo desde el momento que se presiona el freno hasta su completa detención junto con la componente lateral de la aceleración debido a la curva.

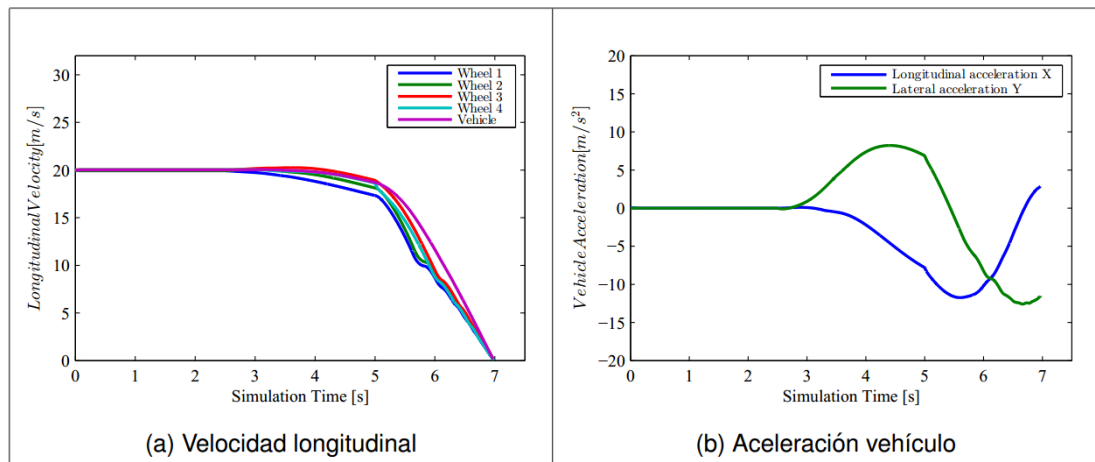


Figura 104. Test [4] Velocidad y aceleración del vehículo.

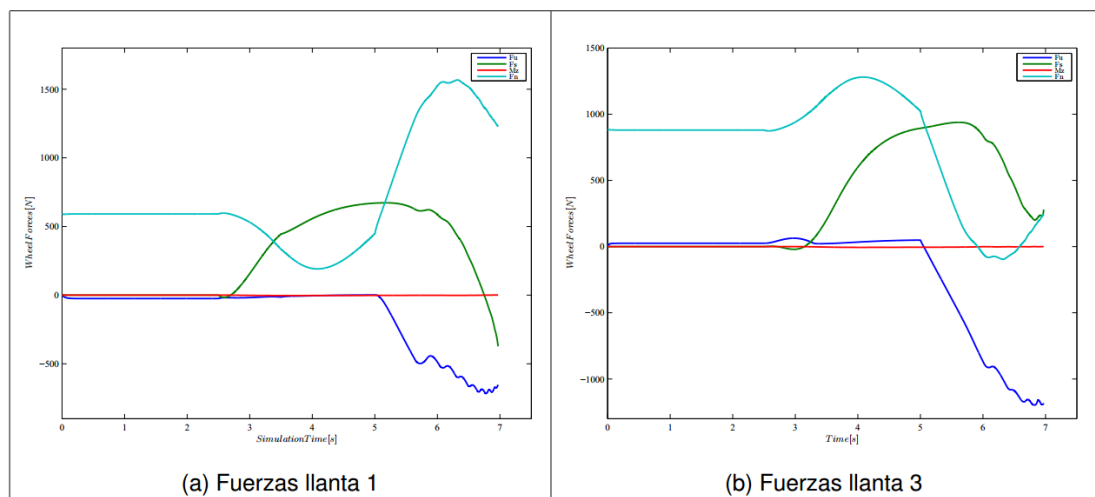


Figura 105. Test [4] Fuerzas en las llantas 1 y 3.

La **Figura 106** muestra la variación angular con respecto a los ejes X, Y, Z que el vehículo realizó durante la simulación. En este caso solo hubo una variación del *yaw* de aproximadamente $\frac{\pi}{2}$ radianes o 90 grados coincidiendo con la trayectoria de la **Figura 103** (a). Además, existe una pequeña variación del *pitch* debido a la distribución de peso de atrás hacia adelante causada por el frenado.

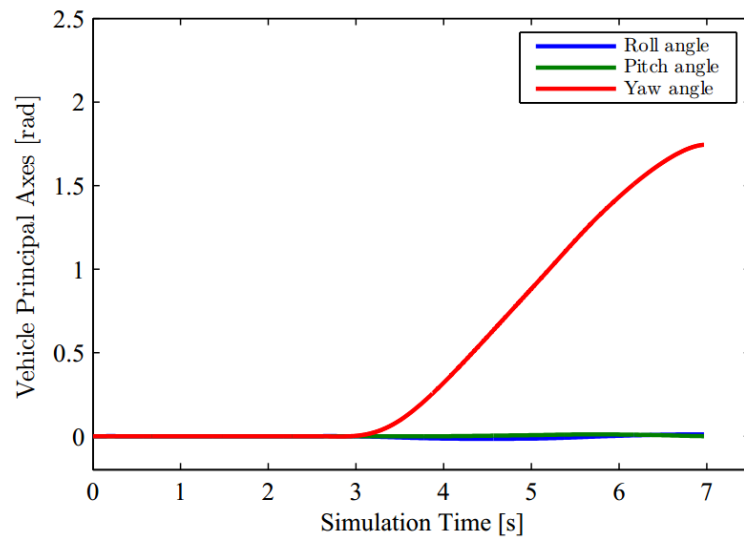


Figura 106. Test [4] Variación angular del vehículo.

6.1.5.2. Análisis de freno y ABS - Test 4

Una vez activado el sistema antibloqueo de frenos, las llantas y el vehículo se detuvieron completamente casi al mismo tiempo indicando que el ABS funcionó previendo el bloqueo de las llantas. Una vez que el deslizamiento longitudinal de las llantas haya llegado al valor deseado (0.2) el controlador tomará acción e intentará mantener este valor. Este comportamiento se puede observar en la **Figura 107**. También se puede apreciar la variación del desplazamiento lateral causado por la curva.

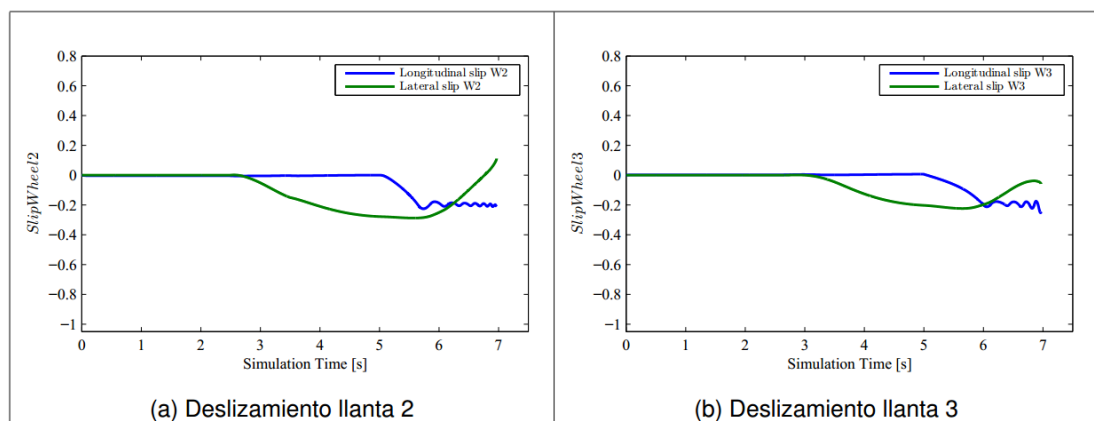


Figura 107. Test [4] Deslizamiento horizontal y lateral.

En la **Figura 108** se puede observar la acción del controlador del ABS en la presión de frenado

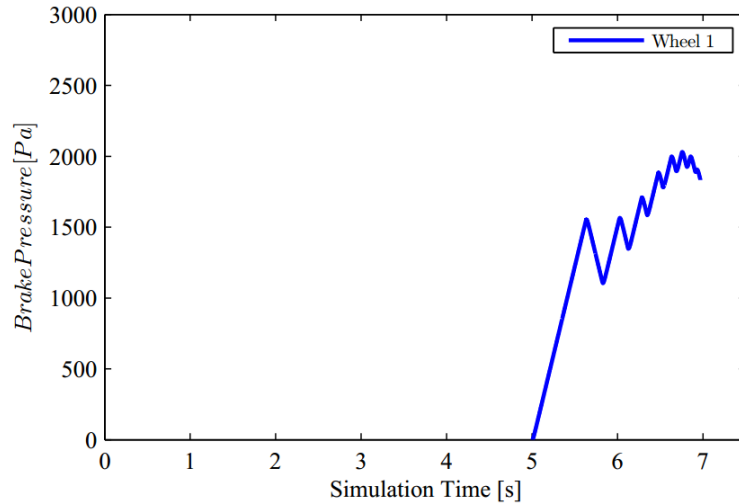


Figura 108. Test [4] Presión de frenado.

6.1.5.3. Análisis de suspensión y ABC - Test 4

Existe una variación de longitud en la suspensión debido a la distribución de carga durante el frenado (variación del ángulo *pitch*) y durante la curva (variación del ángulo *roll*) (véase **Figura 109** (a)). Lo mismo se puede decir para las fuerzas ejercidas por el actuador activo que deben compensar la variación de la suspensión mencionada anteriormente (véase **Figura 109** (b)).

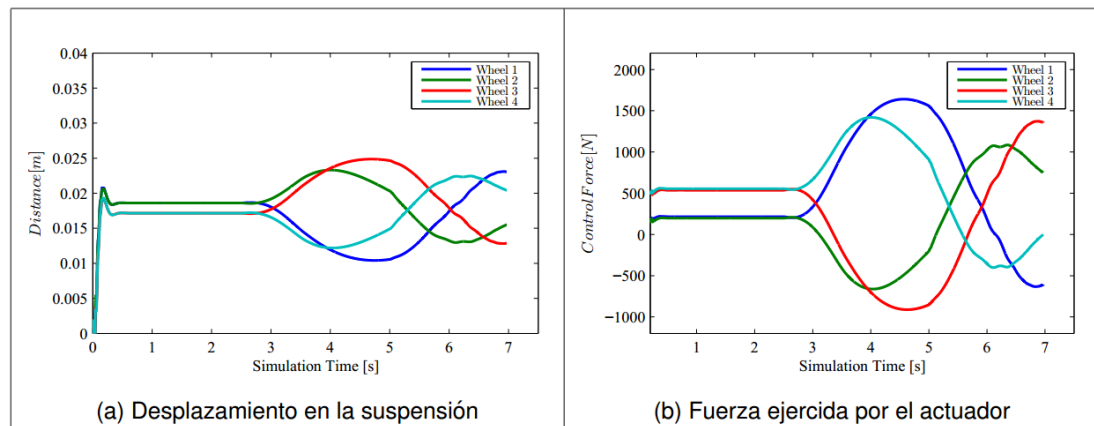


Figura 109. Test [4] Suspensión y ABC.

6.1.6. Test 5

Esta prueba consiste en la simulación de un vehículo moviéndose a lo largo de la trayectoria en línea recta de la **Figura 110** (a), en este caso el camino tiene perturbaciones (véase **Figura 110** (b)).

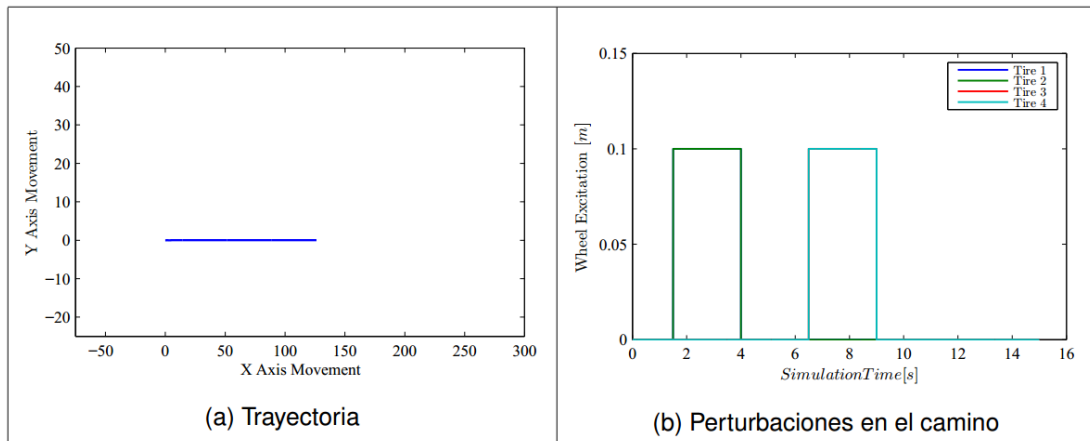


Figura 110. Test [5] Trayectoria de la simulación.

6.1.6.1. Análisis movimiento - Test 5

La velocidad tangencial de cada llanta y la velocidad longitudinal del vehículo se mantienen constante a lo largo del tramo, esto implica que la aceleración tanto longitudinal como lateral sean cero (véase **Figura 111**).

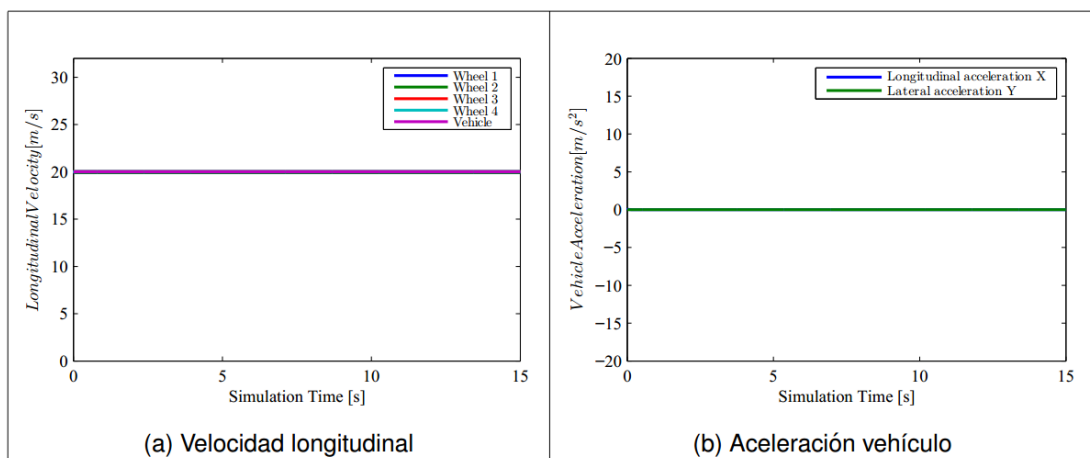


Figura 111. Test [5] Velocidad y aceleración del vehículo.

La **Figura 112** muestra las fuerzas presentes en la llanta frontal y delantera 1 y 3 respectivamente (Fuerza de tracción - azul, Fuerza lateral - verde, Torque de la llanta con respecto al eje Z - rojo, Fuerza normal - celeste). Es evidente la influencia de las perturbaciones en la fuerza normal de las llantas.

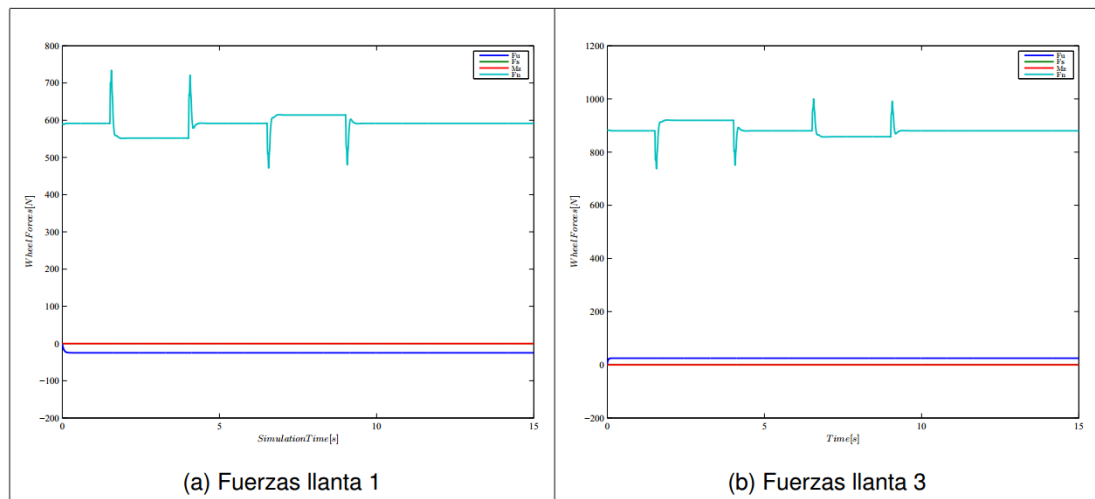


Figura 112. Test [5] Fuerzas en las llantas 1 y 3.

La **Figura 113** muestra la variación angular con respecto a los ejes X, Y, Z que el vehículo realizó durante la simulación. Existe una pequeña variación del *pitch* y del *roll* causadas por las perturbaciones en el camino.

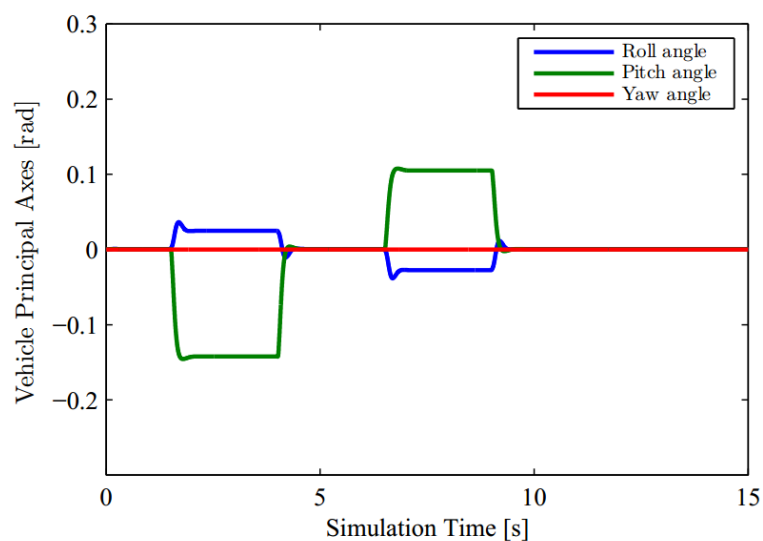


Figura 113. Test [5] Variación angular del vehículo.

Debido a que el vehículo está en movimiento en una trayectoria en línea recta el deslizamiento longitudinal y el deslizamiento lateral de las llantas es nulo (véase **Figura 114**).

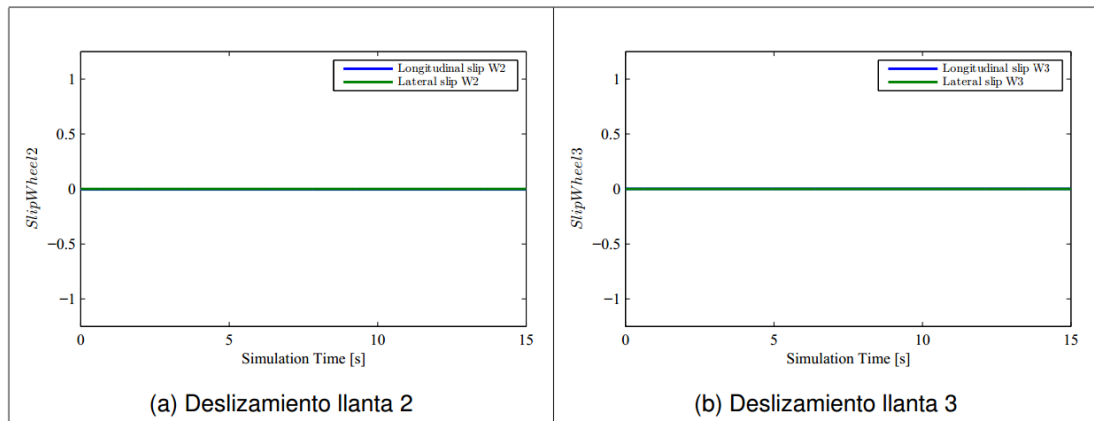


Figura 114. Test [5] Deslizamiento horizontal y lateral.

6.1.6.2. Análisis de suspensión y ABC - Test 5

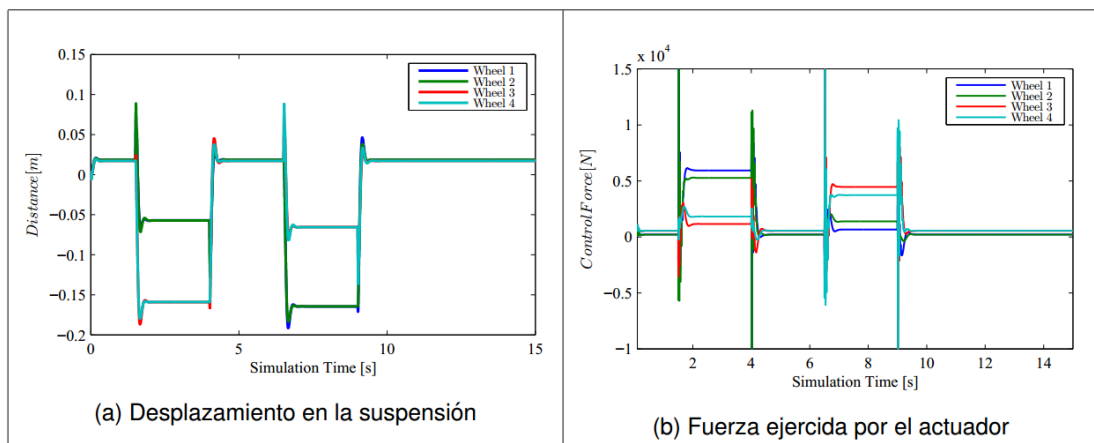


Figura 115. Test [5] Suspensión y ABC.

6.1.7. Test 6

Esta prueba consiste en la simulación de un vehículo moviéndose a lo largo de la trayectoria de la **Figura 116 (a)**, en este caso el camino tiene perturbaciones (véase **Figura 116 (b)**).

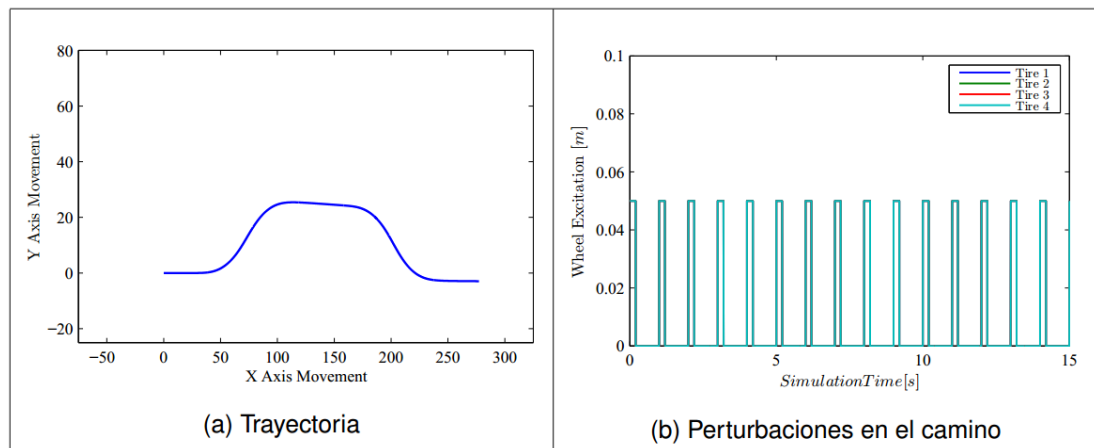


Figura 116. Test [6] Trayectoria de la simulación.

6.1.7.1. Análisis movimiento - Test 6

La velocidad tangencial de cada llanta varía ligeramente en los tramos donde hay curvas, esto implica que exista una aceleración longitudinal y lateral (véase **Figura 117**).

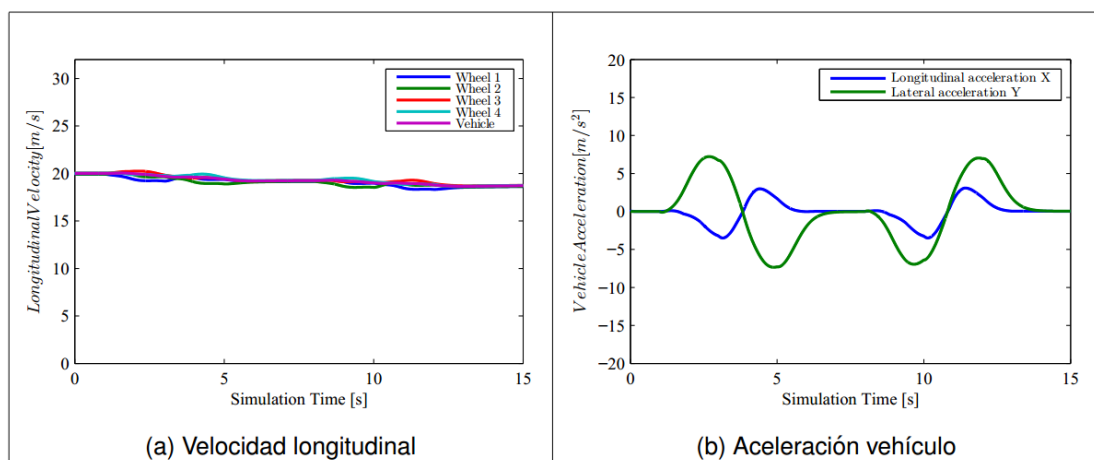


Figura 117. Test [6] Velocidad y aceleración del vehículo.

La **Figura 118** muestra las fuerzas presentes en la llanta frontal y delantera 1 y 3 respectivamente (Fuerza de tracción - azul, Fuerza lateral - verde, Torque de la llanta con respecto al eje Z - rojo, Fuerza normal - celeste). Es evidente la influencia de las perturbaciones en la fuerza normal de las llantas.

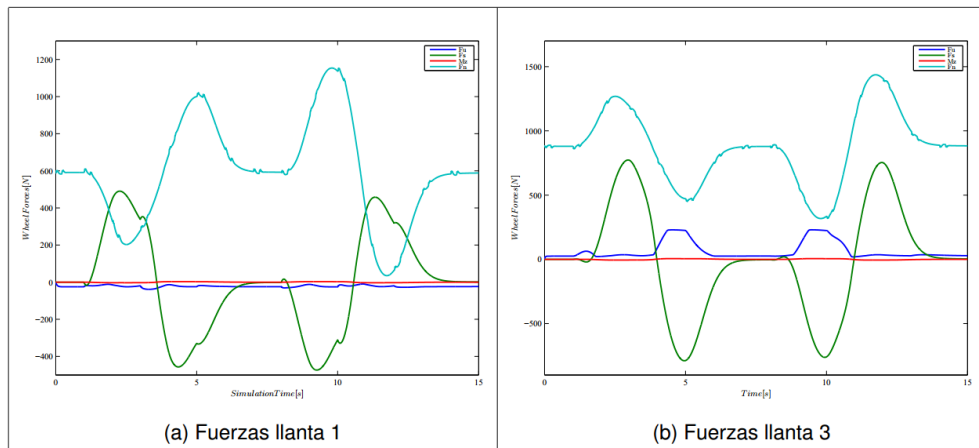


Figura 118. Test [6] Fuerzas en las llantas 1 y 3.

La **Figura 119** muestra la variación angular con respecto a los ejes X, Y, Z que el vehículo realizó durante la simulación. En este caso solo hubo una variación del *yaw* en las curvas de aproximadamente 0.7 radianes o 40 grados coincidiendo con la trayectoria de la **Figura 81** (a). Además, existe una pequeña variación del *pitch* y del *roll* causadas por las perturbaciones en el camino.

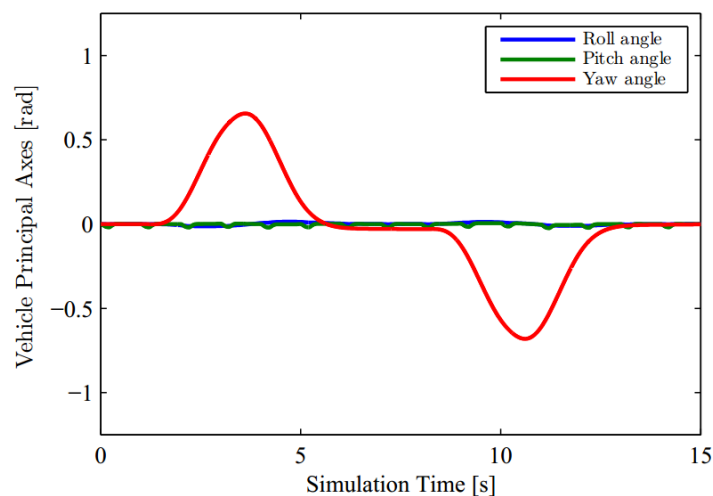


Figura 119. Test [6] Variación angular del vehículo.

Debido a que el vehículo está en movimiento el deslizamiento longitudinal de las llantas es nulo, por otra parte, existe una variación de ángulo de deslizamiento lateral causado por las curvas de la trayectoria (véase **Figura 120**).

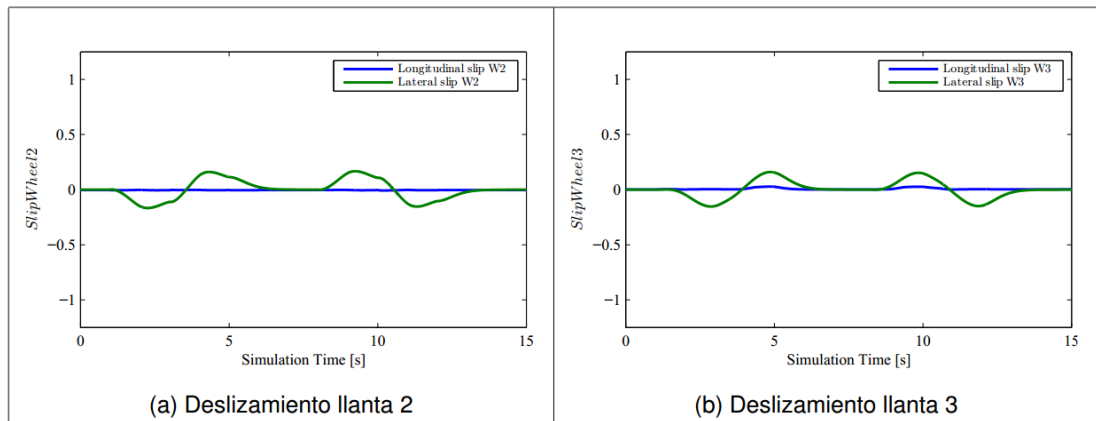


Figura 120. Test [6] Deslizamiento horizontal y lateral.

6.1.7.2. Análisis de suspensión y ABC - Test 6

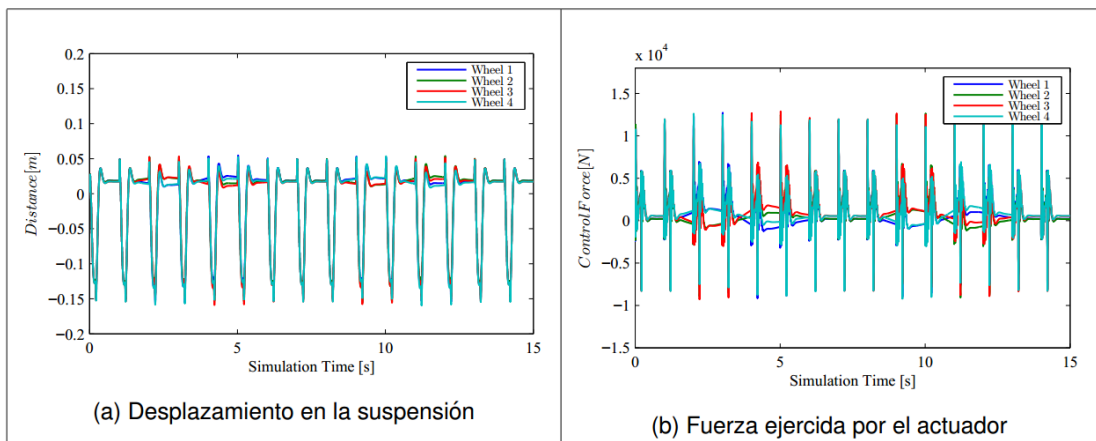


Figura 121. Test [6] Suspensión y ABC.

6.1.8. Medición de los parámetros de rendimiento en Simulink

Es posible medir el tiempo total de la simulación en Simulink usando el comando `tic,[t,x,y]=sim('KartmodelV2');toc` en MATLAB.

El tiempo de la simulación del modelo v2 (véase **Tabla 9**) es mayor al tiempo de simulación del primer modelo en MATLAB. Es por eso que se implementó como solución final la versión 3 utilizando la función-S en C.

Tabla 9.

Grados de libertad de algunos modelos de vehículo

Tiempo de muestreo [s]	Tiempo de Simulación [s]	
	v1	v2
0.01	3.182	30.723
0.001	31.243	324.714

v1 *MATLAB function block*

v2 *Lvl-2 MATLAB S-function block*

Luego de esto se comparó el rendimiento de este modelo con las otras versiones con la conclusión de que este modelo es el más rápido como se ve en la **Tabla 10**.

Tabla 10.

Grados de libertad de algunos modelos de vehículo

Tiempo de muestreo [s]	Tiempo de Simulación [s]		
	v1	v2	v3
0.01	3.182	30.723	1.924
0.001	31.243	324.714	17.217

v1 *MATLAB function block*

v2 *Lvl-2 MATLAB S-function block*

v3 *C S-function block*

6.2. DSPACE

Como se mencionó anteriormente, el software Control Desk es una plataforma que permite enlazar el modelo realizado en Simulink con el equipo DSPACE mediante una interfaz gráfica, por lo que los datos y gráficas simuladas serán los mismos que los obtenidos en Simulink.

La **Figura 122** muestra los resultados obtenidos en el software **DSPACE Control Desk 3.72**, en el cual se testeó el modelo del GO-KART.

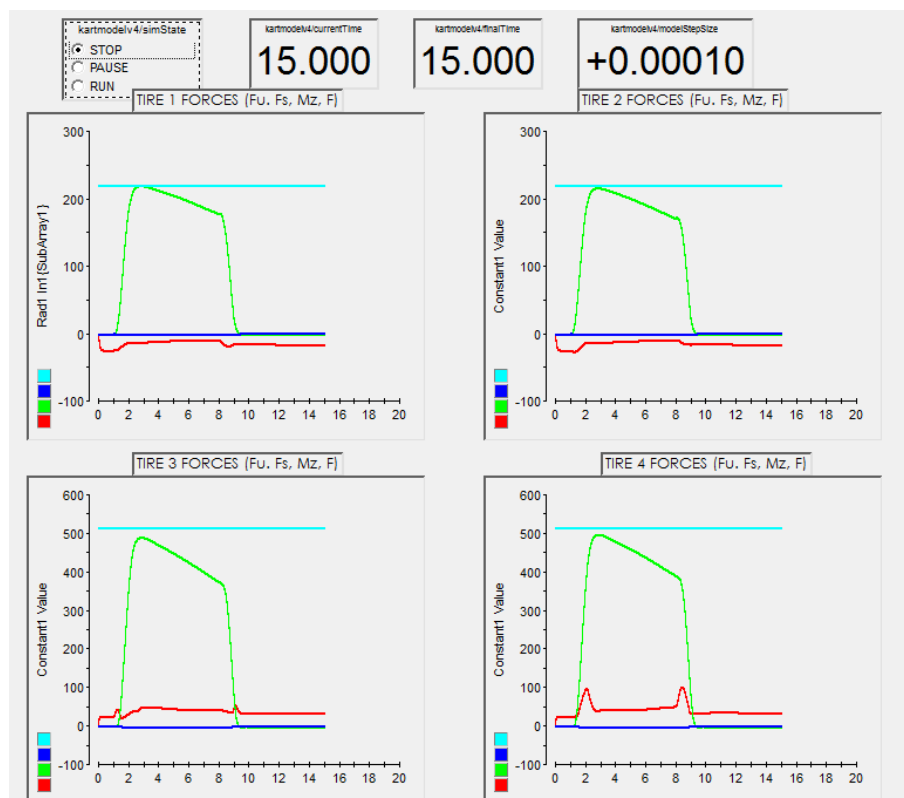


Figura 122. Resultados obtenidos en DSPACE, modelo GO-KART.

Se seguirán los mismos pasos del [capítulo 5](#) para transferir el modelo del vehículo complejo a la plataforma DSPACE. En este caso se utilizará la versión **DSPACE Control Desk 4.2 Next Generation**, la cual es una versión actualizada en comparación con la usada en el anterior modelo. Estos son los resultados obtenidos:

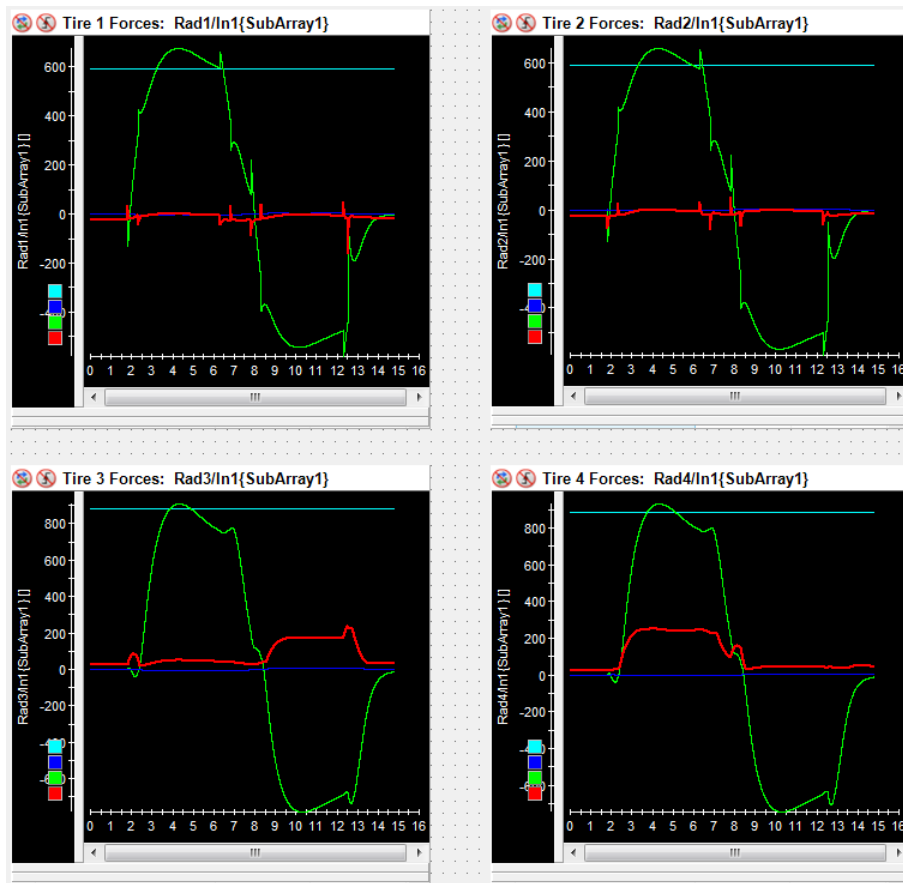


Figura 123. Resultados obtenidos en DSPACE, modelo complejo.

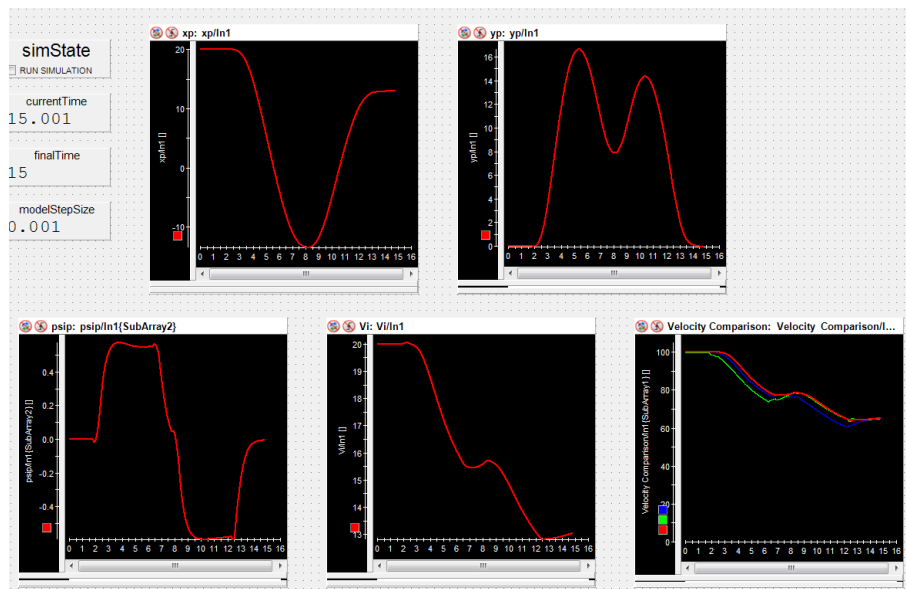


Figura 124. Resultados obtenidos en DSPACE, modelo complejo.

6.3. MEDICIÓN DE LOS PARÁMETROS DE RENDIMIENTO

6.3.1. Simulink

Tabla 11.

Medición del rendimiento del modelo en Simulink

Modelo en Simulink			
<i>Tiempo de muestreo</i>	<i>t1</i>	<i>t2</i>	<i>tPromedio</i>
0.001 [s]	18.72 [s]	17.22 [s]	17.97 [s]
0.0001 [s]	404.82 [s]	358.12 [s]	381.47 [s]

6.3.2. DSPACE

Tabla 12.

Medición del rendimiento del modelo en ControlDesk

Modelo en DSPACE			
<i>Tiempo de muestreo</i>	<i>t1</i>	<i>t2</i>	<i>tPromedio</i>
0.001 [s]	15.214 [s]	15.217 [s]	15.216 [s]
0.0001 [s]	17.68 [s]	17.914[s]	17.797 [s]

6.4. LIMITACIONES DEL MODELO

Una limitación importante del modelo se originó en la configuración del tiempo fijo de simulación. Como se vio en el [capítulo 3](#) el primer modelo uso un tiempo de simulación de **0.01 s**, este mismo tiempo fue utilizado para el modelo complejo, pero surgió un conflicto en la simulación originando datos erróneos y la imposibilidad de continuar.

Esto se debe a que el tiempo 0.01 s no pudo soportar la complejidad del nuevo modelo, por lo que fue necesario cambiar el tiempo fijo a **0.001 s**.

6.5. EXPORTACIÓN DE DATOS Y ANIMACIÓN

Un programa de animación desarrollado en C++ por el profesor Christian Oertel fue utilizado para apreciar de mejor manera la variación de movimiento del vehículo a lo largo de la trayectoria definida. Una función importante de este software es la capacidad de visualizar las fuerzas que actúan en las llantas y en el vehículo al mismo tiempo que la simulación está corriendo, esto permite al usuario entender de mejor manera los gráficos y análisis realizados anteriormente.

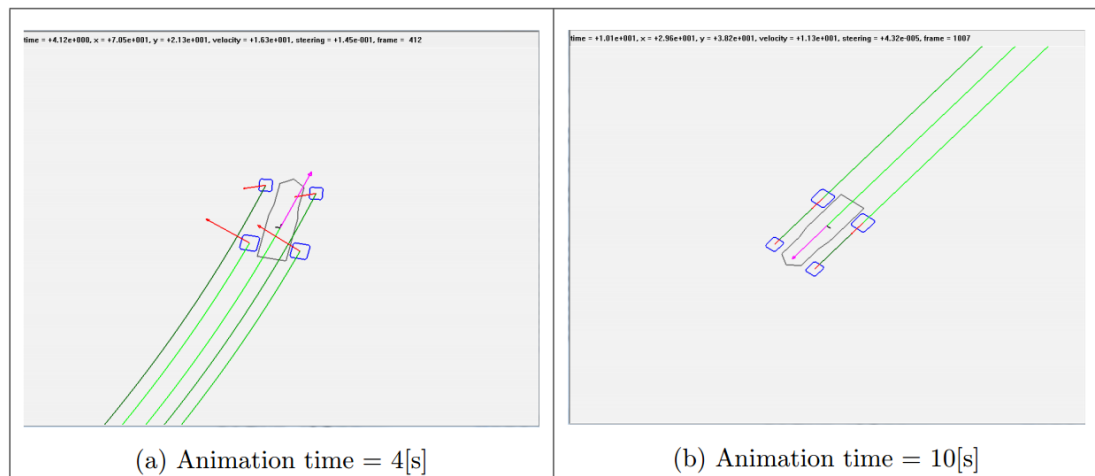


Figura 125. Visualización de la trayectoria del vehículo.

CAPÍTULO 7

7. CONCLUSIONES Y TRABAJO FUTURO

7.1. CONCLUSIONES

- Un modelo de vehículo de 14 grados de libertad fue desarrollado, analizado, probado, usando el software MATLAB/Simulink, las simulaciones hardware y software in-the-loop fueron implementadas en el equipo DSPACE 1006 usando como software ControlDesk.
- El modelo de vehículo consta de subsistemas que se encuentran presentes en vehículos modernos como el sistema antibloqueo de frenos y el control activo del chasis, además se realizó el modelamiento de: un sistema de dirección con su respectivo controlador, la dinámica del chasis y de los neumáticos, un sistema de frenos y un sistema de suspensión.
- El modelo de vehículo fue testeado en condiciones representativas de manejo, calidad del camino y en diferentes trayectorias logrando un comportamiento adecuado. En adición, todos los problemas de conectividad, software y simulación fueron solucionados.
- La implementación del sistema antibloqueo de frenos y del control activo de chasis en el modelo del vehículo mostró mejoras significativas durante el manejo del vehículo en condiciones difíciles.
- Este trabajo representa una base sólida para futuros desarrollos y mejoras.

7.2. TRABAJO FUTURO

A continuación, se detallan las recomendaciones dadas por el co-tutor Prof. -Ing. Christian Oertel durante la defensa en la Universidad de Ciencias Aplicadas de Brandeburgo.

Mejorar la precisión del vehículo, usando los siguientes conceptos para obtener una mejor aproximación a la realidad: fuerza aerodinámica, resistencia a la rodadura, datos específicos de distancias y pesos dependiendo de la aplicación, uso de elementos complementarios al chasis del vehículo como la barra estabilizadora.

Implementar un controlador más robusto para la dirección y para el sistema antibloqueo de frenos.

Trabajar con controladores reales utilizando los protocolos CAN o FLEXRAY.

Utilizar patrones de testeo estandarizados como la [ISO 3888-1:1999](#) o la [ISO 8093:1985](#) para comparar resultados de las pruebas.

REFERENCIAS BIBLIOGRÁFICAS

- Aficionados a la mecánica. (2014). *Sistema de Dirección*. Obtenido de Aficionados a la mecánica: <http://www.aficionadosalamecanica.net/direccion-asistida-electr.htm>
- Anyu, C., Xiliang, D., & Shugen, H. (Enero de 2011). Co-simulation Study on EPS System Based on ADAMS and MATLAB. *2011 Third International Conference on Measuring Technology and Mechatronics Automation*. Institute of Electrical & Electronics Engineers (IEEE). doi:10.1109/icmtma.2011.199
- Bardini, R., Schramm, D., & Hiller, M. (2014). *Vehicle Dynamics: Modeling and Simulation* (1 ed.). Springer-Verlag Berlin Heidelberg.
- Campa. (2014). What is the best way to implement my algorithm in Simulink. MathWorks, Inc.
- Cao, X., & Cleghorn, W. L. (Agosto de 2010). Examples and application of ADAMS software in the mechanics of machines teaching. *2010 5th International Conference on Computer Science {&} Education*. Institute of Electrical & Electronics Engineers (IEEE). doi:10.1109/iccse.2010.5593596
- Dorsch, V. (2014). Simulation of Vehicle Dynamics Control by active Steering Systems. SIMPACK.
- Driving Fast. (s.f.). *Weight Transfers*. Obtenido de Driving Fast: <http://www.drivingfast.net/weight-transfers>
- dSpace. (s.f.). *DS1006 Processor Board*. Obtenido de dSpace: https://www.dspace.com/en/inc/home/products/hw/modular_hardware_introduction/processor_boardsds1006.cfm
- dSPACE GmbH. (2015). *DS1006 Processor Board*.
- dSPACE GmbH. (2016). Hardware-in-the-Loop Testing Systems.

- Fernandez, J. (2012). A Vehicle Dynamics Model for Driving Simulators. *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*. Chalmers University of Technology.
- Formula 1 Dictionary. (s.f.). *Motions of F1 Car*. Obtenido de Formula 1 Dictionary: http://www.formula1-dictionary.net/motions_of_f1_car.html
- Gambarotta, A., Ruggiero, A., Sciolla, M., & Lucchetti, G. (s.f.). Hil/sil system for the simulation of turbocharged diesel engines., (págs. 49-53).
- Gamespy. (s.f.). *Forza Motorsport 3 Driving 101*. Obtenido de Gamespy: http://xbox360.gamespy.com/xbox-360/forza-3/guide/page_10.html
- Ghaffari, A. (2009). *dSPACE and Real-time Interface in Simulink*. San Diego State University.
- Gietelink, O., Ploeg, J., DeSchutter, B., & Verhaegen, M. (2006). Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations. *Vehicle System Dynamics*, 44, 569-590.
- Goldwing Facts. (2 de Enero de 2013). *Design Differences Between Car and Motorcycle Rim/Tire*. Obtenido de Goldwing Facts: <http://www.goldwingfacts.com/forums/10-reference-faq-forum/400426-design-differences-between-car-motorcycle-rim-tire.html>
- Guvenc, B., Guvenc, L., & Karaman, S. (2009). Robust Yaw Stability Controller Design and Hardware-in-the-Loop Testing for a Road Vehicle. *IEEE Transactions on Vehicular Technology*, 58, 555-571.
- HiTech Auto Mechanical. (2016). *Front Rear Brake Replacement ABS Service*. Obtenido de HiTech Auto Mechanical: [www.hitechautomechanical.com.au/Front Rear-Brake-Replacement-ABS-Service](http://www.hitechautomechanical.com.au/Front-Rear-Brake-Replacement-ABS-Service)
- Isermann, R., Schaffnit, J., & Sinsel, S. (1999). Hardware-in-the-loop simulation for the design and testing of engine-control systems. *Control Engineering Practice*, 643-653. doi:[http://dx.doi.org/10.1016/S0967-0661\(98\)00205-6](http://dx.doi.org/10.1016/S0967-0661(98)00205-6)

- Lee, M. H., Lee, H. M., K.S., L. E., S.K., H. A., & J.I, B. A. (2011). Development of a Hardware in the Loop simulation system for electric power steering in vehicles. *International Journal of Automotive Technology*, 12, 733-744.
- MathWorks. (1997). *Real-Time Workshop For Use with SIMULINK*. MathWorks.
- MathWorks. (s.f.). *Modeling and antilock braking system*. Obtenido de MathWorks:
http://www.mathworks.com/examples/simulink/mw/simulink_product-sldemo_absbrake-modeling-an-anti-lock-braking-system
- Mathworks. (s.f.). *Products: Simulink*. Obtenido de Mathworks:
<http://www.mathworks.com/products/simulinkk>
- Mathworks. (s.f.). *What is an S-function*. Obtenido de Mathworks:
<http://de.mathworks.com/help/simulink/sfg/what-is-an-s-function.html>
- Mercedes-Benz Active Body Control - Fully Active Suspension Technology* (2012). [Película].
- Michalek, D., Gehsat, C., Trapp, R., & Bertram, T. (2005). Hardware-in-the-loop-Simulation of a Vehicle Climate Controller with a combined HVAC and Passenger Compartment Model. *International Conference on Advanced Intelligent Mechatronics*, 3.
- Misselhorn, W. E., Theron, N. J., & Els, P. S. (Enero de 2006). Investigation of hardware-in-the-loop for use in suspension development. *Vehicle System Dynamics*, 44, 65-81. doi:10.1080/00423110500303900
- MSC ADAMS. (s.f.). *ADAMS Documentation*. Obtenido de MSC Software:
<https://simcompanion.mscsoftware.com/infocenter/index?page=content&cat=1VMO50&channel=DOCUMENTATION>
- National Instruments. (s.f.). *MicroNova Uses LabVIEW FPGA for Exact and Comprehensive Engine Simulation*. Obtenido de National Instruments:
<http://sine.ni.com/cs/app/doc/p/id/cs-6577>

- Naus, G., Ploeg, J., Van de Molengraft, M., Heemels, W., & Steinbuch, M. (2010). Design and implementation of parameterized adaptive cruise control An explicit model predictive control approach. *Control Engineering Practice*, 18, 882-892.
- Oertel, C. (2008). Einführung in die Systemdynamik. *FH Brandenburg*.
- Oertel, C. (2011). RMOD-K Formula Documentation. *FH Brandenburg*.
- Oertel, C. (s.f.). Kart Kursmodell. *FH Brandenburg*.
- Popular Science. (s.f.). *How NASA Goes to Space Without Leaving the Earth*. Obtenido de Popular Science: <http://www.popsci.com/how-nasa-goes-to-space-without-leaving-earth>
- Racing Car Dynamics. (4 de Octubre de 2014). *THE ABSOLUTE GUIDE TO THE RACING TYRES – PART 1: LATERAL FORCE*. Obtenido de Racing Car Dynamics: <http://racingcardynamics.com/racing-tires-lateral-force/>
- Rodríguez, D., & Bordóns, C. (2005). Análisis y Control de Sistemas en Espacio de Estado. En *Apuntes de Ingeniería de Control*. Departamento de Ingeniería de Sistemas y Automática.
- Sarhadi, P., & Yousefpour, S. (2014). State of the art: hardware in the loop modeling and simulation with its applications in design, development and implementation of system and control software. *International Journal of Dynamics and Control*, 3, 470-479. doi:10.1007/s40435-014-0108-3
- Shigley, J. a. (2004). *Mechanical Engineering Design* (7 ed.). McGraw Hill.
- SIMPACT. (s.f.). *SIMPACT Automotive Flyer*. Obtenido de SIMPACK: http://www.simpack.com/fileadmin/simpack/doc/flyer/SIMPACT_3DS_Flyer_Automotive_Int.pdf
- SIMPACT. (s.f.). *SIMPACT HIL&SIL Flyer*. Obtenido de SIMPACK: http://www.simpack.com/fileadmin/simpack/doc/flyer/SIMPACT_3DS_Flyer_Realtime_Int.pdf

- Sorniotti, A., Morgando, A., & Velardocchia, M. (2006). Active roll control: system design and hardware-in-the-loop test bench. *Vehicle System Dynamics*, 44, 489-505.
- SpeedLab. (s.f.). *How suspension works*. Obtenido de SpeedLab: <http://www.speedlab.com.ph/index.php/shop-services/suspension-works>
- Waltermann, P. (2009). *Hardware-in-the-loop The Technology for Testing Electronic Control units in Automotive Engineering*. Paderborn: dSPACE GmbH.
- Yi, Z., Min-min, X., Jin-yi, Q., & Hu, Z. (2010). Research on co-simulation using ADAMS and MATLAB for automobile active suspension system. *2010 International Conference on Computer Application and System Modeling (ICCSM) 2010*. Institute of Electrical & Electronics Engineers (IEEE). doi:10.1109/iccasm.2010.5622268
- Ying, F., Yiming, G., & Hongni, Z. (Octubre de 2010). Control for vehicle handling stability based on ADAMS and Matlab. *2010 International Conference on Computer Application and System Modeling (ICCSM) 2010*. Institute of Electrical & Electronics Engineers (IEEE). doi:10.1109/iccasm.2010.5620413
- Zhang, X., & Mi, C. (2011). *Vehicle power management: modeling, control and optimization*. Springer Science & Business Media.