Michigan Technological University
Digital Commons @ Michigan Tech

Dissertations, Master's Theses and Master's Reports - Open

Dissertations, Master's Theses and Master's Reports

2015

DEVELOPMENT OF HARDWARE-IN-THE-LOOP SIMULATION SYSTEM FOR ELECTRIC POWER STEERING CONTROLLER TESTING

Chuanliangzi Liu
Michigan Technological University

Follow this and additional works at: https://digitalcommons.mtu.edu/etds

Part of the Electro-Mechanical Systems Commons

Copyright 2015 Chuanliangzi Liu

Recommended Citation

Liu, Chuanliangzi, "DEVELOPMENT OF HARDWARE-IN-THE-LOOP SIMULATION SYSTEM FOR ELECTRIC POWER STEERING CONTROLLER TESTING", Master's report, Michigan Technological University, 2015.
https://doi.org/10.37099/mtu.dc.etds/992

Follow this and additional works at: https://digitalcommons.mtu.edu/etds

Part of the Electro-Mechanical Systems Commons

DEVELOPMENT OF HARDWARE-IN-THE-LOOP SIMULATION SYSTEM FOR ELECTRIC POWER STEERING CONTROLLER TESTING

By

Chuanliangzi Liu

A REPORT

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Mechanical Engineering

MICHIGAN TECHNOLOGICAL UNIVERSITY

2015

© 2015 Chuanliangzi Liu

This report has been approved in partial fulfillment of the requirements for the Degree of MASTER OF SCIENCE in Mechanical Engineering.

Department of Mechanical Engineering – Engineering Mechanics

Report Advisor: *Dr. Bo Chen*

Committee Member: *Dr. John E. Beard*

Committee Member: *Dr. Jeffrey B. Burl*

Department Chair: *Dr. William W. Predebon*

Contents

LIST OF FIGURES v
ACKNOWLEDGEMENTS vii
ABSTRACT viii
1 INTRODUCTION 1
1.1 Motivations 1
1.2 Literature Review 1
1.2.1 Hardware-In-the-Loop Simulation 1
1.2.2 Application of HIL Simulation for EPS Testing 2
2 HARDWARE IN THE LOOP SIMULATION 4
2.1 Concept of Hardware-In-the-Loop Simulation 4
2.2 dSPACE EcoLine Simulator 5
2.2.1 DS1006 Processor Board 7
2.2.2 DS2211 I/O board 7
2.2.3 DS5202 FPGA Base Board 8
2.2.4 DS5202 Electric Motor HIL (EMH) Solution 8
2.2.5 Power Supply 8
2.2.6 Wiring and Signal Conditioning Components 8
3 EPS HIL SIMULATION SETUP 10
3.1 EPS system 10
3.2 Overview of EPS HIL Simulation Setup 12
3.3 Sensor Signal Generation 14
3.3.1 Steering Wheel Torque Sensor Simulation 14
3.3.2 Current Sensor Simulation 15
3.3.3 Position Sensor Simulation 16
3.4 PWM Signal Measurement 18
3.5 CAN Module Setup 19
3.5.1 CAN Setup Blocks 20
3.5.2 CAN Communication Block 20
4 ASM VEHICLE DYNAMICS MODEL 22
4.1 Introduction to ASM Vehicle Dynamics Model 22

4.2 ASM Vehicle Dynamics Model Description 23
4.2.1 Vehicle Movement Subsystem 24
4.2.1.1 Vehicle Motion Car 25
4.2.1.2 Vehicle and Wheel Velocities 26
4.2.1.3 Mass Forces and Torques 27
4.2.1.4 Coordinate Transformation 28
4.2.2 Suspension Subsystem 28
4.2.2.1 Rotation Switch 29
4.2.2.2 Front Symmetric Suspension Kinematics 32
4.2.2.3 Suspension Compliance 36
4.2.2.4 Relative Velocities 40
4.2.2.5 Relative Position 41
4.2.2.6 Front Suspension Forces 42
4.2.2.7 Generalized Front Suspension Forces 43
4.2.3 Steering Subsystem 46
4.2.3.1 Force on Steering Rack 47
4.2.3.2 Generalized Force and Torques 48
4.2.3.3 Steering Mass 49
4.2.3.4 Displacement of Steering rack 50
4.2.4 Tire Subsystem 50
5 MODEL INTEGRATION AND OFFLINE SIMULATION OF INTEGRATED MODEL 51
5.1 The Integration of ASM Model with Steering Model 51
5.2 Offline Validation Results of the Integrated Mechanical Model 55
5.2.1 Maneuver Signal Design 55
5.2.2 Ramp Steer Test 61
5.2.3 Single Lane Change Test 66
6 CONCLUSIONS 71
REFERENCES 72
APPENDIX 74

LIST OF FIGURES

Figure 2.1 Hardware-In-the-Loop System Architecture 4
Figure 2.2 dSPACE EcoLine Simulator – Engine Solenoid 6
Figure 2.3 dSPACE HIL Simulator Subsystems 6
Figure 3.1 Physical Structure of an EPS system 10
Figure 3.2 EPS electric control system 11
Figure 3.3 dSPACE HIL test bench for EPS ECU Testing 12
Figure 3.4 Overview of dSPACE HIL Simulation for EPS system 13
Figure 3.5 DS2211 I/O Board Sensor Simulation Interface 14
Figure 3.6 DS5202 EMH Board Sensor Simulation Setup Blocks 15
Figure 3.7 Sine Encoder Signals 17
Figure 3.8 DS5202_EMH_ANALOG_SENSOR SETUP 18
Figure 3.9 PWM signal measurement blocks 19
Figure 3.10 CAN Module Setup blocks 20
Figure 3.11 RTICANMM MainBlock in transmitting 'Vehicle Speed' signal 21
Figure 4.1 ASM Vehicle Dynamics model 22
Figure 4.2 Vehicle Dynamics Subsystem Upper Level Block Diagram 23
Figure 4.3 ASM System of Coordinates 24
Figure 4.4 Suspension Subsystem 29
Figure 4.5 Camber Angle 30
Figure 4.6 Caster Angle 31
Figure 4.7 Toe Angle 32
Figure 4.8 Wheel Position 33
Figure 4.9 Wheel Orientation 34
Figure 4.10 Spring & Damper & Stabilizer Displacements 35
Figure 4.11 Compliance Longitudinal Force 37
Figure 4.12 Compliance Lateral Force 37
Figure 4.13 Compliance Long Torque 38
Figure 4.14 Compliance Aligning Torque 39
Figure 4.15 Front spring forces 43
Figure 4.16 Block diagram of Steering Subsystem 46
Figure 4.17 Electric Power Steering Torque 48
Figure 5.1 Integrated Mechanical Model Input/Output 51
Figure 5.2 The Block Diagram of Integrated Mechanical Model 52
Figure 5.3 Model Integration Input/Output in ASM model 54
Figure 5.4 (a) Chirp Signal Subsystem in Simulink 56
Figure 5.5 Generated Chirp Signal 57
Figure 5.6 Driving path of a ramp steering input 58
Figure 5.7 Ramp signal design in Simulink 59
Figure 5.8 Generated Ramp Signal 60
Figure 5.9 Driving path of Single Lane Change 60

Figure 5.10 Single Lane Change waveform 61
Figure 5.11 Ramp Steer Time Plots@20kph 63
Figure 5.12 Ramp Steer Cross Plot @20kph 63
Figure 5.13 Ramp Steer Time Plots @60kph 64
Figure 5.14 Ramp Steer Cross Plot @60kph 65
Figure 5.15 Ramp Steer Time Plots @100kph 66
Figure 5.16 Ramp Steer Cross Plot @100kph 66
Figure 5.17 Single Lane Change Time Plots @20kph 67
Figure 5.18 Single Lane Change Cross Plot @20kph 68
Figure 5.19 Single Lane Change Time Plots @60kph 69
Figure 5.20 Single Lane Change Cross Plot @60kph 69
Figure 5.21 Single Lane Change Time Plots @100kph 70
Figure 5.22 Single Lane Change Cross Plot @100kph 70

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Bo Chen, for giving me the opportunity to work on this research project, providing the work space and equipment, and her aspiring guidance during the project work.

I would like to thank the rest of my committee members: Dr. John E. Beard and Dr. Jeffrey B. Burl, for their valuable time, encouragement and insightful comments.

My sincere thank also goes to Mr. Anthony Champagne and Mr. Keyur Patel from Nexteer Automotive, for their guidance and technical support in this project.

I would like to thank Senior Design Team 6 supervised by Dr. Jeffrey B. Burl in Michigan Tech Electrical and Computer Engineering Department, for their calibration work in this project.

I thank my fellow labmates in Intelligent Mechatronics and Embedded Systems Laboratory (IMES), especially to Ming Cheng, for her good suggestions and assistance during my research.

Last but not least, I would like to thank my family for their support and encouragement through my entire life especially during my research project in the past year.

ABSTRACT

The Electronic Control Unit (ECU) of an Electric Power Steering (EPS) system is a core device to decide how much assistance an electric motor applies on steering wheel. EPS ECUs play an important role in EPS systems. The effectiveness of EPS ECUs needs to be thoroughly tested before mass production. Hardware-in-the-loop (HIL) simulation provides an efficient way for the development and testing of embedded controllers. This report focuses on the development of HIL system for testing EPS controllers. The hardware of the HIL system employs a dSPACE HIL simulator. The EPS plant model is an integrated model consisting of Vehicle Dynamics model of the dSPACE Automotive Simulation Model (ASM) and the Nexteer model. The report presents the design of EPS HIL system, the simulation of sensors and actuators, the functions of ASM Vehicle Dynamics model, and the integration method of ASM Vehicle Dynamics model with Nexteer model. The offline simulation of the integrated model is performed and the results for different driving maneuvers are presented. The real-time HIL testing will be conducted in the future to examine the performance of an entire HIL system.

1 INTRODUCTION

1.1 Motivations

With increasing demands on driving handiness flexibility, ride comfort, and safety, EPS system has been gaining popularity in recent years and widely used in modern cars. An ECU is a core component of EPS system since it decides how much assistant power the EPS provides in different situations. Considering the Steering System deterministic influence on vehicle stability, developing algorithms for an EPS ECU is not only for the effectiveness of the steering system, but also the whole car performance. Therefore, a repeated testing of EPS ECUs at an early development phase, prior to road testing, is necessary to check the reliability under various operating scenarios.

HIL Simulation is a widely used technology in automotive industry for the development and testing of real-time embedded control systems. HIL system provides an effective simulation of physical plant for ECU design and testing. HIL testing shortens the development time; cuts down the cost; and reduces the risk from in-vehicle driving test, while improves the ECU software quality at the same time. HIL system enables to test the behavior and effect of EPS in various critical situations.

In this report, HIL system serves as a simulated vehicle to connect to an EPS ECU and test its control performance in real time. The goal of this research is to develop a dSPACE HIL system that can be used to test an EPS ECU for various driving maneuvers in the laboratory of IMES.

1.2 Literature Review

1.2.1 Hardware-In-the-Loop Simulation

Waeltermann [1] provides an overview of the HIL technology. He defines HIL simulation as a method used in the electronics development process in which real subsystems, including one or more ECUs, controllers, or intelligent mechatronic components interact with subsystems that are simulated in real time. The basic hardware components of HIL system include host PC, real-time processor system, I/O boards and signal conditioning, Bus systems, electrical loads and load simulation, electrical fault simulation, real components and power supply. The software

components consist of the operating software MATLAB/Simulnk, the real-time software and the dynamic plat model. Several Automotive HIL applications are thoroughly described, such as engine test system, vehicle dynamics and electric drives.

Pfister et al. [2] gives a brief summarization of the developments of HIL simulation technology. The significance of the wide adoption of HIL simulation in automotive industry is explained in details. First, HIL is an effective tool for designing, testing and evaluating ECUs. Second, HIL makes it possible to assess the performance of final products in advance of prototyping and also make design optimizations timely. Third, HIL which uses computer simulations is able to reduce development cost and shorten product development cycles. Forth, the rapid development in simulation software and computer hardware facilitates the use of HIL which requires real-time simulation.

Schuette and Waeltermann [3] demonstrate the use of HIL simulation for the development of modern vehicle dynamics controllers. The specific sensors and actuators commonly used for vehicle dynamics control and the interfaces for HIL simulation are described. The simulation models are outlined. Application examples of HIL simulation for vehicle dynamics ECUs are presented.

1.2.2 Application of HIL Simulation for EPS Testing

Grunewald [4] introduces functional testing of an electric power steering using HIL simulations. A HIL test bench solution developed by Daimler AG is employed for testing functionalities of EPS. The paper introduces a simulation environment for the vehicle dynamics with an EPS connected to it as a real device. The complex EPS software is examined, including functions such as active return, steering feel function, active damping, and etc. Not only the behavior of the steering system is analyzed, but also its interaction with other systems is evaluated by means of combined test benches.

Lee et al. [5] focused on the establishment of the Hardware-In-the-Loop Simulation (HILS) system to be applied in the development process of an electronic steering device. The developed electronic steering device was installed in the HILS system and applied with a control algorithm, which could accomplish improvements on the precision of steering torque and the feeling of steering. In addition, malfunction situations were tested and self-diagnosis function was verified.

Kim and Song [6] aim at developing EPS control logic for the reduction of steering torque exerted by a driver, realization of various steering feels, and improvement of return-to-center performance. To verify the proposed control logic, a HILS system was constructed with a load motor attached to a rack-and-pinion system to simulate the load resulting from lateral and longitudinal tire forces from road.

Segawa et al. [7] conducted preliminary study concerning quantitative analysis of steering system using HIL Simulator.

2 HARDWARE IN THE LOOP SIMULATION

This chapter introduces the basic concept and architecture of HIL system and describes the components of dSPACE HIL Simulator. For each component, their functionalities and hardware configuration are presented. The detailed introduction of dSPACE HIL simulator components facilitates to further discussions of their applications in EPS HIL system in the next chapter.

2.1 Concept of Hardware-In-the-Loop Simulation

HIL simulation is an increasingly prevalent technique in automotive industries, which has been used in the development and test of complex real-time embedded systems. Figure 2.1 illustrates the basic architecture of the HIL system.

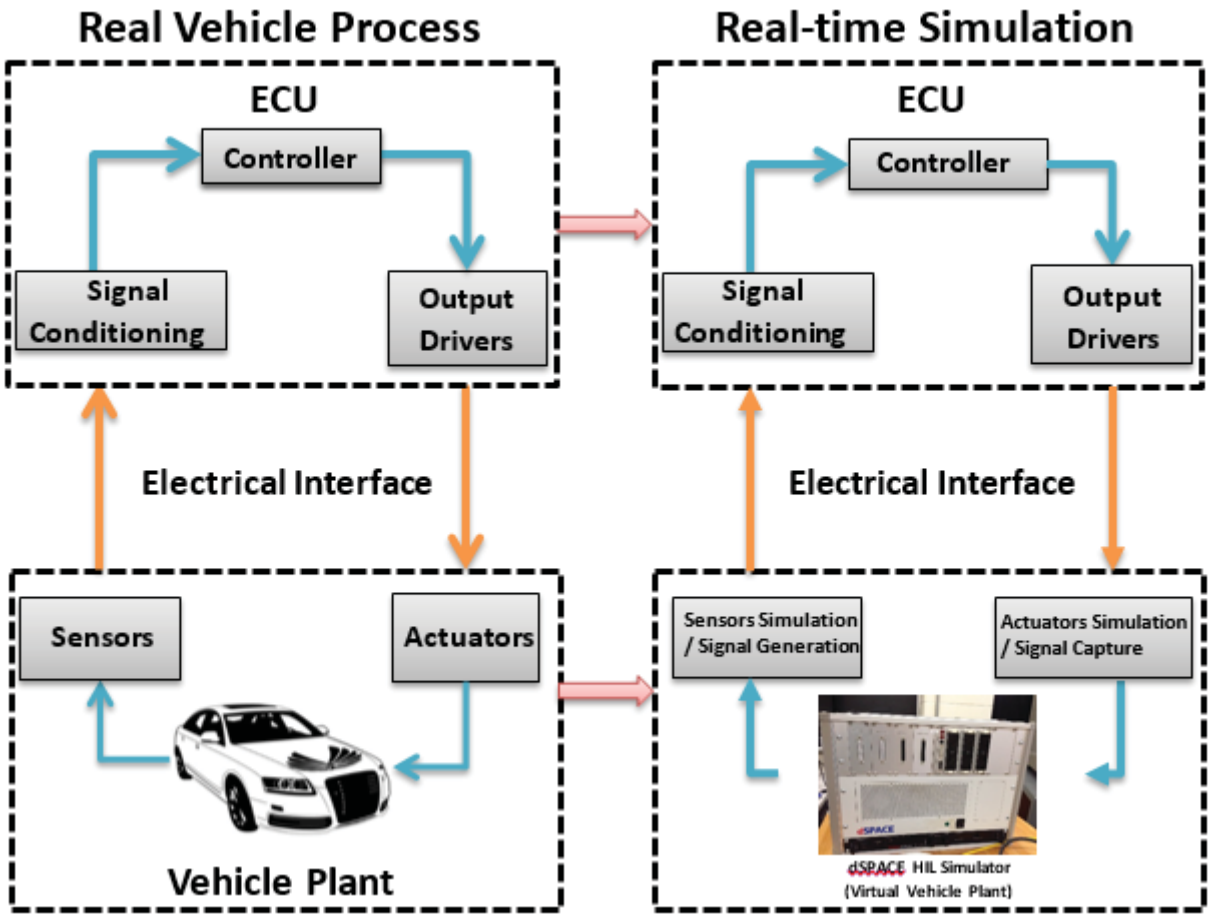

Figure 2.1 Hardware-In-the-Loop System Architecture

As shown in Figure 2.1, HIL simulation includes plant simulation, sensors and actuators emulation. The plant is simulated as a dynamic system, represented by a series of mathematical equations. The electrical emulations act as the interface between the plant simulation and the embedded system under test. The procedure of HIL system process is presented as follows: emulated sensor signals from plant simulation are read by embedded controller under test, which will generate actuator control signals by implementing control algorithms in the controller. Then, the actuation signals from controller will update the variables in the plant simulation. This loop continues to examine the performance of embedded control algorithms. Therefore, the core concept of HIL simulation is to develop and test an embedded system by connecting it with a virtual plant, which is emulated from dynamic models in HIL system platform.[1] [8]

Nowadays, HIL simulation becomes a fundamental and indispensable technology in automotive applications. First, it provides an efficient way to enhance the quality of testing. Second, the tight development schedules always require most of the controller testing to be completed via HIL simulation before the prototype is available. Third, HIL simulation reduces the testing cost.[2]

2.2 dSPACE EcoLine Simulator

dSPACE EcoLine simulator is a HIL simulator to test ECU. There are five variants of simulators tailored to test different ECU types. The one in IMES Lab is Engine Solenoid Simulator, used for HIL simulation of ECUs for engines with solenoid injectors. It is connected to a host PC by a bus for purpose of downloading the real-time model to the simulator and controlling the simulation [9].

Figure 2.2 shows the front view of the dSPACE EcoLine simulator in IMES Lab. The general structure with specific modules and components on front panel are designated in the figure. Figure 2.3 shows the hardware components in dSPACE HIL simulator and the internal connection and signals between each component.

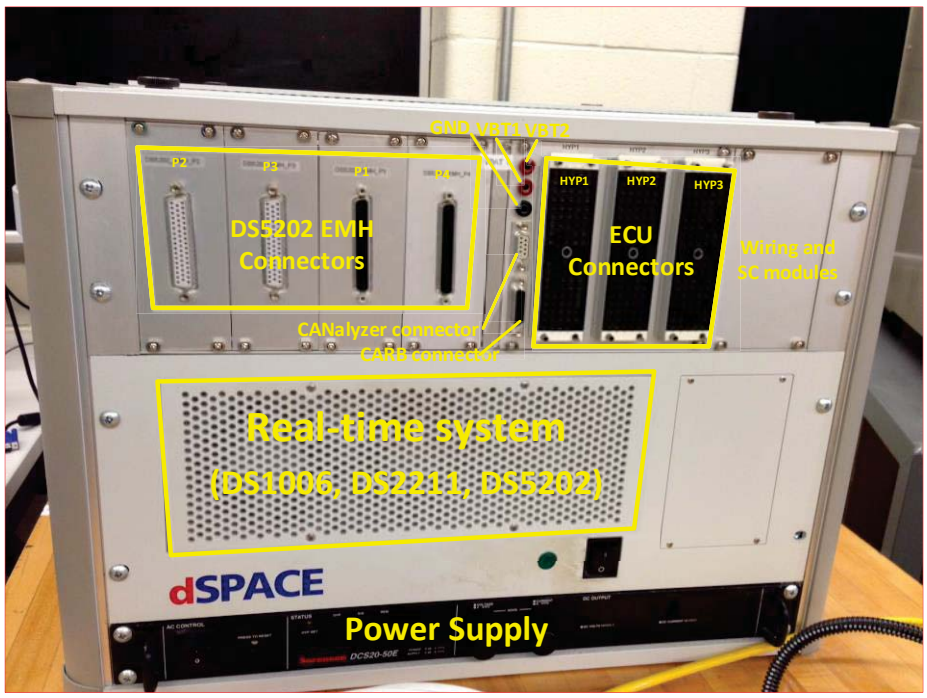

Figure 2.2 dSPACE EcoLine Simulator – Engine Solenoid

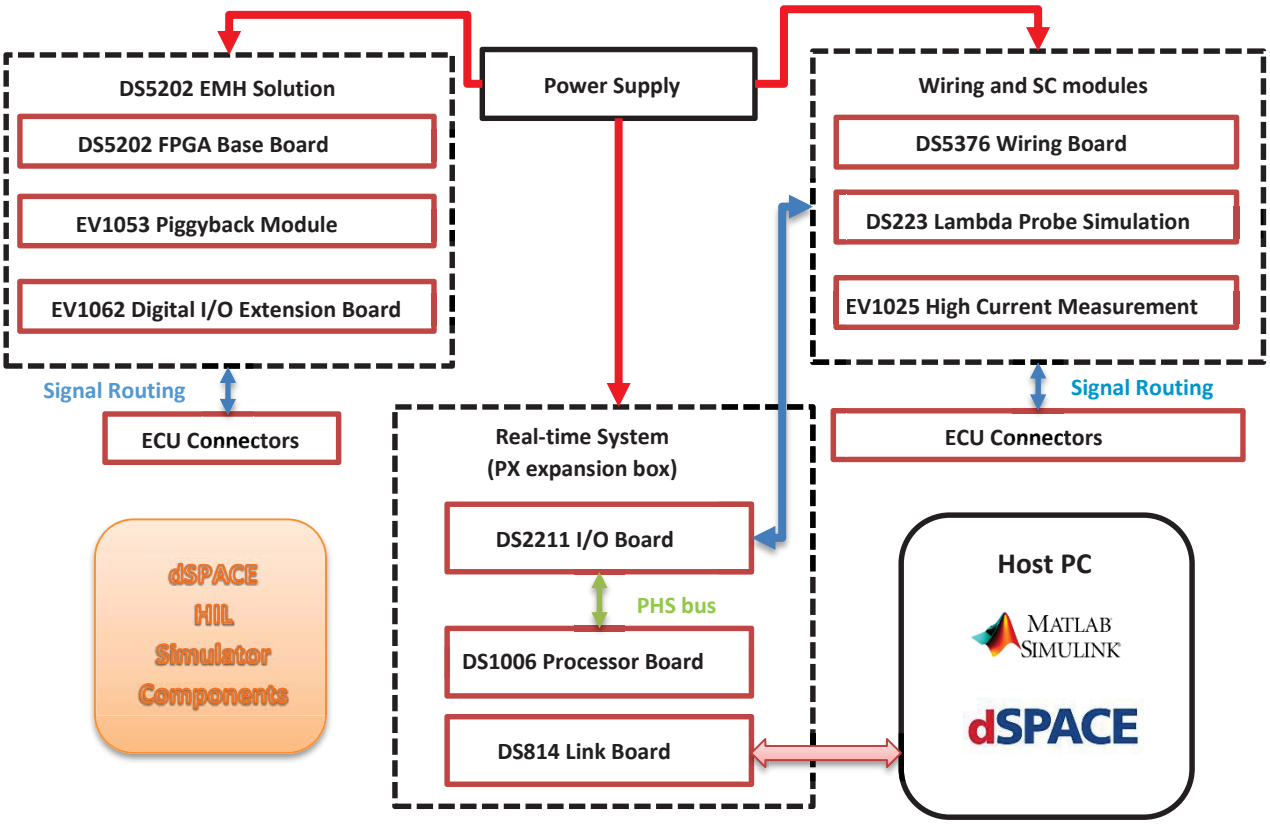

Figure 2.3 dSPACE HIL Simulator Subsystems

As illustrated in Figure 2.3, the core components of the real-time simulator are DS1006 Processor Board and DS2211 HIL I/O Board. DS5202 FPGA Base board is added with Electric Motor HIL (EMH) Solution to simulate the electric motor. Engine signals from the engine ECU are routed to the real-time system by DS5376 Wiring Board and two signal conditioning modules, DS223 and EV1025 [9]. Simulink models can be directly downloaded into real-time system through auto-code generation. The online calibration task can be realized through dSPACE ControlDesk Next Generation (CDNG). Elaboration of each component is presented in the following subsections.

2.2.1 DS1006 Processor Board

The DS1006 Processor Board, core of the dSPACE's modular hardware, computes the simulation model in real time. Also, it communicates with DS2211 I/O board via PHS (Peripheral High-Speed) bus. The DS1006 Processor Board is built on Quad-Core AMD Opteron™ processor, a multi-processor system that can be set up by assigning each individual core to compute different parts of the model in parallel. In this way, the multi-core processor improves the computing speed significantly.[10]

2.2.2 DS2211 I/O board

The DS2211 I/O board offers a sensor and actuator interface to generate and measure I/O signals and supplies Angular Processing Unit (APU), which provides engine HIL core functions.

The Sensor and Actuator Interface (SAI), consisting of standard I/O and timing I/O, includes several functional units:[11]

- 16 Analog to Digital Converter (ADC) channels
- 20 Digital to Analog Converter (DAC) channels
- 10 Digital to Resistance Converter (D/R) channels for resistance generation
- 16 digital input and digital output channels (Bit I/O)
- 24 PWM signal measurement channels with 16 shared with digital input, used to measure the duty cycle and frequency of a PWM signal input
- 9 PWM signal generation channels, used to generate a PWM signal with specific duty cycle and frequency, which can also be used to generate Square-Wave signals

In this research, PWM signals are measured by channels on DS5202 FPGA Board rather than DS2211 Board because the high-precision requirement of 3-phase PWM signals measurement cannot be satisfied by DS2211. Also, Engine EcoLine is used as a HIL simulator to test EPS ECU rather than an engine ECU.

2.2.3 DS5202 FPGA Base Board

The DS5202 FPGA Base Board is designed for HIL simulations that require high-timing and high-resolution signal generation and measurement such as high-precision digital capturing of 3-phase PWM signals and electric motor simulation. For the purpose of fast and precise response of electric motor signals so as to decide the assistant torque to be applied in real time, using DS5202 to simulate the electric motor is an ideal option. The DS5202 FPGA Base Board can be programmed by different FPGA code and carry different types of I/O piggyback modules for different applications. The normal operation of DS5202 requires compatible version of DS5202 firmware and the I/O piggyback module.[12]

2.2.4 DS5202 Electric Motor HIL (EMH) Solution

The EMH Solution, which is based on the DS5202 FPGA Base Board, combines all of the essential I/O functions necessary for electric motors HIL simulation on just one board. It can both capture high-precision 3-phase PWM signals and simulate position sensor signals.

In this research, the 3-phase PWM signals coming from the EPS ECU are measured by EMH Solution, b and c phase current calculated by DS1006 processor board are then sent back to the ECU via analog output channels on EMH Solution.

In our system configuration, EV1053 Piggyback Module with EV1062 Digital I/O Extension Board is plugged onto the DS5202 FPGA Base Board for electric motor simulation, as shown in Figure 2.3. The EV1053 Piggyback Module provides standard analog and digital inputs and outputs necessary for simulation of an electric motor. The EV1062 Digital I/O Extension Board provides 15 input channels for center-aligned PWM measurement, two high-speed digital output channels and two digital input channels.[12]

2.2.5 Power Supply

The Power Supply Unit is used to simulate the car battery in dSPACE EcoLine simulator. It contains one direct high rail and three switchable high rails and can be remotely controlled by the model simulation. The output voltage of the power supply is 0-20 V, and the maximum total output current is 50A.[9]

2.2.6 Wiring and Signal Conditioning Components

The DS5376 Wiring Board is used to route signals from I/O board to ECU connectors and vice versa.

Signal Conditioning (SC) modules in dSPACE HIL simulator is used for some of the signals which require additional signal conditioning. The types of the SC modules depend on the simulator

variant. In Engine Solenoid variant, DS223 and EV1025 modules are necessary components for signal conditioning.[9]

3 EPS HIL SIMULATION SETUP

This chapter introduces EPS system and presents the HIL simulation setup for EPS system using dSPACE Ecoline simulator. Sensors and actuator simulations are demonstrated with details by introducing the basic principle, setup and interface blocks to be used. The setup and configuration of PWM Signal Measurement and CAN Module are introduced as well.

3.1 EPS system

Figure 3.1 illustrates the physical structure of a steering system. In addition to the components that a conventional column-type steering system always has, including steering wheel, steering column, and rack and pinion mechanism, the EPS system has an assist motor connected to the steering shaft through gears. The motor is a BLDC (brushless dc) motor and provides assisting torque to support driver's steering.

Depending on an electric motor as a power source, EPS system augments the steering effort on steering wheel to improve the portability and controllability of steering comparing to traditional mechanical steering which uses driver's turning force to maneuver.

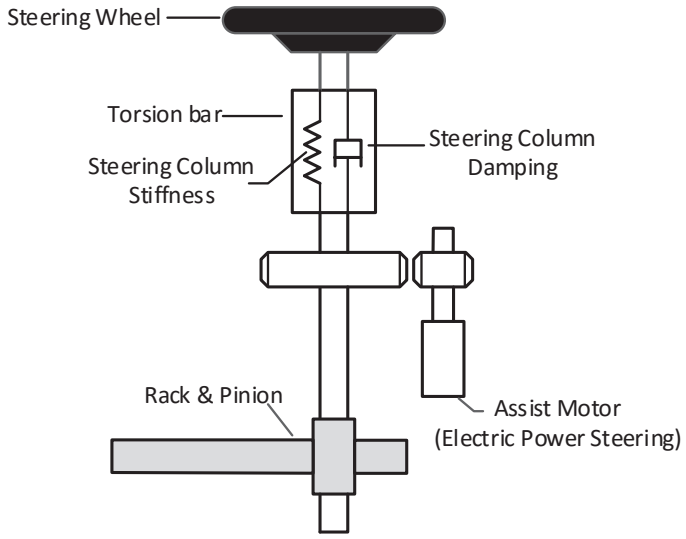

Figure 3.1 Physical Structure of an EPS system

The main purpose of an EPS system, which is to provide assist to a driver, is achieved by an EPS control system, as shown in Figure 3.2.

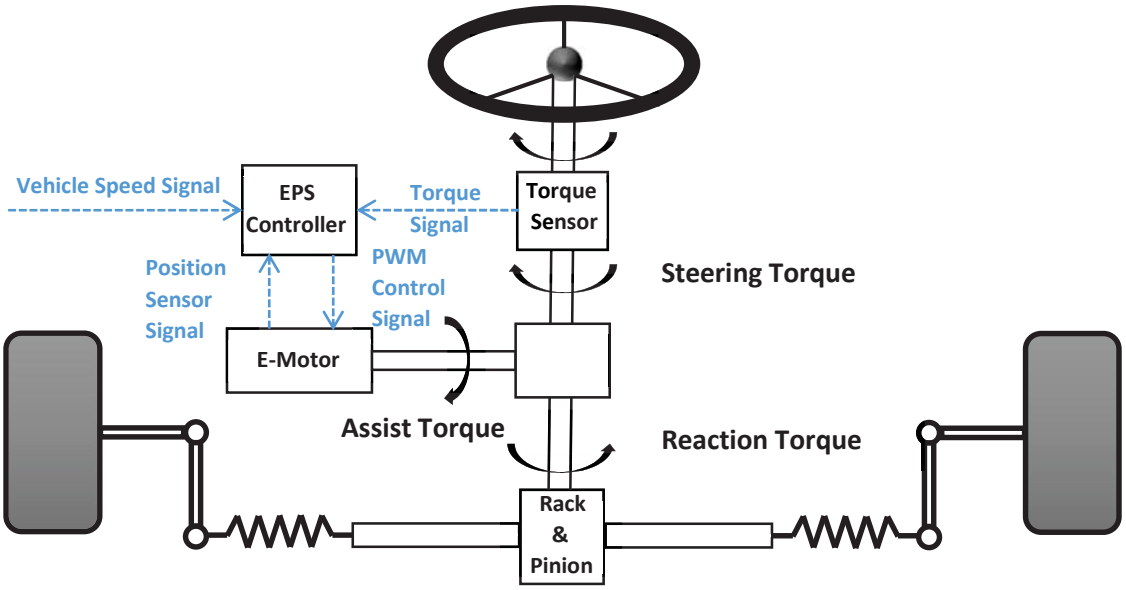

Figure 3.2 EPS Electric Control System

Generally, an EPS system consists of a steering wheel torque sensor, vehicle speed sensor, EPS ECU, electric motor, rack and pinion, and so on.

The basic principle of an EPS system can be summarized as follows: the torque sensor mounted on steering column measures the driver's torque and sends a torque signal to the controller proportional to this torque. Also, the EPS controller takes in vehicle speed signal. These input values are processed in the controller, and a three-phase PWM control signal is generated. The PWM control signals determine the three-phase currents flowing in the electric motor, which decide the motor assist torque. The assist torque is then applied to the steering column through gear mechanism. Also, a motor position sensor measures the motor positions and sends positions signals to EPS controller.

Since the electric motor is connected to the steering column, the magnitude and direction of assist torque will be acted on the wheels via rack and pinion system. The amount of assistance being applied could be varied in different driving conditions whether a vehicle is stopped, driving slowly or fast, which is realized by EPS ECU with specific control logic. EPS system provides more reliable and effective support comparing to other power steering systems.

3.2 Overview of EPS HIL Simulation Setup

For this research, a dSPACE HIL test bench is employed to analyze the behavior of the steering system and test their functionality. Figure 3.3 shows the implementation of the HIL test bench for EPS ECU testing including all the hardware and software used.

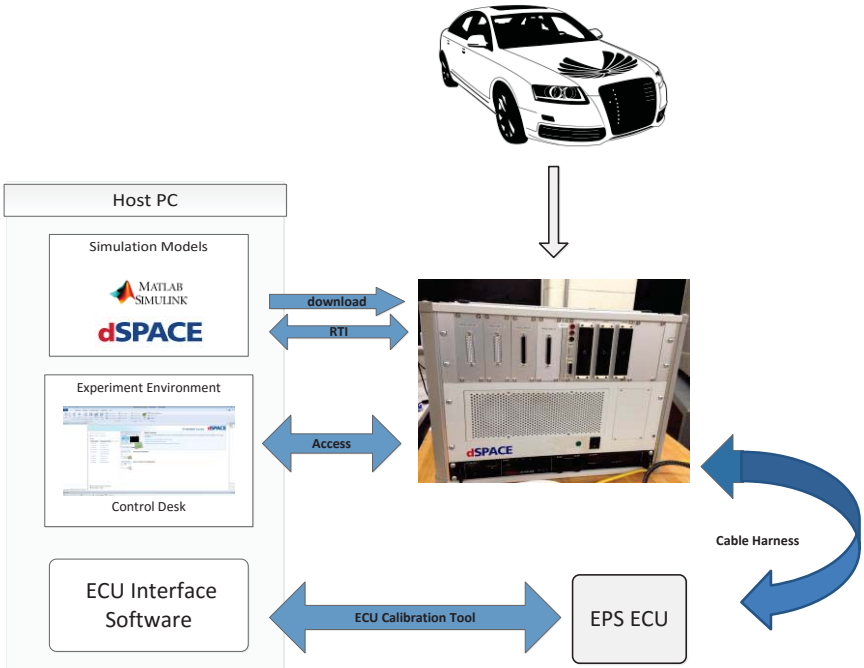

Figure 3.3 dSPACE HIL Test Bench for EPS ECU Testing

As shown in Figure 3.3, a vehicle plant model is built in Simulink and dSPACE ASM. Real-Time Interface (RTI) blocks are necessary in the model to connect the Simulink and dSPACE ASM model to dSPACE I/O board. C code can be automatically generated from dSPACE/Simulink models and downloaded to real-time hardware. The dSPACE simulator can be considered as a virtual vehicle connected with an EPS ECU through cable harness. An ECU Calibration Tool connects the EPS ECU to the host PC through an USB cable. The ECU Interface Software can be used to monitor and change variable values of the ECU. Modifications to variable values is either through a CAN interface or by modifying the source binary file and re-programming the ECU's non-volatile memory. CDNG is the dSPACE experiment software for seamless ECU development. It provides access to simulation platforms and connected bus systems, and can perform measurement.[3]

Figure 3.4 shows an overview of dSPACE HIL simulation for EPS system, including brief illustration of plant model, sensors and actuators simulation, and signal flow between dSPACE HIL simulator and EPS controller.

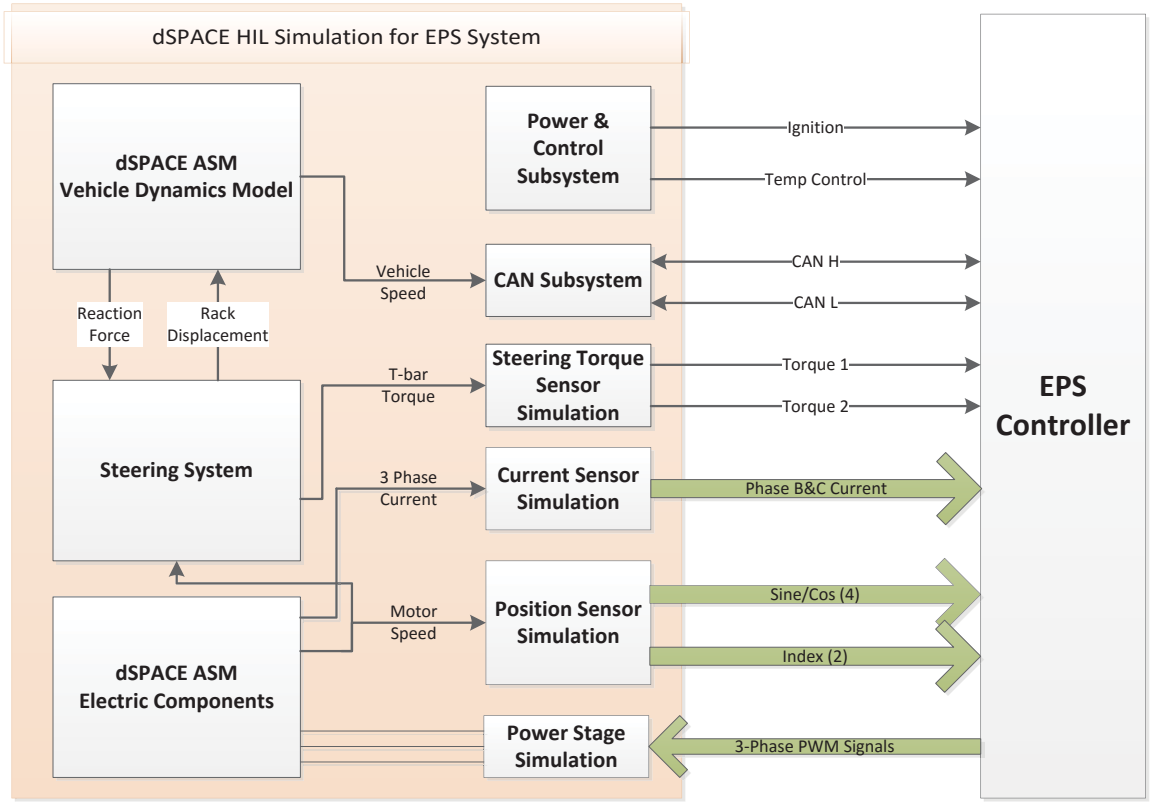

Figure 3.4 Overview of dSPACE HIL Simulation for EPS system

As shown in Figure 3.4, the plant model consists of three main parts, dSPACE ASM Vehicle Dynamics Model, Steering System and dSPACE ASM Electric Components. Reaction force and rack displacement are communicated between the first two models. Detailed analyses of these models are discussed later.

Signals communicated between HIL simulator and EPS controller are shown in Figure 3.4. EPS controller is powered by an ignition signal of dSPACE HIL simulator and a temperature sensor is simulated. The vehicle speed is calculated by the plant model and transmitted through CAN bus. Steering torque sensor is simulated and two torque signals are sent to EPS controller. The EPS controller processes the information by implementing the control logic and determines the 3-phase PWM control signals. A current sensor is also emulated to measure B and C phase current flowing in the electric motor. A position sensor determines the motor position and sends out four sine and

cosine signals and two index signals, which will be explained later. Sensors and actuator emulation, PWM signal measurement and CAN Module setup are discussed in detail below.

3.3 Sensor Signal Generation

In this research, torque sensor and temperature sensor are simulated by using DS2211 I/O board DAC channels. Board and channel numbers for different sensors to be simulated are specified. Detailed discussion on torque sensor simulation will be presented in later subsection.

Because high precision calculation is required by EPS electric motor, DS5202 EMH Solution is used to simulate and measure all the electric motor signals, including three phase motor current signals and motor position signals. DS5202 EMH SETUP block is necessary for all functions used on DS5202 EMH Board.

3.3.1 Steering Wheel Torque Sensor Simulation

Steering Wheel Torque sensor is used to measure the torque that a driver applies on the steering wheel.

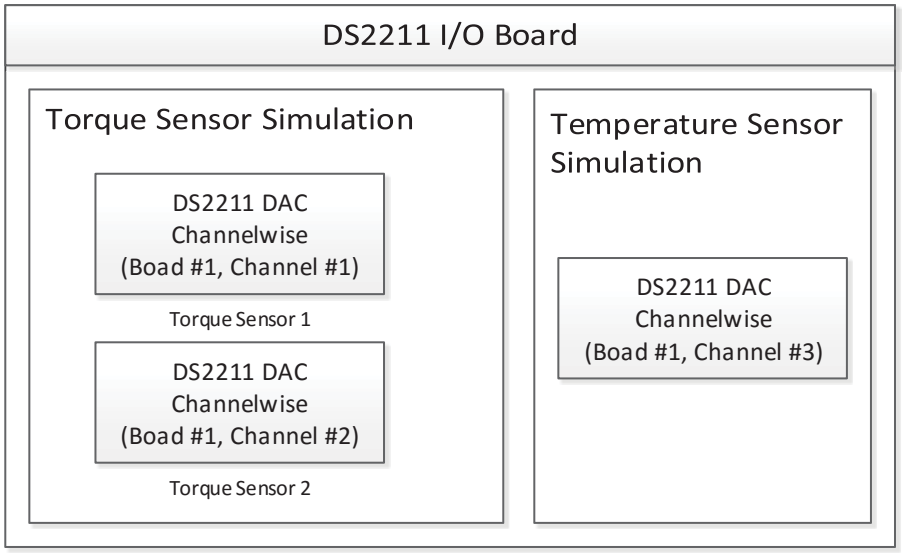

Figure 3.5 DS2211 I/O Board Sensor Simulation Interface

Torque sensor is simulated through DAC channels on DS2211 I/O board, which contain 20 parallel D/A converters with 12-bit resolution. As illustrated in Figure 3.5, the DS2211DAC block is a Channel-wise D/A Conversion block, which is used to write to individual D/A channels. The channel number must be specified for individual channels. The board number should be selected as 1 in our HIL system.

There are 11 DS2211 DAC channels on ECU 2 connector of the dSPACE Ecoline simulator which can be selected in consistent with the channel number specified to simulate the torque sensor.

3.3.2 Current Sensor Simulation

Current sensor is used to measure three-phase current passing through the stator of EPS electric motor. The current signals are simulated as two-phase analog voltage signals through the DAC channel on DS5202 EMH solution board and then sent to EPS controller. Figure 3.6 shows the blocks used for sensor simulation on DS5202 EMH Board.

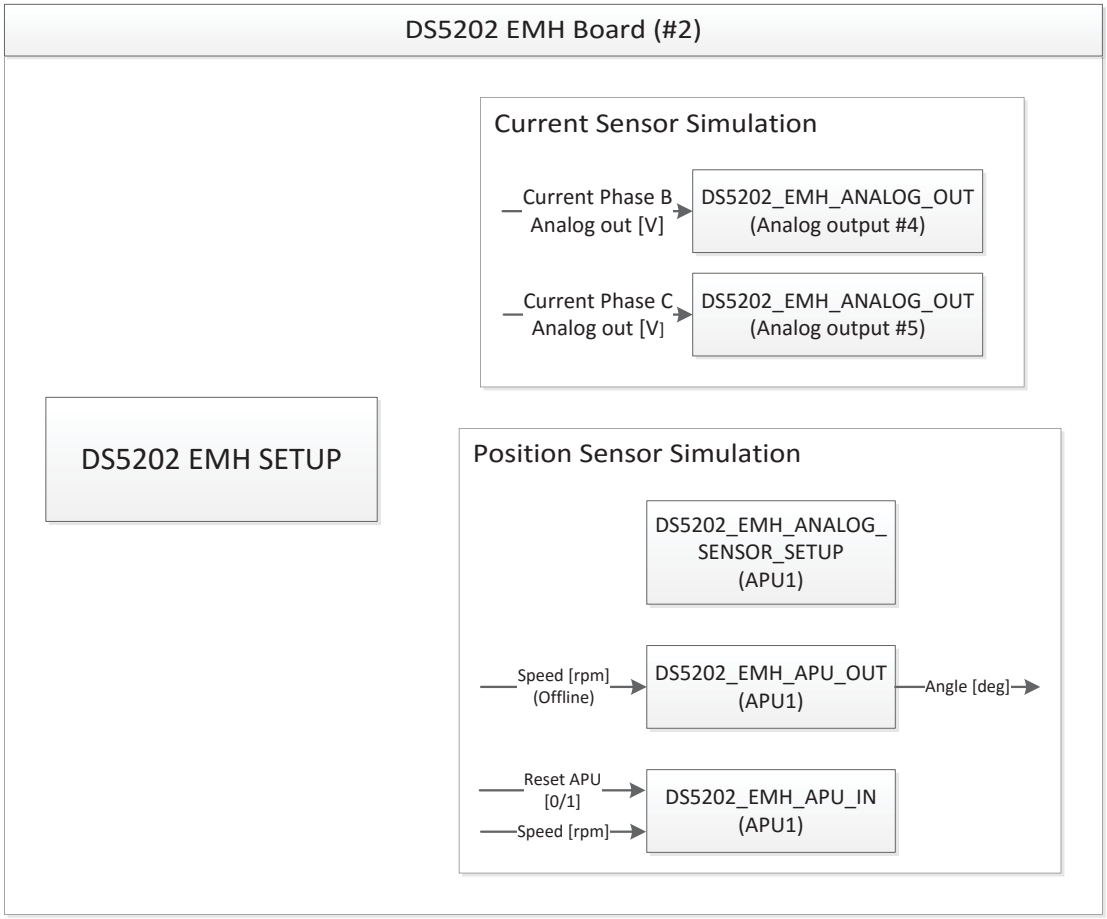

Figure 3.6 DS5202 EMH Board Sensor Simulation Setup Blocks

DS5202_EMH_SETUP block is required if any DS5202 EMH Solution I/O is used in the model. It is used to initialize the DS5202 EMH Solution and configure the signal behavior for all digital inputs. Inside the DS5202_EMH_SETUP block, we are able to specify the signal behavior, high active or low active for each of eight gates of the PWM groups.

Analog output block, DS5202_EMH_ANALOG_OUT, as shown in Figure 3.6, represents a single D/A channel used for simulating current signals. There are 7 analog output channels on DS5202 EMH solution board. Therefore, two output channels on DS5202 EMH solution board are selected to simulate B and C phase current.

Six out of seven analog output channels on DS5202 EMH solution board are on 50-Pin Female Sub-D Connector P2 of dSPACE Simulator Ecoline. Therefore, two of the channels could be selected for current sensor simulation. To be compatible with the Analog output number selected in the DS5202_EMH_ANALOG_OUT block, the pins designated for B and C phase current signals on EPS controller must be connected to the specified pins on Connector P2.

In this research, once the B and C phase motor currents are measured, the corresponding analog voltage signals are calculated and sent to the EPS controller through DS5202_EMH_ANALOG_OUT block.

3.3.3 Position Sensor Simulation

In this system, a sine encoder is simulated as the electric motor position sensor, which is realized by angular processing unit (APU) on DS5202 EMH Solution Board to receive the angular rotor speed and calculate the rotor position. Simulink blocks used in motor position sensor simulation are shown in Figure 3.6.

The simulated sine encoder consists of three output channels, A, B, and C (also called index), as shown in Figure 3.7. The Sine incremental signals A and B are 90 degree phase-shifted with each other for the determination of rotational direction. The output voltage (Vpp) and DC offset voltage are defined as 1.0V and 2.5V, respectively [12]. The index generates one pulse per revolution, which is useful to decide the absolute position by counting the number of full revolutions.

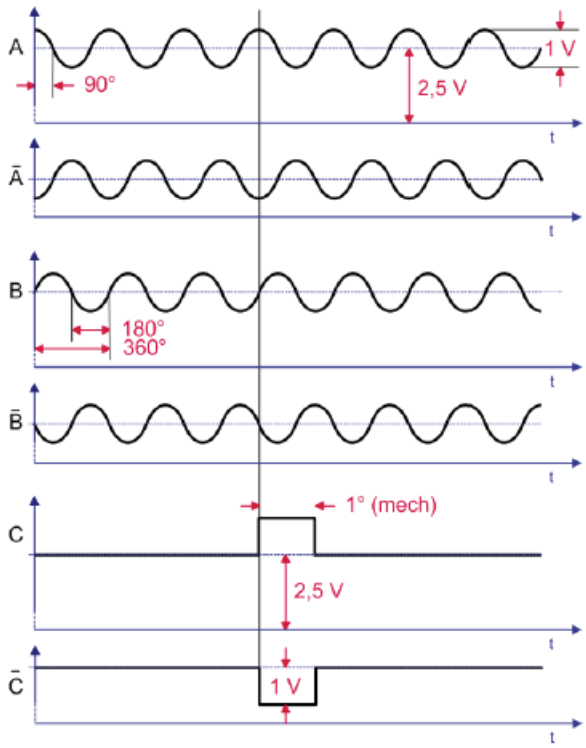

Figure 3.7 Sine Encoder Signals

To generate angular position signals, RTI blocks DS5202_EMH_APU_IN and DS5202_EMH_ANALOG_SENSOR_SETUP must be included in the model. Figure 3.6 illustrates the input and output signals of these two blocks.

The DS5202_EMH_ANALOG_SENSOR_SETUP block is used to generate analog position sensor signals. The DS5202_EMH_APU_IN block, with motor speed as its input, must be used at the same time to set up the APU and generate position sensor information.[12] The parameters configuration for analog sensor setup is shown in Figure 3.8.

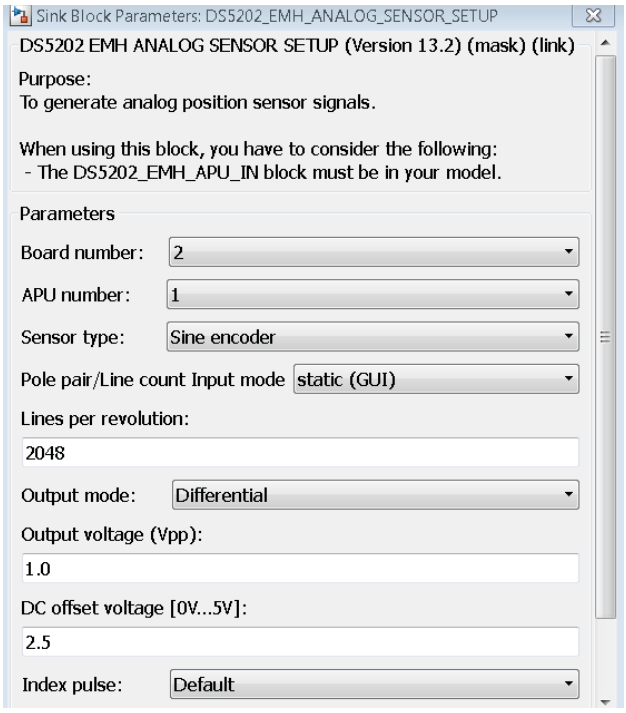

Figure 3.8 DS5202_EMH_ANALOG_SENSOR SETUP

In the DS5202_EMH_ANALOG_SENSOR_SETUP configuration dialog, 'Sine encoder' should be selected as the sensor type to simulate the motor position sensor. The default values of lines number per revolution of encoder, Output voltage (Vpp) and DC offset voltage are 2048, 1.0V and 2.5V, respectively. With selected parameter values of the Sine encoder and a given motor speed, the output signals for the simulated position sensor can be expected, which is an effective way to calibrate the position sensor and will be discussed later. The DS5202_EMH_APU_OUT block is used to generate the absolute APU angle, which is necessary in this research because the absolute motor position angle is needed in the real time model.

Six specified pins on the 50-Pin Female Sub-D Connector P3 of dSPACE Simulator Ecoline, shown in Figure 2.2, are set specifically for the six signals, A, A^1, B, B^1, C (index), C^1 ($index^1$) in Figure 3.7, which are generated by DS5202_EMH_APU_IN. Therefore, the pins of the EPS controller defined for the six motor positions signals must be connected to the corresponding pins on dSPACE P3 Connector, to enable the perfect position sensor signal communication between the dSPACE HIL Simulator and the EPS controller.

3.4 PWM Signal Measurement

For standard PWM measurement, the EV1053/EV1062 provides up to 8 channels with a resolution of 16 bits (The channels are shared with center aligned PWM input signals). Figure 3.9 shows all the blocks used for PWM signal measurement.

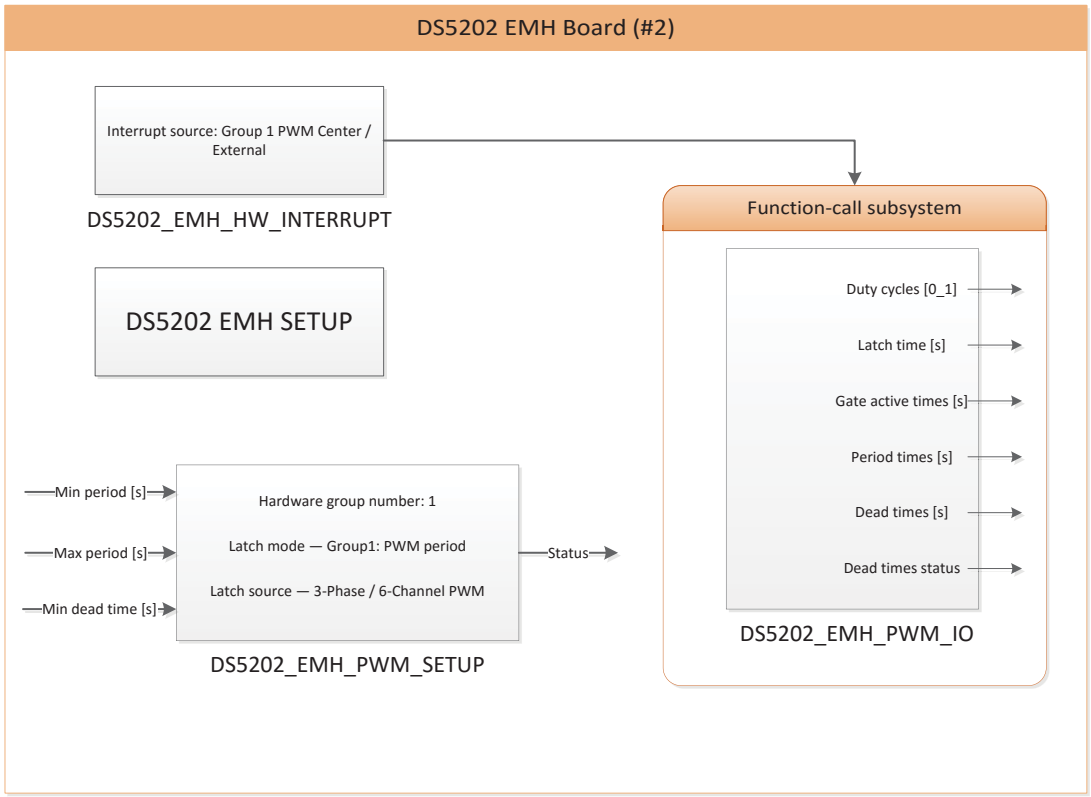

Figure 3.9 PWM Signal Measurement Blocks

This INTERRUPT block defines the external PWM signals received from EPS ECU as trigger source in the model. DS5202_EMH_PWM_IO in a function-call subsystem is used to measure duty cycles, latch time, gate active times, period times and dead times of up to 8 center aligned PWM input signals. The DS5202_EMH_PWM_SETUP block must be in the model at the same time for using the IO block.[12]

DS5202_EMH_PWM_SETUP is for the purpose of setting up one of the center aligned PWM measurement groups. As illustrated in Figure 3.9, the DS5202_EMH_PWM_SETUP block has three inputs, minimum PWM period, maximum PWM period and minimum dead time.[12]

3.5 CAN Module Setup

The RTI CAN MultiMessage Blockset is used for setting up CAN in dSPACE HIL applications. In order to work with RTI CAN MultiMessage Blockset, we need to make sure that the Microsoft Windows SDK compiler is used to build MEX functions which is initialized via the 'mex –setup' command.[13]

Figure 3.10 shows all the blocks used for CAN Module Setup. Detailed explanation of individual blocks is discussed in the following sub-sections.

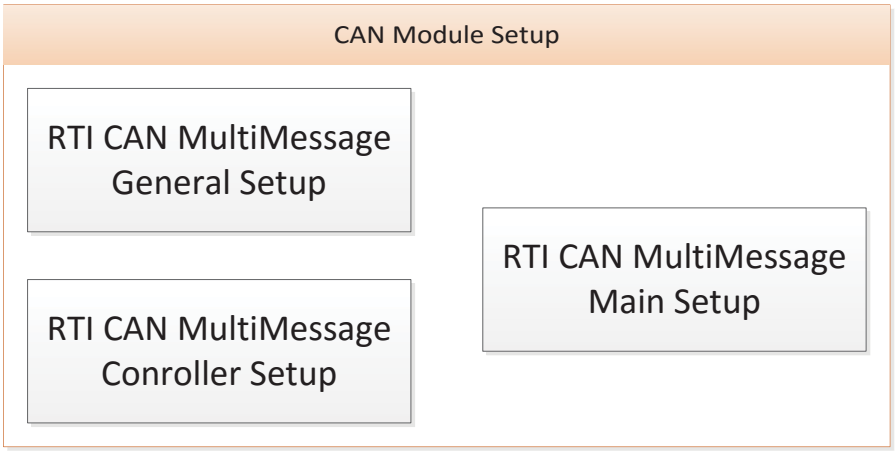

Figure 3.10 CAN Module Setup Blocks

3.5.1 CAN Setup Blocks

The rticanmmlib (RTI CAN MultiMessage Blockset) provides CAN Setup blocks including RTICANMM General Setup and RTICANMM Controller Setup.

For General Setup, the destination folder to save generated files by all RTICANMM blocks in the model must be specified in RTICANMM General Setup block. The folder should be in an absolute or relative path with the model root folder. RTICANMM Controller Setup block is to initialize the CAN Controller by specifying basic information including Transceiver type, Termination resistance and Baud rate. Then use Options Menu in Controller Setup block to create S-Function for Controller Setup block. To be notice that only one RTICANMM General Setup block is necessary in a model, which will apply the settings to all CAN blocks, but each CAN controller must have individual RTICANMM Controller Setup block within a model.[13]

3.5.2 CAN Communication Block

The RTI CAN MultiMessage MainBlock enables users to configure CAN communication general settings. In General Settings option, a database container (DBC) file is selected as database file for the description of CAN communication data. To identify CAN signals, the RTICANMM

MainBlock dialog displays their names together with the names of the corresponding messages as ‘MessageName.SignalName’. On the Model Signals (TX) page, select the signals as TX Signals to be transmitted from the model among all of the available signals. Similarly, the signals to be captured are selected as RX signals on the Model Signals (RX) Page. All the files for RTICANMM communication such as S-function are created by single clicking ‘create’ in the dialog.[13]

In this research, ‘Vehicle Speed’ signal needs to be transmitted from dSPACE simulator to EPS controller via CAN Controller, as shown in Figure 3.4, so RTICANMM blocks are required in the model. RTICANMM MainBlock used for transmitting ‘Vehicle Speed’ signal is illustrated in Figure 3.11.

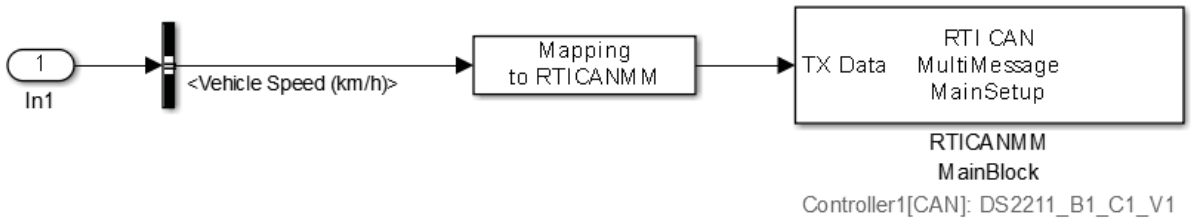

Figure 3.11 RTICANMM MainBlock in Transmitting ‘Vehicle Speed’ Signal

The basic procedure to use RTICANMM blocks in the model can be described as follows. First, the model needs to be saved before adding the first RTICANMM blocks to it. Second, there is a certain sequence to add these blocks into a model. Specifically, RTICANMM General Setup block needs to be added first followed by a RTICANMM Controller Setup block. In the next step, RTICANMM MainBlocks needs to be added in the model for each CAN controller. To be highlighted, ‘Vehicle Speed’ signal is selected as the TX signal to be transmitted by this CAN controller and no messages expect to receive in our situation.[14]

4 ASM VEHICLE DYNAMICS MODEL

ASM Vehicle Dynamics Model is used to simulate steering reaction force, which is the summation of the forces on both steering racks delivered from the road. Steering reaction force has significant influence on vehicle maneuverability so an accurate estimation of reaction force is important to evaluate the vehicle steering behavior and test EPS ECU functions. Subsystems in ASM Vehicle Dynamics Model are explained into details below.

4.1 Introduction to ASM Vehicle Dynamics Model

The ASM Vehicle Dynamics Model from dSPACE is a complete dynamics model used to simulate a vehicle and the environment. The ASM Vehicle Dynamics Blockset provides open Simulink models for real time simulation of vehicle dynamics. They support an effective vehicle controller development process from function design to ECU testing. The open model approach offers flexibility to meet individual requirements by adding or replacing ASM components with customer models.[15]

The ASM Vehicle Dynamics Model is designed as a multibody system with 24 degrees of freedom, which consists of five subsystems including Soft ECU, Engine, Drivetrain, Vehicle Dynamics and Environment, as illustrated in Figure 4.1.

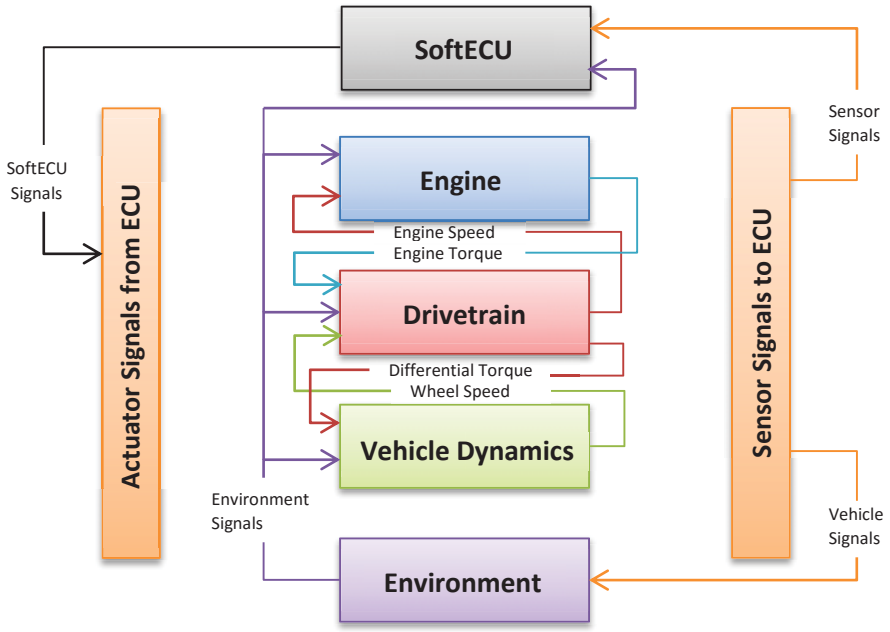

Figure 4.1 ASM Vehicle Dynamics Model

This chapter gives a detailed description of the Vehicle Dynamics Subsystem in ASM Vehicle Dynamics Model, which will be incorporated into EPS Model to perform functional testing of EPS ECU via dSPACE HIL Simulator.

The Vehicle Dynamics Subsystem provides main components, including two semi-empirical tire models, a nonlinear table-based vehicle suspension model with kinematics and compliance, a steering model, aerodynamics model and brake model.[15] In addition, there is a Vehicle Movement Subsystem to simulate the motion of the vehicle. Major descriptions are made for Vehicle Movement, Suspension and Steering Subsystem to discuss how vehicle speed and steering reaction force are calculated from this model.

The block diagram in Figure 4.2 illustrates an upper-level structure of the Vehicle Dynamics Subsystem. Detailed description of the model is discussed in the next sub-section.

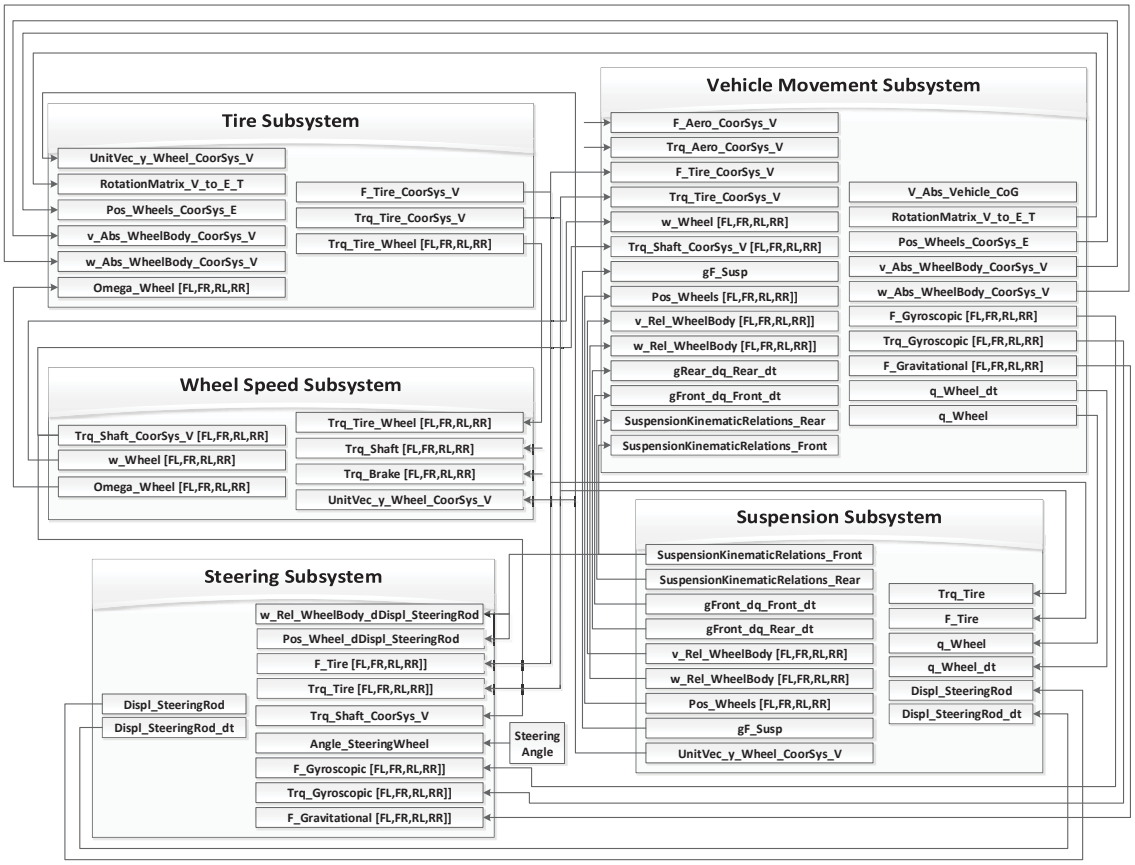

Figure 4.2 Vehicle Dynamics Subsystem Upper Level Block Diagram

4.2 ASM Vehicle Dynamics Model Description

The ASM Vehicle Dynamics model is based on four systems of coordinates, which is illustrated in Figure 4.3.

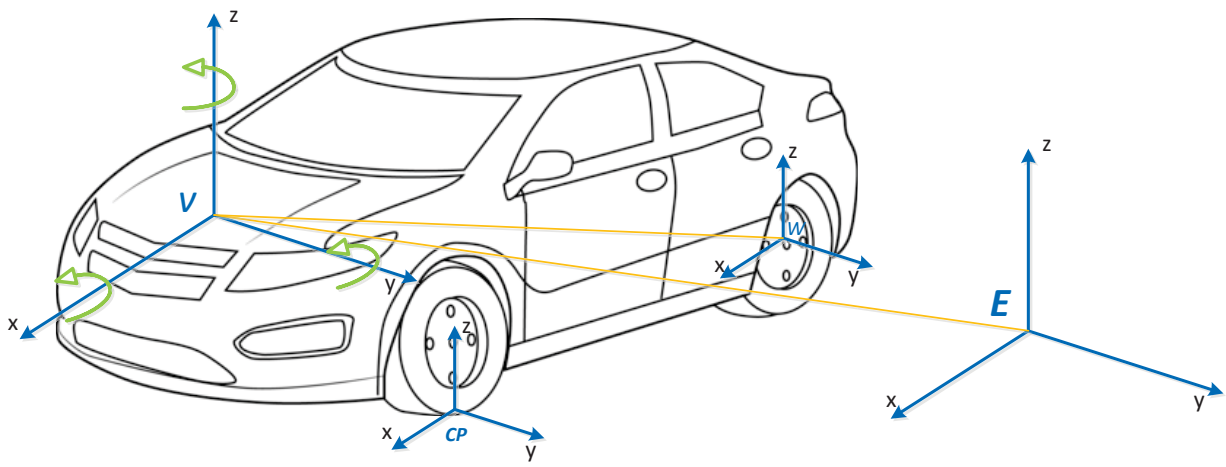

Figure 4.3 ASM System of Coordinates

The four coordinates are listed below:[16]

- Earth coordinate system, index E
  It represents a fixed reference system.
- Vehicle reference coordinate system, index V
  It is fixed to the vehicle body with origin of the coordinate at mid-point between the front wheel centers. The x-axis is in longitudinal direction of the vehicle and points forwards. The y-axis is in lateral direction and points left. According to the coordinate system right-hand rule, the z-axis is vertical and points upwards.
- Wheel coordinate system, index W
  It is fixed on a wheel with origin at the wheel center. The direction of the axis depends on the wheel orientation.
- Contact point coordinate system, index CP
  It has an origin at contact point of wheel and road. Both of the x and y axes are on the road plane.

4.2.1 Vehicle Movement Subsystem

In the Vehicle movement block, the motion of the vehicle and the vertical wheel movements are calculated as functions of tires, aerodynamics, suspension, and mass forces and torques. ASM Vehicle Dynamics uses a multibody system (MBS) technique to write the equation of motion. In this subsystem the equation of motion is solved for the vehicle degrees of freedom. These degrees

of freedom are used to calculate the vehicle and wheel velocities and also to calculate the vehicle and wheel positions in the earth reference system.

Vehicle generalized degrees of freedom are divided into three components: translational vehicle velocities, angular vehicle velocities and vertical speed of the wheel.

4.2.1.1 Vehicle Motion Car

The objective of 'Vehicle Motion' subsystem is to calculate the vehicle degrees of freedom by solving the dynamical equation of motion. In ASM model, a multibody system (MBS) technique is used to generate the equation of motion since the whole vehicle could be considered as a mechanical system with many degrees of freedom, defined as a multibody system. The equation of motion represented as MBS technique can be written as follows:

$$M\ddot{q} = Q \tag{4-1}$$

$$\ddot{q} = M^{-1}Q$$

where

$\ddot{q}$ is the generalized degrees of freedom vector with 10 elements in it, including the vehicle and wheel accelerations and vehicle angular accelerations of vehicle reference system. Then the vehicle and wheel velocities and wheel positions can be calculated by applying integral on the degrees of freedom.

M is a 10×10 generalized mass matrix, which is a function of the wheel position and kinematic suspension relations. It is not constant and is calculated at every simulation step.

Q is a 10×1 generalized forces and torques vector, including generalized forces and torques in vehicle reference coordinate, and generalized forces on wheels. They are calculated from the tire, aerodynamics, mass forces, and torques and suspension forces in the direction of the relevant degrees of freedom.

The generalized forces on vehicle is the sum of all forces acting on vehicle in x,y,z direction respectively. All of these forces are represented in vehicle reference coordinate system, including tire forces on each tire (F_Tire[FL,FR,RL,RR][x y z]), aerodynamic forces (F_Aero_CoorSys_V[x,y,z]), external forces (F_External_CoorSys_V[x;y;z]), gravitational forces of the vehicle and wheels (F_Gravitational[V,FL,FR,RL,RR][x,y,z]), gyroscopic forces of the vehicle and wheels (F_Gyroscopic[V,FL,FR,RL,RR][x,y,z]).

The generalized torques on vehicle consists of two major parts, torque generated from forces, and torque directly acting on the vehicle. The first part can be divided into the torque generated by tire

forces relative to position of wheel, gravitational and gyroscopic forces respect to position of center of gravity on vehicle body. The second part torque includes tire torques (Trq_Tire[FL,FR,RL,RR][x y z]), aerodynamic torques (Trq_Aero_CoorSys_V[x,y,z]), gyroscopic torques (Trq_Gyroscopic[V,FL,FR,RL,RR][x,y,z]), and external torques (Trq_External_CoorSys_V[x;y;z]). The sum of all of these torques in x,y,z direction is the generalized torque on vehicle.

4.2.1.2 Vehicle and Wheel Velocities

In this subsystem, the velocities of the vehicle center of gravity (CoG) and the wheels are calculated according to the generalized degrees of freedom, position of CoG, and the wheels' positions and relative velocities.

The absolute velocity, angular velocity of wheel and vehicle center of gravity (CoG) on vehicle are calculated according to generalized degrees of freedom, relative velocities, position of CoG and wheel positions.

For the vehicle CoG, the translational and angular velocities can be calculated as follows:

$$v_{CoG} = v_V + \omega_V \times r_{CoG} \tag{4-2}$$

$$\omega_{CoG} = \omega_V$$

where

v_V Vehicle velocity in vehicle coordinate system

ω_V Vehicle angular velocity in vehicle coordinate system

r_{CoG} Position of vehicle center of gravity

v_{CoG} Velocity of vehicle center of gravity

ω_{CoG} Angular velocity of vehicle center of gravity

And for the wheel center:

$$v_{Wi} = v_V + \omega_V \times r_{Wi} + \dot{r}_{Wi} \tag{4-3}$$

$$\omega_{Wi} = \omega_V + \omega_{Wi_rel}$$

where

i FL, FR, RL and RR tire

r_{Wi} Wheel position for each tire

$\dot{r}_{Wi}$ Wheel velocity relative to the vehicle

ω_{Wi_rel} Wheel angular velocity relative to the vehicle

v_{Wi} Velocity of wheel center

ω_{Wi} Angular velocity of wheel center

4.2.1.3 Mass Forces and Torques

To calculate the generalized forces and torques, it is necessary to calculate not only the external forces and torques, i.e., tire and aerodynamics, but also the mass forces and torques. The mass forces and torques depend on residual acceleration, gravity acceleration, and angular velocity.

- Gravitational Force

  The production of acceleration of gravity and the mass of entire vehicle is the gravitational force of the vehicle. Acceleration of gravity times mass on each of the four wheels is the gravitational force acting on the corresponding wheels.

- Gyroscopoic Force

  Residual Acceleration times mass of total vehicle is the Gyroscopoic force on the total vehicle. Residual Acceleration times mass on each of the four wheels is the Gyroscopoic force acting on the corresponding wheels.

- Residual Acceleration

The expression of acceleration for each of the wheels can be written as follows:

$$a_{Wi} = \dot{V}_V + (\dot{\omega}_V \times r_{Wi}) + [\omega_V \times (V_{Wi} + \dot{r}_{Wi})] \tag{4-4}$$

where

a_{Wi} Acceleration of wheel center

Residual acceleration are terms in acceleration expression that are not the derivative of velocity. So the residual acceleration of each wheel is the third term of the above acceleration expression:

$$a_{RWi} = \omega_V \times (V_{Wi} + \dot{r}_{Wi}) \tag{4-5}$$

where

a_{RWi} Residual acceleration of wheel center

4.2.1.4 Coordinate Transformation

Transformation from vehicle reference coordinate system to earth coordinate system is necessary since all variables are described in vehicle reference coordinate system.

The position and orientation of vehicle reference coordinate system against earth coordinate system is related to part of vehicle degrees of freedom: translational vehicle velocities and angular vehicle velocities described in vehicle reference coordinate system.

The derivative of orientation angles roll, pitch and yaw around x,y,z axis are calculated from degree of freedoms, the angular vehicle velocities with respect to vehicle reference coordinate system. So the orientation angles can be calculated simply by integration and plus the initial angles. Then, the overall rotation matrix is a 3by3 matrix calculated from the orientation angles, which is used to calculate the vehicle velocity in the earth coordinate system.

The vehicle position in earth coordinate system can be calculated by integration plus the initial vehicle position respect to earth.

4.2.2 Suspension Subsystem

The Suspension Subsystem shown in Figure 4.4 calculates the suspension kinematics and the suspension forces in the spring, shock absorber (damper), and the stabilizer.

Kinematics is another way of saying suspension geometry (toe, camber, etc.). Compliance is another way of saying suspension stiffness (how far do things bend when road loads come up through the tires). The kinematics and compliance are configured via look up tables, generated by K&C Tester (Kinematics and Compliances Tester) or by a suspension analysis program like ADAMS/Car Suspension Kit.

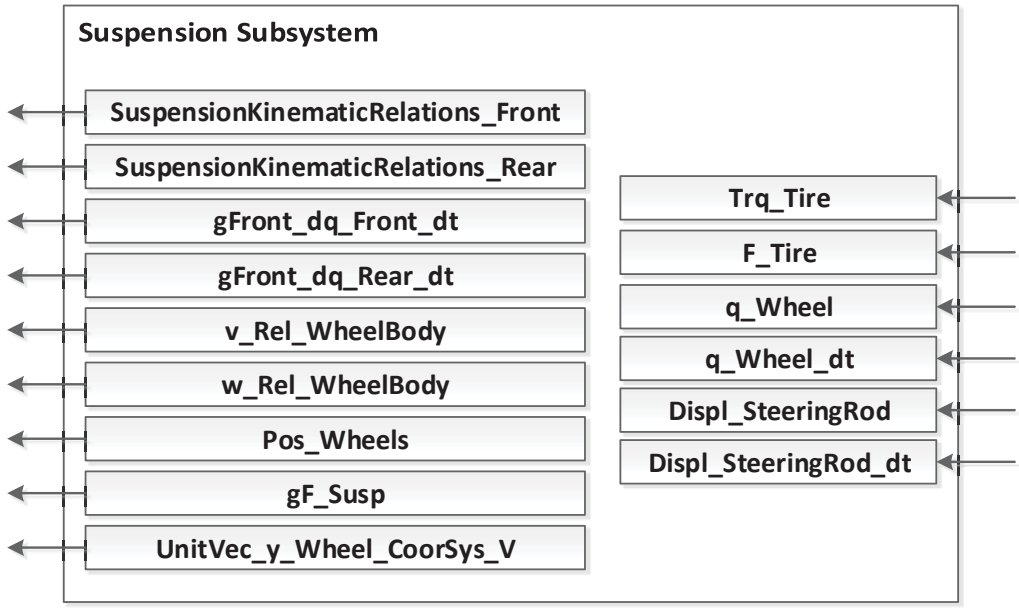

Figure 4.4 **Suspension Subsystem**

4.2.2.1 Rotation Switch

Suspension orientation can be described with three rotation angles which are defined with respect to wheel coordinate system: α (rotation about x-axis), β (rotation about y-axis) and γ (rotation about z-axis). They are defined in ASM with certain relation to camber, caster and toe angles, which is a conventional way to define wheel orientations on a vehicle. A brief introduction to camber, caster and toe angles is given, associated with the definition of three angles of rotation α, β, γ, in ASM.

- Camber angle

Camber angle is an angle between the vertical axis of wheel and vertical axis of vehicle when viewed from front or rear. The definition of camber angle for ASM is illustrated in Figure 4.5. Positive camber is when the distance from upper part of the wheel to the vehicle is greater than the lower part. For the left side of vehicle, positive camber angle nearly equals to the negative α angle about x axis.

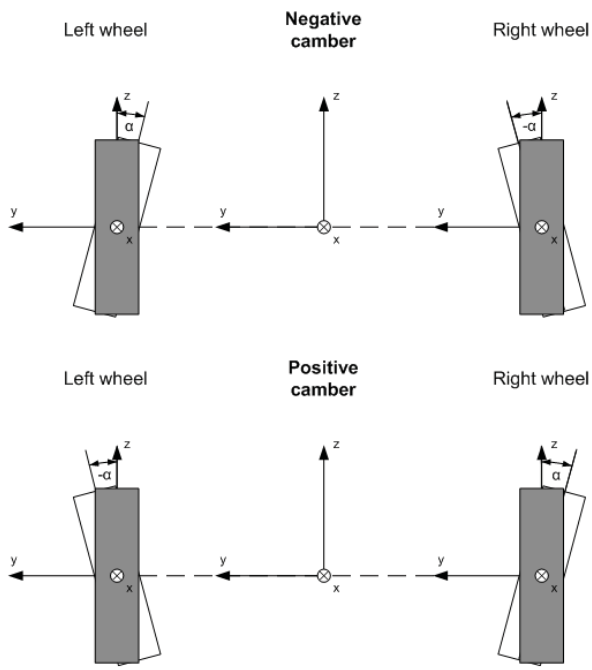

Figure 4.5 Camber Angle

- Caster angle

Caster is an angle to which the steering pivot axis is tilted forward or rearward from vertical, as viewed from left or right side of vehicle. The definition of caster angle is shown in Figure 4.6. Positive caster angle is when the upper pivot line is leaned backward father than bottom pivot. The positive caster angle nearly equals to the negative β about y axis when viewed from each side of vehicle.

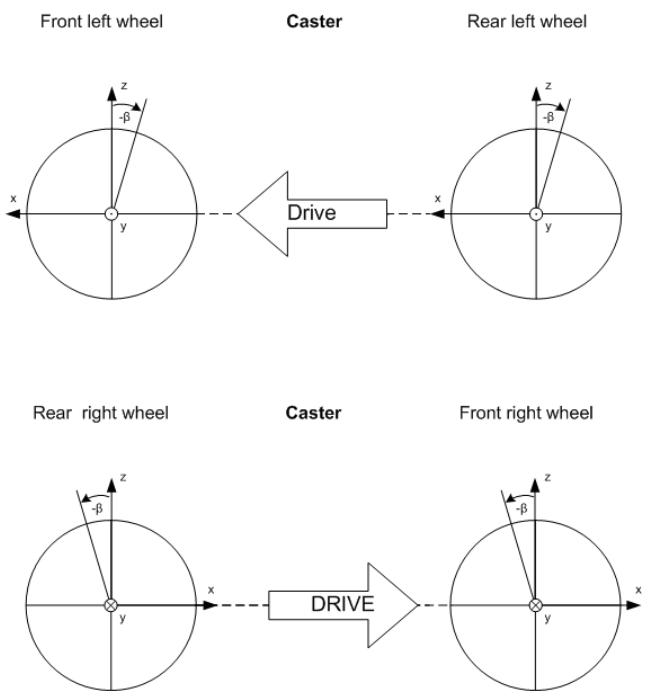

Figure 4.6 Caster Angle

- Toe angle

Toe angle identifies the exact direction the tires are pointed comparing to the longitudinal axis of the vehicle when viewed from top. Figure 4.7 illustrates the toe angle defined in ASM. Positive toe, or toe in, which means the front of the wheel pointing in towards the centerline of the vehicle, is nearly equal to a negative γ for left side of the vehicle. The definition of toe out is the other way round.

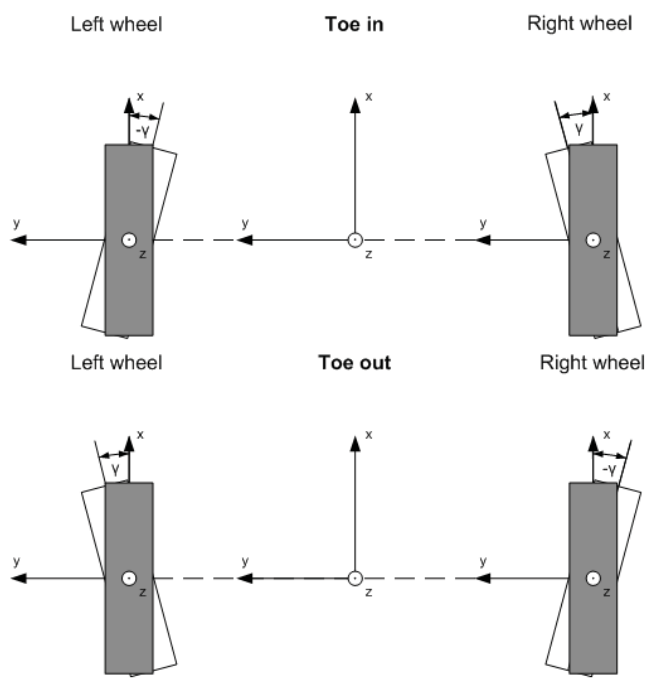

Figure 4.7 Toe Angle

Because a right-handed coordinate system is used in ASM, the left wheel positive toe angle is a negative rotation about the z-axis, the positive camber angle is a negative rotation about the x-axis, and the positive caster angle is a negative rotation about the y-axis. Therefore, the positive camber, caster and toe angle are in accordance with the negative alpha, beta and gamma.

In ASM model two rotation orders are ready for user to choose. The order of the rotation angles plays an important role in parameterizing values such as the suspension kinematics.

- Automotive angle:

  $\gamma - \beta - \alpha$

- Cardan angle:

  Toe – Camber – Caster

4.2.2.2 Front Symmetric Suspension Kinematics

The objective of Suspension Kinematics block is to calculate the relation of wheel position with respect to the displacement of steering rack and vertical wheel displacement of wheel, the relation of wheel angular velocity with respect to the velocity of steering rack vertical wheel speeds of

wheel and the displacements of the spring, damper and stabilizer, according to the steering rack displacement and vertical displacement of wheel.

The suspension kinematics is divided into three parts, Wheel Position, Wheel Orientation, Spring, Damper and Stabilizer Displacements. Inside these three blocks, 2-D look-up tables are used to obtain related kinematics information with two inputs: displacement of steering rack q_{sr} and vertical displacement of left and right front wheel $z_{W_{FL/R}}$.

Wheel position information, including wheel position in xyz directions, and the derivative of wheel position with respect to steering rack displacement and vertical wheel displacement for both of the front right and left wheels, are directly acquired from the Map_Displ_x/y/z_Wheel 2-D look-up tables with the given two inputs, as shown in Figure 4.8.

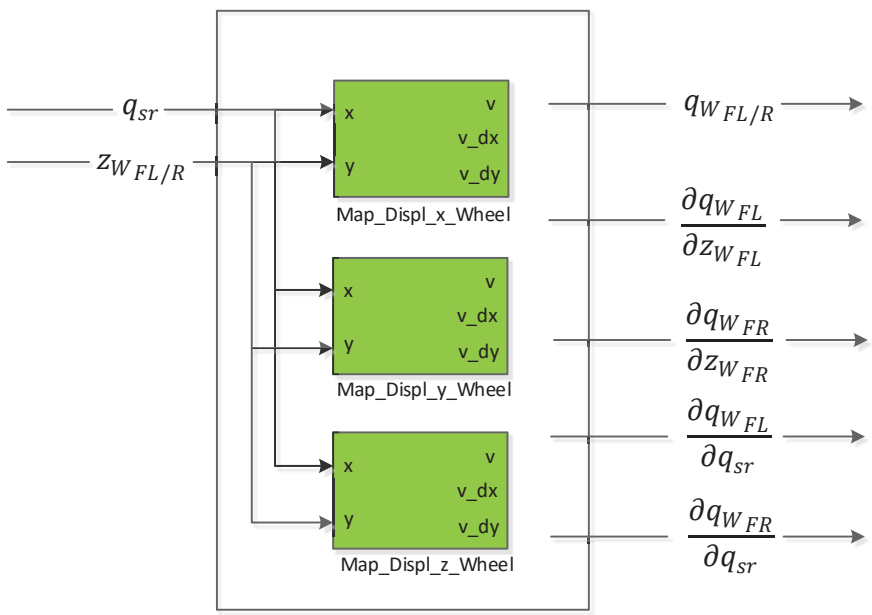

Figure 4.8 Wheel Position

The following equations demonstrate the calculation of front left wheel position, which indicates that the final position of the wheel is the summation of its initial position $q_{W_{FL0}}$ and relative change of the position $\Delta q_{W_{FL}}$ obtained from the look-up tables in Figure 4.8. The front right wheel position $\Delta q_{W_{FR}}$ is calculated in the same way.

$$\Delta q_{W_{FL}} = \Delta q_{W_{FL}}(q_{sr}, z_{W_{FL}}) \qquad (4\text{-}6)$$
$$q_{W_{FL}} = q_{W_{FL0}} + \Delta q_{W_{FL}}$$

where

q_{sr}: Steering rack displacement.

$z_{W_{FL}}$: Vertical displacement of front left wheel.

$\Delta q_{W_{FL}}$[x,y,z]: Relative change of front left wheel position.

$q_{W_{FL0}}$[x,y,z]: Initial position of front left wheel.

$q_{W_{FL}}$[x,y,z]: Front left wheel position.

The wheel orientation information, including wheel angle α, β, γ and changes in orientation with respect to steering rack displacement and vertical wheel movement, are obtained from the output of Map_Angle_Alpha/Beta/Gamma_Wheel 2-D look-up tables with the same inputs, as shown in Figure 4.9.

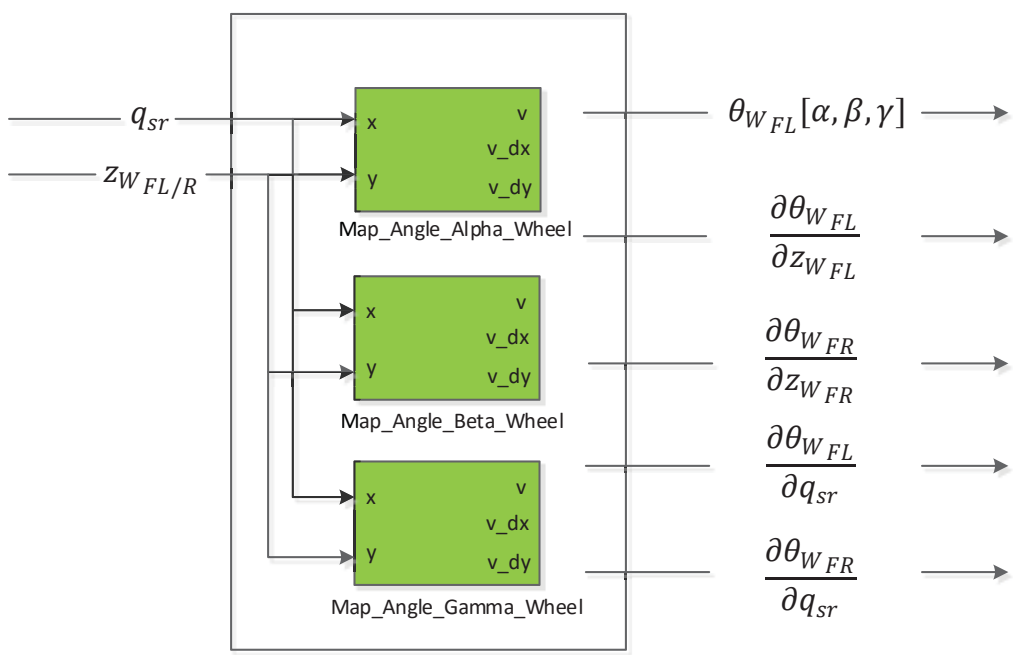

Figure 4.9 Wheel Orientation

The equation for the front left wheel orientation $\theta_{W_{FL}}$ is shown below. And the front right wheel orientation $\theta_{W_{FR}}$ is calculated in the same way.

$$\theta_{W_{FL}}[\alpha, \beta, \gamma] = \theta_{W_{FL}}(q_{sr}, z_{W_{FL}}) \tag{4-7}$$

where

q_{sr}: Steering rack displacement.

$z_{W_{FL}}$: Vertical displacement of front left wheel.

$\theta_{W_{FL}}[\alpha, \beta, \gamma]$: Front left wheel orientation represented by cardan angles.

The vertical displacement of the spring joint points, damper joint points of front left and front right wheel, and vertical displacement of the stabilizer left joint point relative to the right joint point also rely on steering rack displacement and wheel vertical movement. The main dependency is on wheel vertical displacement. Partial differentiation of spring, damper and stabilizer displacement with respect to wheel vertical movement are obtained from Map_Displ_Spring/Damper/Stabilizer 2-D look-up table as shown in Figure 4.10.

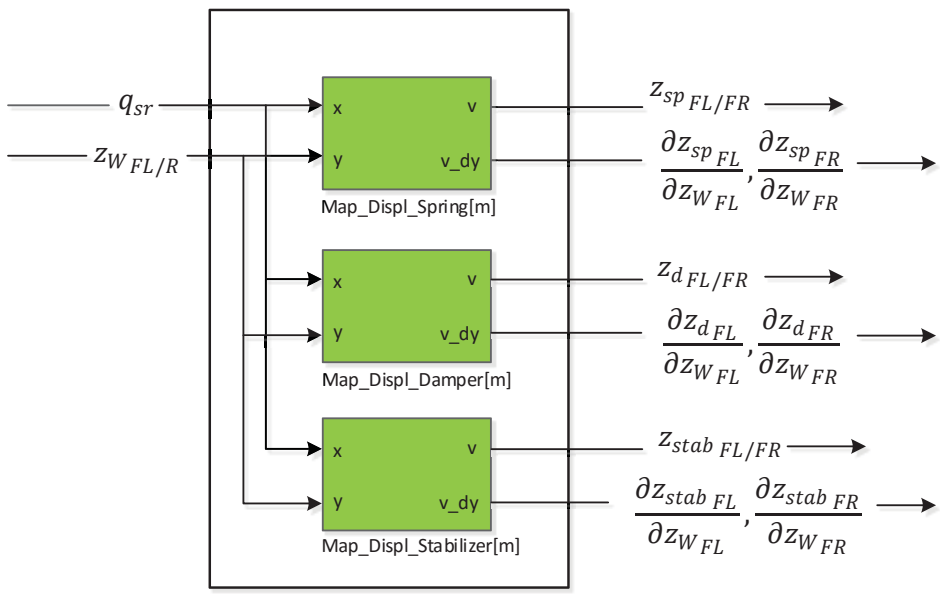

Figure 4.10 Spring & Damper & Stabilizer Displacements

The equations for front left vertical displacements of spring, damper and stabilizer joint points are shown below. And their front right vertical displacements are obtained in the same way.

$$\begin{aligned} z_{sp_{FL}} &= z_{sp_{FL}}(q_{sr}, q_{W_{FL}}) \\ z_{d_{FL}} &= z_{d_{FL}}(q_{sr}, q_{W_{FL}}) \\ z_{stab_{FL}} &= z_{stab_{FL}}(q_{sr}, q_{W_{FL}}) \end{aligned} \tag{4-8}$$

where

$z_{sp_{FL}}$: Vertical displacement of the front left spring joint points.

$z_{d_{FL}}$: Vertical displacement of the front left damper joint points.

$z_{stab_{FL}}$: Vertical displacement of the front left stabilizer joint points.

To calculate the wheel velocity relative to vehicle body and establish the equation of motion in Vehicle Movement block, some partial derivatives must be evaluated, including changes in wheel position and orientation with respect to vertical wheel movement and steering rack displacement, represented as $\frac{\partial q_{W_{FL}}}{\partial z_{W_{FL}}}, \frac{\partial q_{W_{FR}}}{\partial z_{W_{FR}}}, \frac{\partial q_{W_{FL}}}{\partial q_{sr}}, \frac{\partial q_{W_{FR}}}{\partial q_{sr}}, \frac{\partial \theta_{W_{FL}}}{\partial z_{W_{FL}}}, \frac{\partial \theta_{W_{FR}}}{\partial z_{W_{FR}}}, \frac{\partial \theta_{W_{FL}}}{\partial q_{sr}}, \frac{\partial \theta_{W_{FR}}}{\partial q_{sr}}$.

To evaluate the forces acting between the wheel and vehicle body (suspension equivalent force in the direction of wheel center motion), partial derivatives of vertical displacement of spring, damper and stabilizer with respect to vertical wheel movement and steering rack displacement, represented as $\frac{\partial z_{sp_{FL}}}{\partial z_{W_{FL}}}, \frac{\partial z_{sp_{FR}}}{\partial z_{W_{FR}}}, \frac{\partial z_{d_{FL}}}{\partial z_{W_{FL}}}, \frac{\partial z_{d_{FR}}}{\partial z_{W_{FR}}}, \frac{\partial z_{stab_{FL}}}{\partial z_{W_{FL}}}, \frac{\partial z_{stab_{FR}}}{\partial z_{W_{FR}}}$, must be evaluated.

4.2.2.3 Suspension Compliance

Suspension compliance kinematics describes additional elastic wheel displacements caused by the forces and torques acting on the wheel. The additional displacements are considered in x and y directions and α and γ angles. These variables are defined as functions of the longitudinal force, lateral force, longitudinal torque and self-aligning torque.

The additional x,y displacements and α and γ angles caused by tire longitudinal force in x direction denoted as F_x, are obtained from four 2-D look-up tables with longitudinal force as table input, as shown in Figure 4.11. $\Delta q_{W_{Compl_Fx}}$ is used to represent $\Delta x, \Delta y, \Delta\alpha, \Delta\gamma$ caused by F_x applied on left and right tires.

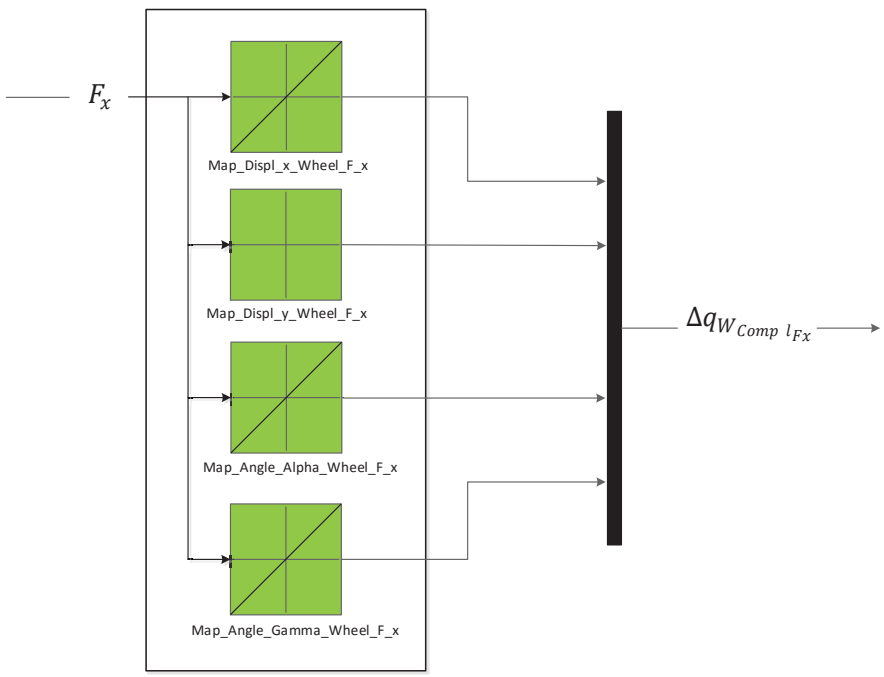

Figure 4.11 Compliance Longitudinal Force

Similarly, the additional x,y displacements and α and γ angles caused by tire lateral force F_y, are obtained from the similar 2-D look-up tables with lateral force as table input, as shown in Figure 4.12. Also, $\Delta q_{W_{Compl_Fy}}$ represents $\Delta x, \Delta y, \Delta\alpha, \Delta\gamma$ caused by F_y.

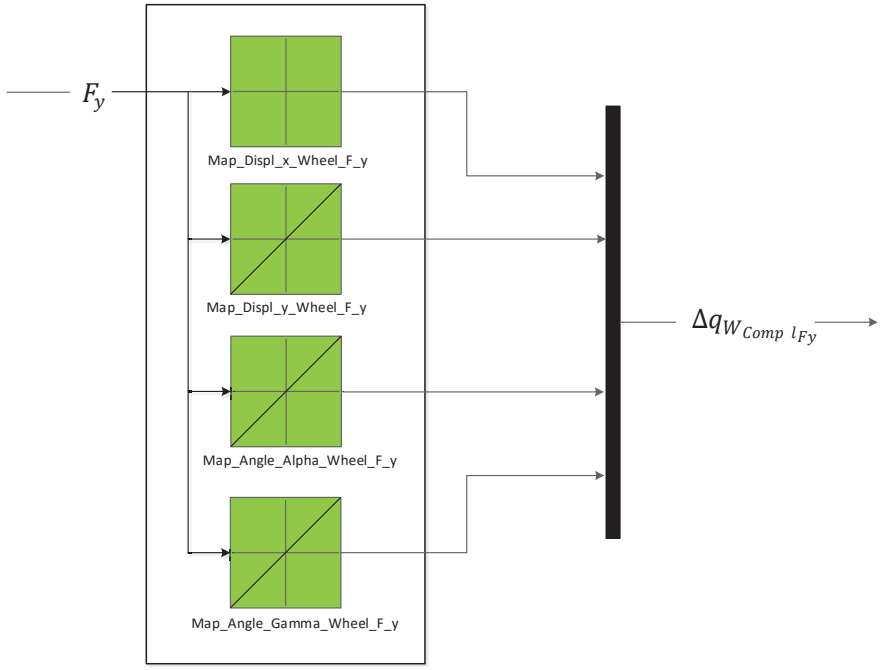

Figure 4.12 Compliance Lateral Force

The influence of the tire torque about longitudinal axis Trq_x on tire displacements and orientations, represented as $\Delta q_{W_{Compl_Trqx}}$, are obtained from corresponding 2-D look-up tables, as shown in Figure 4.13.

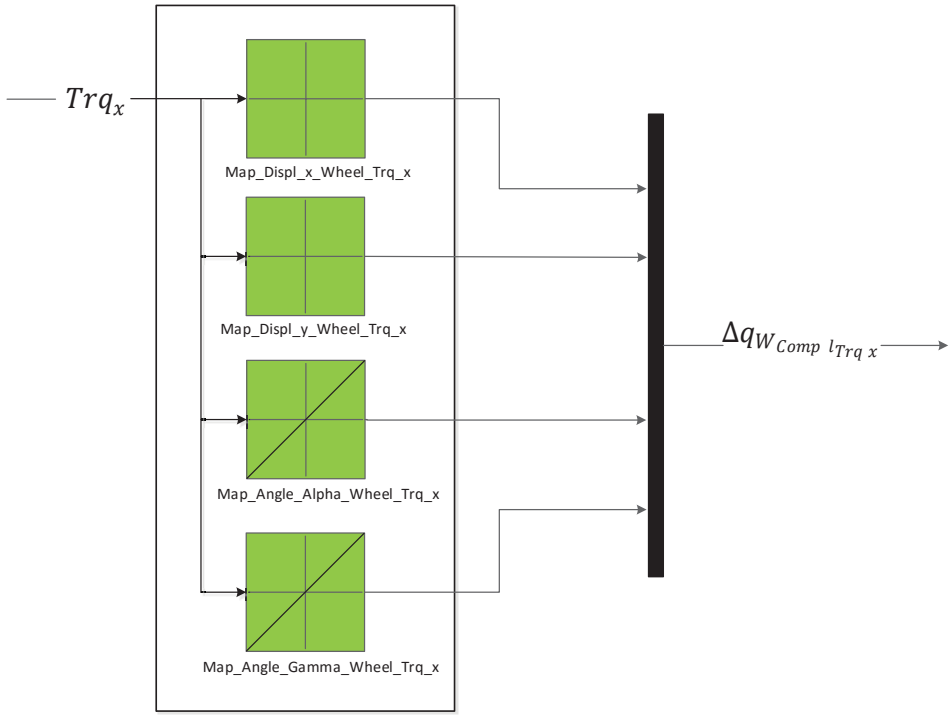

Figure 4.13 Compliance Long Torque

Displacements in x,y direction and α and γ angles caused by aligning torque (Trq_z) about vertical axis, are acquired similarly from the corresponding 2-D look-up tables, as shown in Figure 4.14. $\Delta q_{W_{Compl_Trqz}}$ is used to demonstrate $\Delta x, \Delta y, \Delta\alpha, \Delta\gamma$ caused by Trq_z.

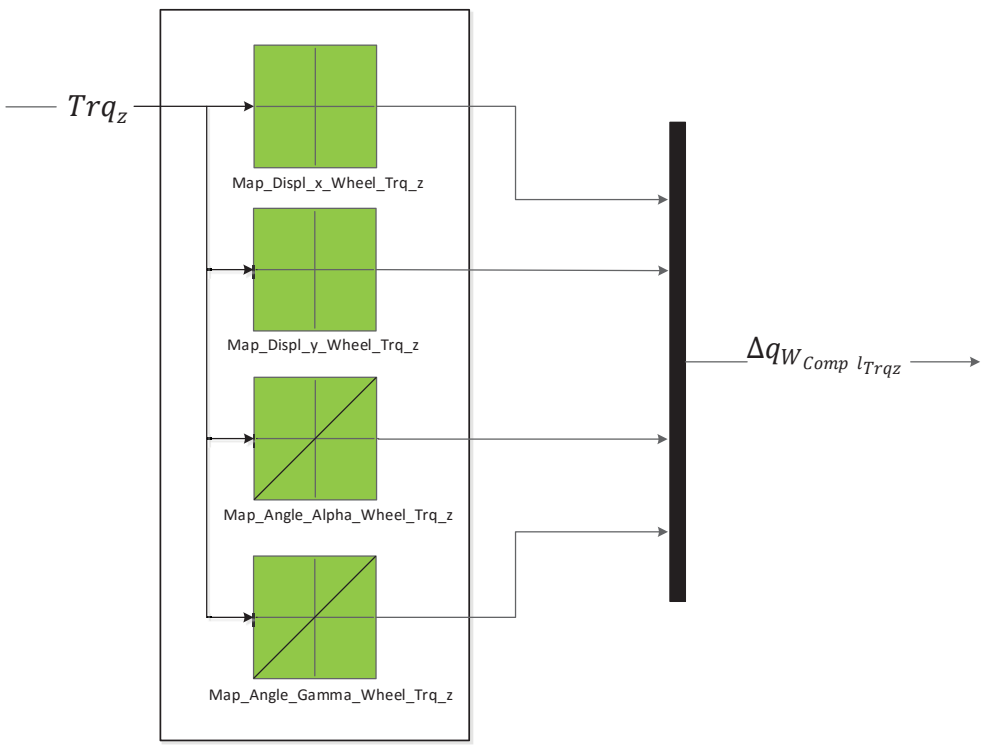

Figure 4.14 Compliance Aligning Torque

According to above model description, it can be seen that the acting torque about longitudinal or vertical axis on wheel has no effect on the displacements of wheel in these two directions.

As demonstrated in the following equation, the total additional displacements and angles due to compliance Δx_{Compl}, Δy_{Compl}, $\Delta \alpha_{Compl}$, $\Delta \gamma_{Compl}$ are the summation of the additional movements of the tire caused by four different forces or torques, F_x, F_y, Trq_x, Trq_z.

$$\Delta q_{W_{Compl_{Fx}}} + \Delta q_{W_{Compl_{Fy}}} + \Delta q_{W_{Compl_{Trqx}}} + \Delta q_{W_{Compl_{Trqz}}} = \begin{bmatrix} \Delta x_{Compl}[Left; Right] \\ \Delta y_{Compl}[Left; Right] \\ \Delta \alpha_{Compl}[Left; Right] \\ \Delta \gamma_{Compl}[Left; Right] \end{bmatrix} \tag{4-9}$$

where

$\Delta q_{W_{Compl_{Fx}}}$: Additional displacement and angles caused by longitudinal force F_x.

$\Delta q_{W_{Compl_{Fy}}}$: Additional displacement and angles caused by lateral force F_y.

$\Delta q_{W_{Compl_{Trqx}}}$: Additional displacement and angles caused by torque about longitudinal

axis Trq_x.

$\Delta q_{W_{Compl_{Trqz}}}$: Additional displacement and angles caused by aligning torque Trq_z.

Δx_{Compl}: Total additional displacement in longitudinal direction due to compliance.

Δy_{Compl}: Total additional displacement in lateral direction due to compliance.

$\Delta \alpha_{Compl}$: Total additional angle around wheel x axis due to compliance.

$\Delta \gamma_{Compl}$: Total additional angle around wheel z axis due to compliance.

4.2.2.4 Relative Velocities

Movement and rotation of wheels are caused by two elements: suspension kinematics and elasticity. Relative translational and angular velocities are calculated from part of the outputs of suspension compliance subsystem and suspension kinematics subsystem without any requirement of other parameters.

Front left wheel relative translational velocity is written as:

$$v_{W_{FL}} = \frac{\partial q_{W_{FL}}}{\partial z_{W_{FL}}} \dot{z}_{W_{FL}} + \frac{\partial q_{W_{FL}}}{\partial z_{W_{FR}}} \dot{z}_{W_{FR}} + \frac{\partial q_{W_{FL}}}{\partial q_{st}} \dot{q}_{st} + \dot{q}_{W_{FL_Compl}} [\, \Delta \dot{x}_{Compl}[Left], \Delta \dot{y}_{Compl}[Left] \,] \tag{4-10}$$

where

$\frac{\partial q_{W_{FL}}}{\partial z_{W_{FL}}}$: Changes in front left wheel position with respect to vertical movement of front left wheel.

$\frac{\partial q_{W_{FL}}}{\partial z_{W_{FR}}}$: Changes in front left wheel position with respect to vertical movement of front right wheel. (equals to zeros for independent suspension)

$\frac{\partial q_{W_{FL}}}{\partial q_{st}}$: Changes in front left wheel position with respect to displacement of the steering rack.

$\dot{z}_{W_{FL}}$: Vertical velocity of front left wheel.

$\dot{z}_{W_{FR}}$: Vertical velocity of front right wheel.

$\dot{q}_{st}$: Velocity of steering rack.

$\dot{q}_{W_{FL_Compl}}$: Translational velocity of front left wheel due to compliance.

$v_{W_{FL}}$: Relative translational velocity of front left wheel.

Front left wheel Relative angular velocity is written as:

$$\omega_{W_{FL}} = \frac{\partial \theta_{W_{FL}}}{\partial z_{W_{FL}}} \dot{z}_{W_{FL}} + \frac{\partial \theta_{W_{FL}}}{\partial z_{W_{FR}}} \dot{z}_{W_{FR}} + \frac{\partial \theta_{W_{FL}}}{\partial q_{st}} \dot{q}_{st} + \dot{\theta}_{W_{FL_Compl}}[\,\Delta\dot{\alpha}_{Compl}[Left], \Delta\dot{\gamma}_{Compl}[Left] \tag{4-11}$$

where

$\frac{\partial \theta_{W_{FL}}}{\partial z_{W_{FL}}}$: Changes in front left wheel orientation with respect to vertical movement of front left wheel.

$\frac{\partial \theta_{W_{FL}}}{\partial z_{W_{FR}}}$: Changes in front left wheel orientation with respect to vertical movement of front left wheel. (equals to zeros for independent suspension)

$\frac{\partial \theta_{W_{FL}}}{\partial q_{st}}$: Changes in front left wheel orientation with respect to steering rack displacement of wheel.

$\dot{\theta}_{W_{FL_Compl}}$: Angular velocity of front left wheel due to compliance.

$\omega_{W_{FL}}$: Relative angular velocity of front left wheel.

Calculations of front right wheel relative velocities are the same. Obviously, velocities of wheel body relative to vehicle body consist of four parts; change of wheel velocity with respect to vertical wheel velocity for both front wheels, change of wheel velocity with respect to velocity of steering rack and additional wheel velocity due to suspension compliance.

4.2.2.5 Relative Position

Wheel relative position is the combination of wheel movement due to suspension kinematics and elasticity. It is calculated by making the summation of the wheel position from Suspension Kinematics Subsystem and the wheel position change due to suspension compliance from Suspension Compliance Subsystem.

$$\begin{bmatrix} q_{W_{FL}} \\ q_{W_{FR}} \end{bmatrix} + \begin{bmatrix} q_{W_{FL_Compl}} \\ q_{W_{FR_Compl}} \end{bmatrix} = q_W[FL; FR][x; y; z] \tag{4-12}$$

where

$q_{W_{FL}}$: Front left wheel position.

$q_{W_{FR}}$: Front right wheel position.

$q_{W_{FL_Compl}}$: Additional front left wheel displacement due to suspension compliance.

$q_{W_{FR_Compl}}$: Additional front right wheel displacement due to suspension compliance.

$q_W[x; y; z]$: Front wheels position including compliance effects.

4.2.2.6 Front Suspension Forces

Forces acting between the wheel and vehicle body, are generated by spring, damper, and stabilizer. These forces are functions of spring, damper and stabilizer displacements which are represented by 2-D look-up tables. The equations and corresponding look-up tables with respect to spring, damper and stabilizer forces are described as follows.

$$\begin{aligned} F_{sp_{FL}} &= F_{sp_{FL}}(z_{sp_{FL}}) \\ F_{d_{FL}} &= F_{d_{FL}}(\dot{z}_{d_{FL}}) \\ z_{stab} &= z_{stab_{FL}} - z_{stab_{FR}} \\ F_{stab} &= F_{stab}(z_{stab}) \end{aligned} \tag{4-13}$$

where

$z_{sp_{FL}}$: Vertical displacement of the front left spring joint point.

$F_{sp_{FL}}$: Force of front left spring.

$\dot{z}_{d_{FL}}$: Vertical speed of displacement of the front left damper joint point.

$F_{d_{FL}}$: Force of front left damper.

$z_{stab_{FL}}$: Vertical displacement of the front stabilizer left joint point.

$z_{stab_{FR}}$: Vertical displacement of the front stabilizer right joint point.

z_{stab}: Vertical displacement of the stabilizer left joint point relative to the right joint point.

F_{stab}: Force of stabilizer.

As shown in Figure 4.15, the function $F_{sp_{FL}}$ is represented as a look-up table Map_SpringForceLeft. A look-up table Map_DamperForceLeft is used to represent the function of $F_{d_{FL}}$. The spring and damper force on the right front side are calculated similarly as the left side by using look-up tables.

The stabilizer force F_{stab} is obtained from the look-up table Map_StabilizerForce, with vertical displacement of the stabilizer left joint point relative to the right joint point as the table input.

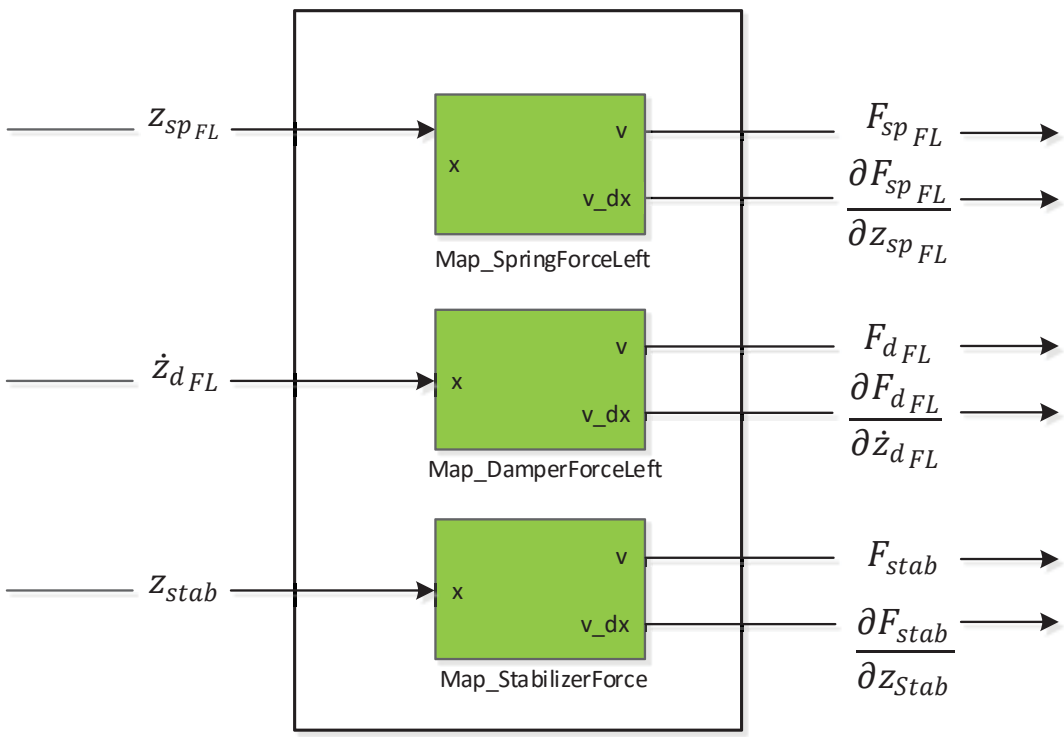

Figure 4.15 Front Spring Forces

4.2.2.7 Generalized Front Suspension Forces

The suspension forces calculated above must be transformed in the direction of the relevant degrees of freedom.

A Jacobian matrix for suspension spring deflection with respect to front wheel vertical displacement is demonstrated as:

$$J_{Sp} = \frac{\partial z_{Sp}}{\partial z_{W_Front}} = \begin{bmatrix} \frac{\partial z_{sp_{FL}}}{\partial z_{W_{FL}}} & \frac{\partial z_{sp_{FL}}}{\partial z_{W_{FR}}} \\ \frac{\partial z_{sp_{FR}}}{\partial z_{W_{FL}}} & \frac{\partial z_{sp_{FR}}}{\partial z_{W_{FR}}} \end{bmatrix} \tag{4-14}$$

where

J_{Sp}: Jacobian matrix for suspension spring deflection with respect to front wheel vertical movement.

$z_{W_Front}[z_{W_{FL}}, z_{W_{FR}}]$: Front wheel vertical movement including vertical movement of front left and front right wheel.

According to spring forces

$$F_{Sp} = \begin{bmatrix} F_{sp_{FL}}(z_{sp_{FL}}) \\ F_{sp_{FR}}(z_{sp_{FR}}) \end{bmatrix} \tag{4-15}$$

where

F_{Sp}: Spring forces in terms of front spring displacement.

The generalized spring forces acting on front left wheel and front right wheel are calculated as follows:

$$[gF_{Sp}]_{2\times1}(FL; FR) = {J_{Sp}}^T F_{Sp} \tag{4-16}$$

where

$[gF_{Sp}]_{2\times1}(FL; FR)$: Generalized spring forces.

Similar calculations are carried out for the damper and stabilizer.

The Jacobian matrix for suspension damper and damper forces are calculated as follow:

$$J_D = \begin{bmatrix} \frac{\partial z_{d_{FL}}}{\partial z_{W_{FL}}} & \frac{\partial z_{d_{FL}}}{\partial z_{W_{FR}}} \\ \frac{\partial z_{d_{FR}}}{\partial z_{W_{FL}}} & \frac{\partial z_{d_{FR}}}{\partial z_{W_{FR}}} \end{bmatrix}, F_D = \begin{bmatrix} F_{d_{FL}}(\dot{z}_{d_{FL}}) \\ F_{d_{FR}}(\dot{z}_{d_{FL}}) \end{bmatrix} \tag{4-17}$$

where

J_D: Jacobian matrix for suspension damper displacement with respect to front wheel

vertical movement.

F_D: Damper forces in terms of front damper velocity.

So, the generalized damper forces applying on front left wheel and front right wheel respectively are calculated as follows:

$$[gF_D]_{2\times1}(FL;FR) = J_D{}^T F_D \tag{4-18}$$

where

$[gF_D]_{2\times1}(FL;FR)$: Generalized damper forces.

The Jacobian matrix for suspension stabilizer and stabilizer forces are shown below:

$$J_{Stab} = \begin{bmatrix} \frac{\partial z_{Stab}}{\partial z_{W_{FL}}} \\ \frac{\partial z_{Stab}}{\partial z_{W_{FR}}} \end{bmatrix}, \quad F_{Stab} = F_{stab}(z_{stab}) \tag{4-19}$$

where

J_{Stab}: Jacobian matrix for suspension stabilizer displacement with respect to front wheel vertical movement.

F_{Stab}: Stabilizer forces in terms of front stabilizer displacement.

So, the generalized stabilizer forces acting on front left wheel and front right wheel respectively are calculated as follows:

$$[gF_{Stab}]_{2\times1}(FL;FR) = J_{Stab}F_{Stab} \tag{4-20}$$

where

$[gF_{Stab}]_{2\times1}(FL;FR)$: Generalized stabilizer forces.

The generalized suspension forces with respect to front left and right wheel are the summation of three forces, spring forces, damper forces and stabilizer forces.

$$\left[gF_{z_{W_{Front}}}\right]_{2\times1}(FL;FR) = [gF_{Sp}]_{2\times1} + [gF_D]_{2\times1} + [gF_{Stab}]_{2\times1} \tag{4-21}$$

where

$[gF_z_{W_{Front}}]_{2\times1}(FL;FR)$: Generalized suspension forces in direction of vertical displacement of left and right wheel.

To stabilize the equation of motion, the derivatives of the generalized forces with respect to the generalized speeds are required. It can be calculated using the Jacobian matrices and the derivatives of the spring, damper and stabilizer forces with respect to spring displacement, damper velocity, and stabilizer displacement, respectively.

4.2.3 Steering Subsystem

Figure 4.16 illustrates the block diagram of the Steering Subsystem. The Steering Subsystem transfers steering wheel angle applied by a driver to displacement of steering rack, which is then passed on to suspension kinematics to decide the movement of wheels including the position and orientation. The Steering Subsystem and its interaction with other subsystems such as Vehicle Movement and Suspension Subsystem are illustrated in Figure 4.2.

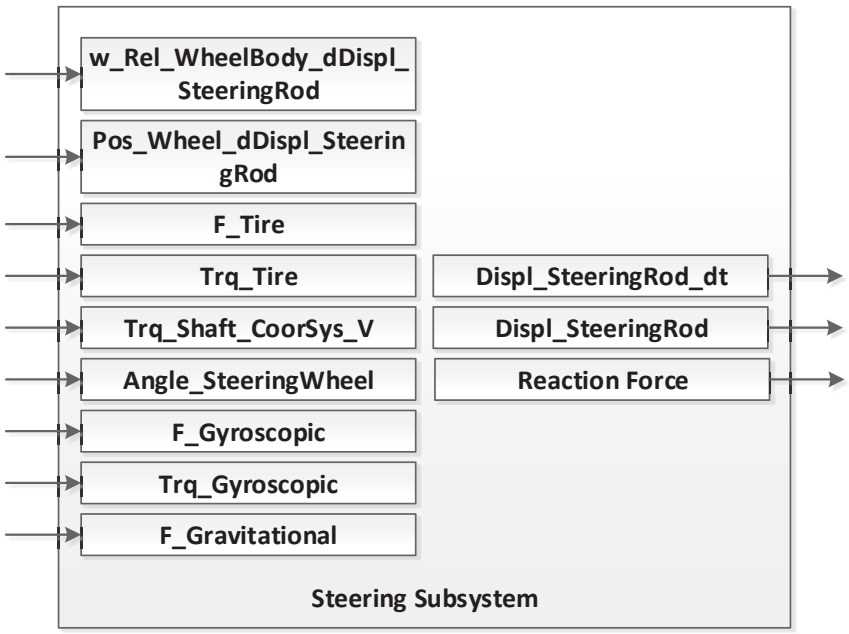

Figure 4.16 Block diagram of Steering Subsystem

There are two different inputs to the steering system: either driver angle or driver torque. If driver angle is used as an input, a steering torque is calculated at the steering column from relative angle of driver steering input and pinion gear angle. If driver torque is used as input, it is considered as the torque that directly acts on the steering rack. In-depth analysis of the Steering Subsystem in ASM Vehicle Dynamics Model is discussed below.

4.2.3.1 Force on Steering Rack

The connection between steering wheel and rack and pinion system is simulated as a torsion bar, which is an assembly of a spring and damping. So the steering torque at steering column is written as:

$$T_{St_Col} = k_{St}\left(\delta_{St_{Wheel}} - \delta_{St_{Gear}}\right) + d_{St}\left(\dot{\delta}_{St_{Wheel}} - \dot{\delta}_{St_{Gear}}\right) \qquad (4\text{-}22)$$

where

k_{st}: Stiffness of the steering column [Nm/rad]

d_{St}: Damping coefficient of the steering column [Nm/rad/s]

$\delta_{St_{Wheel}}$: Angle of steering wheel [rad]

$\delta_{St_{Gear}}$: Angle of pinion [rad]

$\dot{\delta}_{St_{Wheel}}$: Angular velocity of the steering wheel [rad/s]

$\dot{\delta}_{St_{Gear}}$: Angular velocity of the pinion [rad/s]

T_{St_Col}: Torque on the steering column [Nm]

Angle of the pinion gear is related to two parameters. First one is the radian that the gear turns for per meter the steering rack goes, described as i_{st} [rad/m]. Second one is the displacement of the steering rack. Therefore, when turning the steering wheel, the angle of the gear derives from multiplication of these two parameters.

$$\delta_{St_{Gear}} = i_{st} * S_{StR} \qquad (4\text{-}23)$$

where

i_{St}: Radian that the pinion gear turns for per meter the steering rack goes [rad/m]

S_{StR}: Displacement of the steering rack [m]

Force on Steering rack is contributed to the torque at steering column due to driver input and torque exerted from Electric Power Steering device. The model simulates the power steering by a first order time delay dynamics equation related to the time constant τ_{St_Pow} and a function of steering column torque, as illustrated in Figure 4.17 The function of steering torque at column is represented by a look-up table.

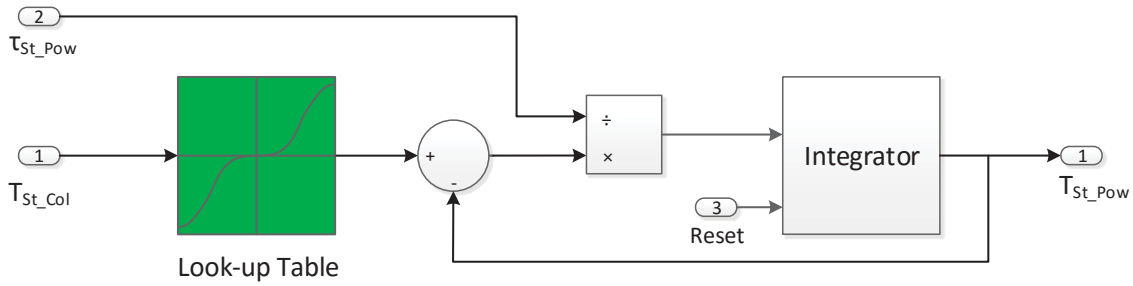

Figure 4.17 Electric Power Steering Torque

The whole equation for simulating electric power steering torque is written as:

$$\tau_{St_Pow}\dot{T}_{St_Pow}=f(T_{St_Col})-T_{St_Pow} \tag{4-24}$$

where

τ_{St_Pow}: Power steering time constant

T_{St_Col}: Torque on the steering column [Nm]

T_{St_Pow}: Power steering torque [Nm]

With all of the variables needed, generalized force on steering rack is calculated as:

$$(T_{St_Col} + T_{St_Pow}) \times i_{St} + F_{StR_External} = F_{St} \tag{4-25}$$

where

F_{St}: Active steering force on steering rack [N]

$F_{StR_External}$: Additional external force on steering rack [N]

4.2.3.2 Generalized Force and Torques

Except for F_{St}, the steering rack is also applied with a reaction force from tire, which will prevent the steering of vehicle. In order to calculate this reaction force, the information of forces and torques on tire is necessary.

The summation of front tire torques consists of three parts: tire torque from Tire Subsystem, gyroscopic torque of the wheel from Vehicle Movement Subsystem and driving torque on the wheel from Tire Subsystem. The summation of front tire forces consists of tire force from Tire Subsystem, gyroscopic force and gravitational force of the wheel from Vehicle Movement Subsystem.

Therefore, the generalized reaction force applied on steering rack is written as:

$$\theta_{FL_{dPos_{St}}} \cdot T_{FL} + \theta_{FR_{dPos_{St}}} \cdot T_{FR} + S_{FL_{dPos_{St}}} \cdot F_{FL} + S_{FR_dPos_St} \cdot F_{FR} = gF_{Tire_St} \tag{4-26}$$

where

$[\theta_{FL_dPos_St}; \theta_{FR_dPos_St}]$: Change of front wheel orientation with respect to the displacement of steering rack [x;y;z]

$[S_{FL_dPos_St}; S_{FR_dPos_St}]$: Change of front wheel position with respect to the displacement of steering rack [x;y;z]

$[T_{FL}; T_{FR}]$: Front tire torque vector in vehicle reference system [x;y;z]

$[F_{FL}; F_{FR}]$: Front tire force vector in vehicle reference system [x;y;z]

gF_{Tire_St}: Generalized reaction force from tire

The generalized force on steering rack, a necessary parameter to estimate the dynamics of steering rack, is calculated by the summation of active steering force F_{St} and reactive force gF_{Tire_St}. These two forces are in opposite direction.

$$F_{St} + gF_{Tire_St} = gF_{St} \tag{4-27}$$

where

gF_{St}: Generalized force on steering rack

4.2.3.3 Steering Mass

Generalized mass in the direction of steering rack displacement depends on the wheel inertias and masses. The equation is written as follows:

$$\theta_{FL_{dPos_{St}}} \cdot I_{FL} \cdot \theta_{FL_{dPos_{St}}} + \theta_{FR_{dPos_{St}}} \cdot I_{FR} \cdot \theta_{FR_{dPos_{St}}} + m_{FL} \cdot S_{FL_{dPos_{St}}} \cdot S_{FL_{dPos_{St}}} + m_{FR} \cdot S_{FR_{dPos_{St}}} \cdot S_{FR_{dPos_{St}}} + m_{StR} + m_{StR_ex} = m_{St} \tag{4-28}$$

where

I_{FL}: Front left wheel inertia.

I_{FR}: Front right wheel inertia.

m_{FL}: Front left wheel mass.

m_{FR}: Front right wheel mass.

m_{StR}: Mass of the steering rack.

m_{StR_ex}: Additional external steering rack mass.

m_{St}: Generalized steering rack mass.

4.2.3.4 Displacement of Steering rack

In order to calculate the displacement of steering rack, several parameters are necessary, including generalized force and torque in steering, mass of steering and friction. The equation of displacement of steering rack is written as follows:

$$(gF_{St} - F_{damping_T} - F_{fric})/(m_{St} + m_{Stab}) = S_{StR_dt2} \qquad (4\text{-}29)$$

where

gF_{St}: Generalized force on steering rack

$F_{damping_T}$: Damping force when steering torque is the input of a steering system

F_{fric}: Friction force

m_{St}: Generalized steering rack mass

m_{Stab}: Mass of stabilization

S_{StR_dt2}: Acceleration of steering rack

The displacement of steering rack is calculated by integrating S_{StR_dt2} for twice in a row. What is worth to pay attention during the integration is that the upper and lower limit of the steering rack displacement, which should be set up based on physical steering system via a saturation block.

4.2.4 Tire Subsystem

ASM Vehicle Dynamics model contains two variants tire models for user to choose, including EasyToUse Tire Model developed by Prof. G. Rill, Germany and Magic Formula Tire Model by Prof. Pacejka, Netherlands [16].

5 MODEL INTEGRATION AND OFFLINE SIMULATION OF INTEGRATED MODEL

The dSPACE ASM Vehicle Dynamics Model is integrated with a Nexteer Steering Model to provide an entire mechanical model with main components, including tire, suspension, steering, aerodynamics and brake models. The Integrated Mechanical Model serves as an important part in the plant model for the calculation of vehicle speed, handwheel torque and motor coupling torque, as shown in Figure 5.1. It takes in signals, including motor angle and motor speed from the EPS Master Model, which can be divided into dSPACE ASM Electric Components, Sensors and Actuators Simulation and CAN Bus.

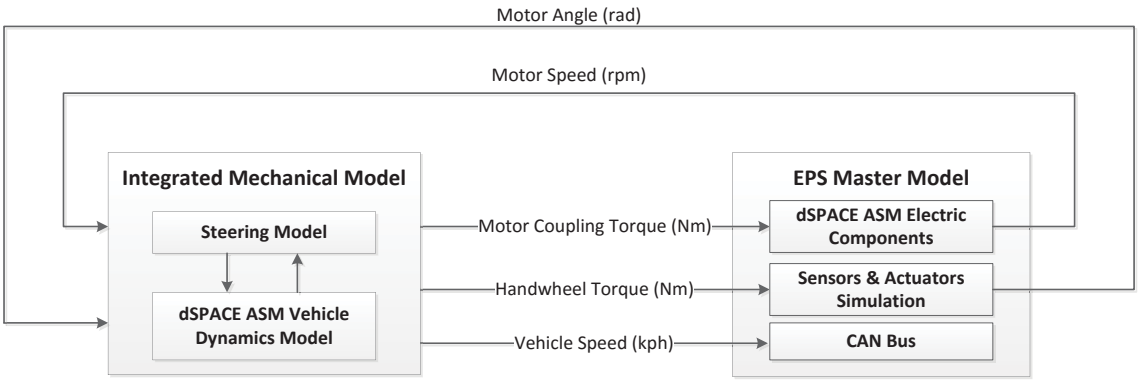

Figure 5.1 Integrated Mechanical Model Input/Output

There are two reasons for the integration of dSPACE ASM Vehicle Dynamics Model and the Nexteer Steering Model. The first is to use a Nexteer Steering Model with real-car tested parameters makes the model performance closer to a real steering system. Second, the dSPACE Vehicle Dynamics Model can simulate the reaction force applied on the steering rack from road. Steering reaction force has a significant influence on vehicle maneuverability. An accurate estimation of reaction force is important to evaluate the vehicle steering behavior and test EPS ECU functions.

5.1 The Integration of ASM Model with Steering Model

In general, the dSPACE Vehicle Dynamics Model will estimate the reaction force, the feedback force from road applied on steering rack. In addition, the vehicle speed will be calculated by ASM

model. The Steering Model calculates the displacement of steering rack by using the reaction force calculated by dSPACE ASM Vehicle Dynamics model (see equation 4-26). The block diagram of the Integrated Mechanical Model is shown in Figure 5.2, which provides basic information about these two models and the interface between them.

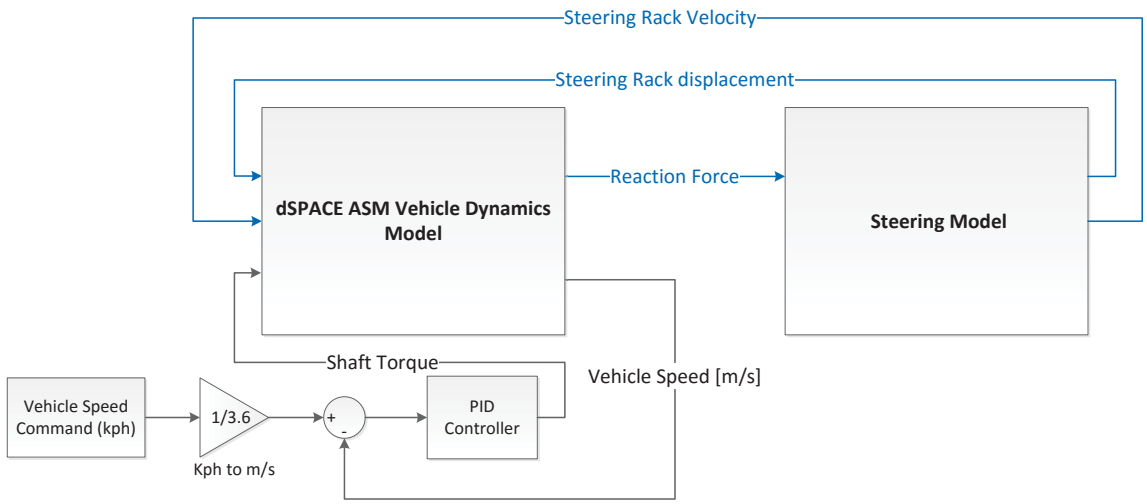

Figure 5.2 The Block Diagram of Integrated Mechanical Model

As seen from Figure 3.2, three sources of torques are applied on the steering rack: Driver Torque from steering wheel, Motor Assist Torque from power steering electric motor and Reaction force from rod. Therefore, as shown in Figure 5.2, the reaction force obtained from dSPACE Vehicle Dynamics Model will be imported into the Steering Model and converted to torque, together with driver torque and motor torque to calculate the displacement and velocity of the steering rack, which will be sent back to the ASM Model. The detailed calculation is represented in equation 5-1, 5-2, 5-3, 5-4 as follows.

The following equation shows the conversion from reaction force to reaction torque on steering rack.

$$T_{Tire_st} = gF_{Tire_St} * r_{pinion} \tag{5-1}$$

where

gF_{Tire_St}: Generalized reaction force from tire

r_{pinion}: Radius of the pinion in rack and pinion steering system

T_{Tire_st}: Reaction torque on steering rack

The following equation shows the calculation of acceleration of steering rack.

$$\frac{T_{HW_st} + T_{MotCoup} + T_{Tire_st}}{J_{Rack}} = S_{StR_dt2} \tag{5-2}$$

where

T_{HW_st}: Hanswheel torque for steering

$T_{MotCoup}$: Motor coupling torque

J_{Rack}: Moment inertia of steering rack

S_{StR_dt2}: Acceleration of steering rack

The following equations show the calculation of angular velocity and rotation angle of steering rack.

$$\int S_{StR_dt2} = \omega_{StR}$$
$$\int \omega_{StR} = \theta_{StR} \tag{5-3}$$

where

ω_{StR}: Steering rack angular velocity

θ_{StR}: Steering rack rotation angle

The following equation shows the conversion of rack angle in radians and rack velocity in radians/s to rack displacement in mm and rack displacement velocity in mm/s.

$$q_{Rk} = \theta_{\mathrm{Rk}} \times \frac{\mathrm{K}_{Rk}}{2\pi}$$
$$q_{Rk}dt = \omega_{\mathrm{Rk}} \times \frac{\mathrm{K}_{Rk}}{2\pi} \tag{5-4}$$

where

q_{Rk}: rack displacement in mm

$q_{Rk}dt$: rack displacement velocity in mm/s

K_{Rk}: a factor which represents how far the steering rack goes for one revolution of the pinion (mm/rev)

In addition, a PID controller is used to control the vehicle speed to track vehicle speed command. The steering behaviors of vehicle for various driving maneuvers are analyzed at a stable and constant vehicle speed.

The key of the model integration lies in the interface between the two models, specifically, the input and output signals between them, which are illustrated in Figure 5.3Figure 5.3. The integration of dSPACE ASM Vehicle Dynamics Model and Nexteer Steering Model is to replace the blocks of calculating displacement and velocity of the steering rack in dSPACE Vehicle Dynamics Model Steering Subsystem with the Nexteer Steering Model.

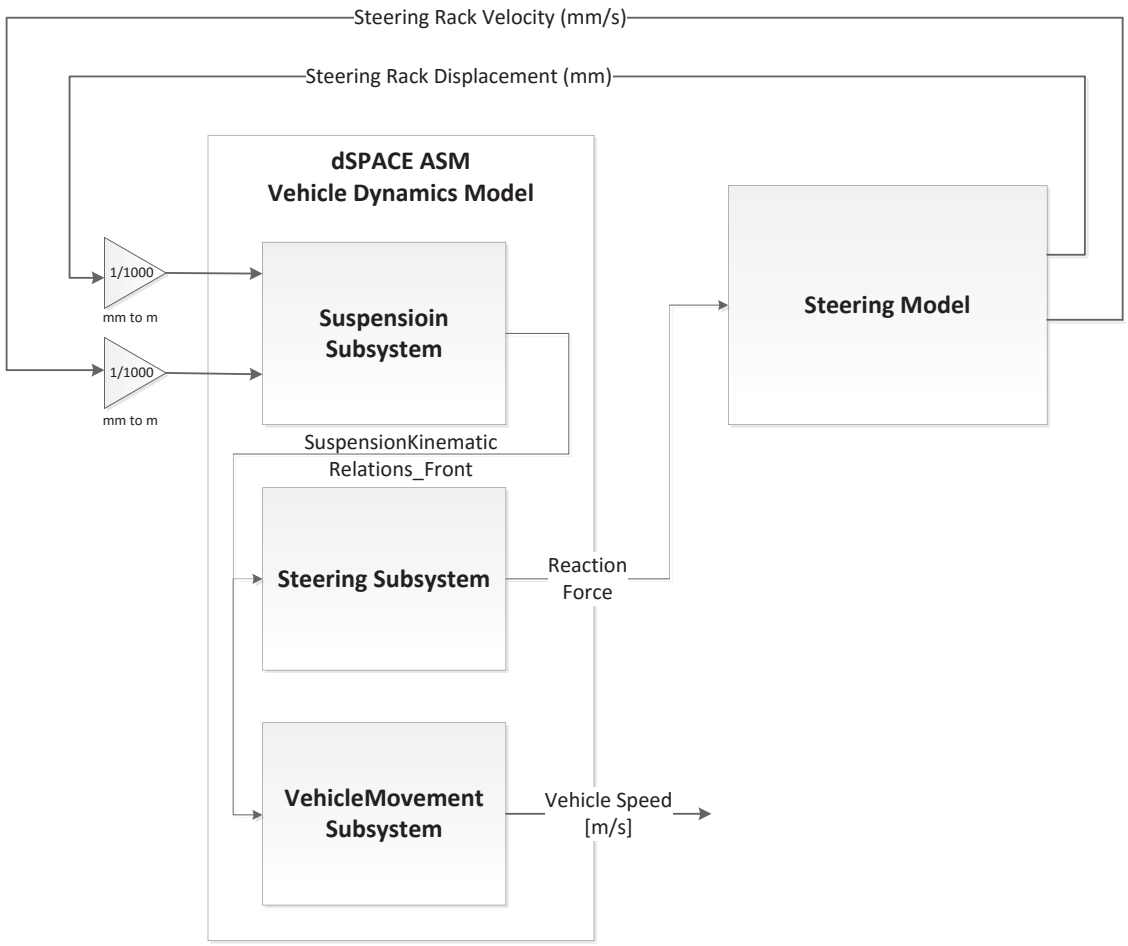

Figure 5.3 Model Integration Input/Output in ASM model

As shown in Figure 5.3, there is a Steering Subsystem in the vehicle dynamics model. The reason that Nexteer Steering Model is used is because that model could better simulate the characteristic

of steering system designed by Nexteer. Therefore, part of the ASM steering subsystem is kept with other subsystems in dSPACE Vehicle Dynamics Model to simulate the reaction force and calculate vehicle speed. And the Steering Model will output steering rack velocity and displacement.

Parameters between the interfaces of the two models to be integrated must be unified. The steering rack displacement and velocity from Steering Model are measured as millimeters, so they must be converted to meters to be compatible with the same variables to be used in the dSPACE Model.

The displacement of the steering rack, calculated by the Steering Model, is served as an input to Suspension Subsystem in dSPACE ASM Vehicle Dynamics model to evaluate the movement and rotation of wheels. The signal 'SuspensionKinematicRelations_Front' from Suspension Subsystem to Steering Subsystem is a combination of two signals, change of wheel position and wheel orientation with respect to the displacement of steering rack, which shows the movement and rotation of wheels due to the movement of steering rack, as discussed in 4.2.2.2. Then, the Steering Subsystem in ASM model is able to estimate the reaction force applied on the steering rack, which is represented as gF_{Tire_St} in 4.2.3.2. The specific calculation of gF_{Tire_St} is described in equation 4-26. Also the equation of motion is established in Vehicle Movement Subsystem to calculate the vehicle speed in equation 4-2.

5.2 Offline Validation Results of the Integrated Mechanical Model

The offline testing of the Integrated Mechanical Model is necessary before using it for real-time simulation. The test procedures will be briefly introduced and detailed maneuver signal design will be demonstrated. Also, test results for various driving maneuvers are analyzed.

5.2.1 Maneuver Signal Design

Three driving maneuvers are used to test the model performance, including chirp steering, ramp steering and single lane change. These three steering input signals are designed using Simulink or in combination with Matlab script [17]. In the future testing, the vehicle behavior is examined under constant speed, so developing steering input signals which could be generated with certain period of time at given time is also an objective for the design process. Detailed design and signals being developed are discussed below.

- Chirp Signal

A chirp is a sinusoidal signal whose frequency linearly varies from f1 to f2 within a time period of T. It is used to represent the input of steering wheel angle. Figure 5.4 shows a Matlab Simulink model to generate a chirp signal.

The following equation represents the generation of a chirp signal.

$$F_{chirp}(t) = \sin\left(\frac{1}{2} \cdot \frac{2\pi(f_2 - f_1)}{T} \cdot t^2 + 2\pi f_1 t \right) \tag{5-5}$$

where

f_1: Initial frequency

f_2: Final frequency

T: Time period for the chirp signal

t: Time

As shown in Figure 5.4, the rising edge of a step input is used to initiate the generation of a chirp signal. The chirp signal generation continues when the "Reset" input keeps high and the generation stops when the "Reset" input changes to zero. Blocks circled in Figure 5.4(b) are the original Chirp block in Simulink. The rest are additional blocks I added.

In future real-time simulation, the step input could be replaced by a constant block, so the model is able to generate a chirp signal whenever changing the constant value from zero to one in CDNG in real-time.

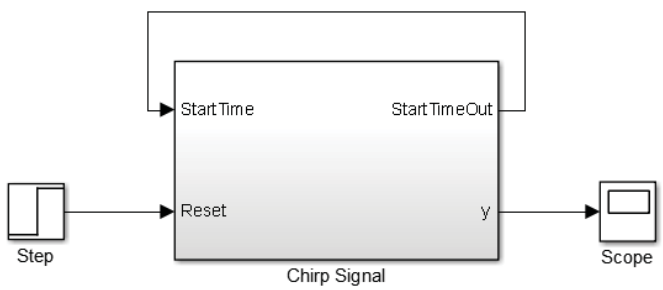

Figure 5.4 (a) Chirp Signal Subsystem in Simulink

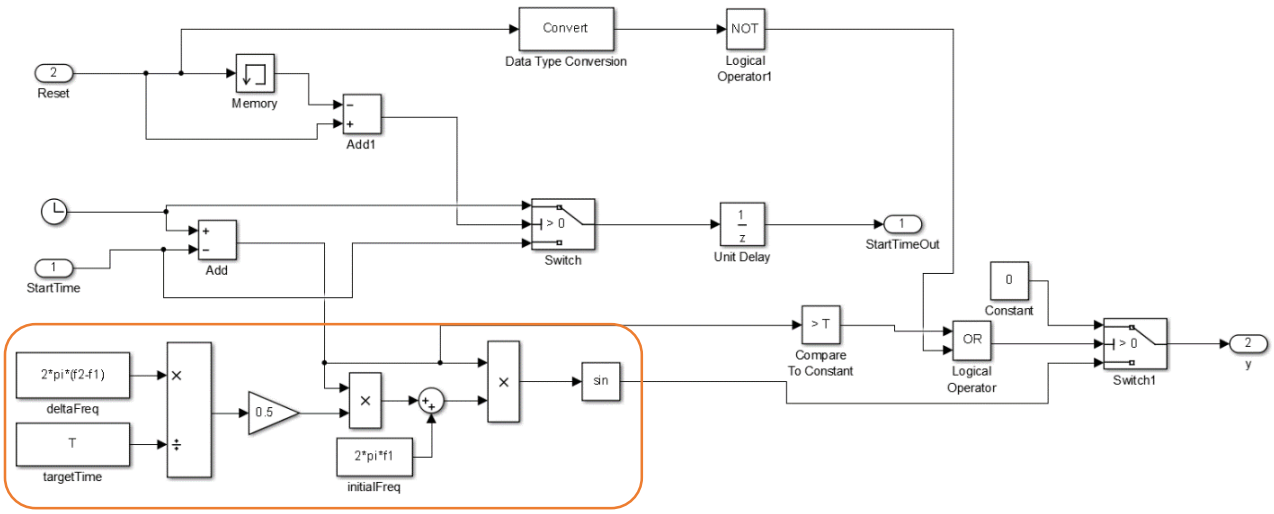

Figure 5.4 (b) Chirp Signal Design in Simulink

Figure 5.5 shows a chirp signal generated by the presented model. The frequency of the signal goes from 1 to 10Hz within 10 seconds.

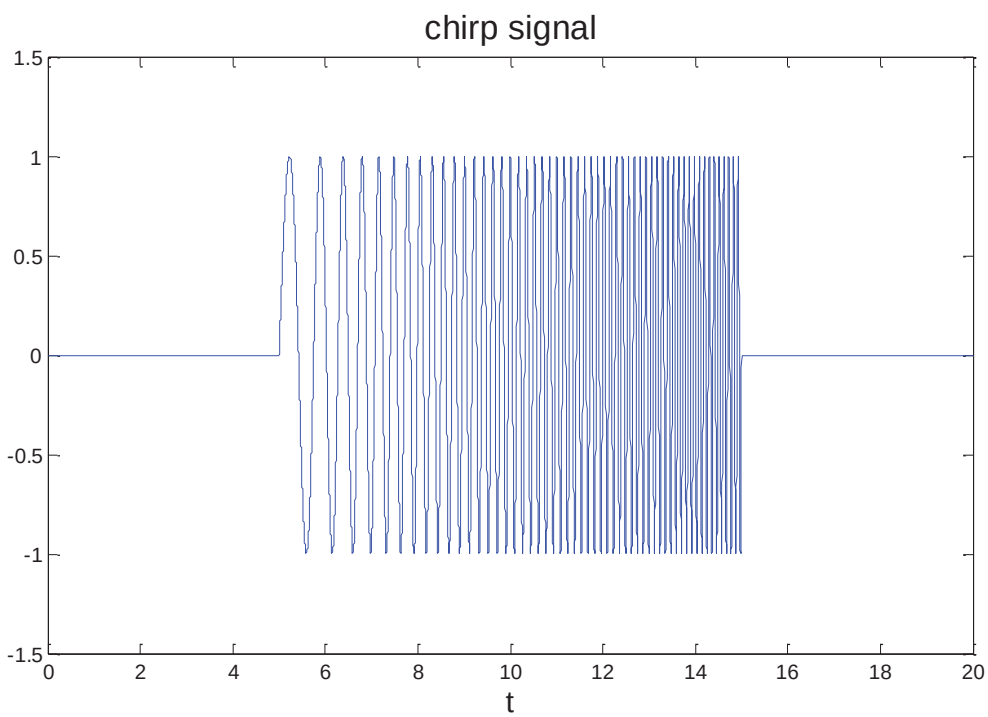

Figure 5.5 Generated Chirp Signal

- Ramp Signal

A ramp signal is used as an input signal of steering wheel angle, which will make a vehicle moving with a driving path as shown in Figure 5.6. This signal is named as 'ramp signal' because the vehicle driving path caused by this steering input looks like a series of mathematical ramp functions.

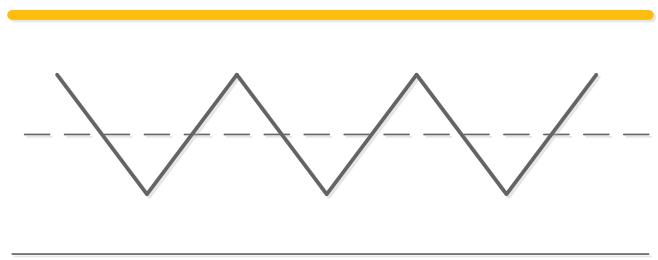

Figure 5.6 Driving Path of a Ramp Steering Input

A ramp signal is designed by using Repeating Sequence block in Simulink, which repeats sequence of numbers specified in a table of time-value pairs. The following equation shows the calculation in Repeating Sequence block.

$$F_{Ramp}(t) = f_{Ramp}([t - t_{start}] - n \cdot T) \qquad (5\text{-}6)$$

where

t_{start}: Start time of the function to be repeated

n: Positive integer (0,1,2…n)

T: Time period for the repeating

t: Time

$f_{Ramp}(\,)$: Ramp signal function

Figure 5.7 illustrates ramp signal model in Simulink. The circled blocks are in the Repeating Sequence block which represents equation (5-6). In equation (5-6), $f_{Ramp}(\,)$ stands for the function in Look-Up Table in Figure 5.7. For generation of a ramp signal, $f_{Ramp}(\,)$ is a single sinusoidal waveform, which forms a ramp signal by repeating process. The rest blocks in Figure 5.7 are developed to generate repeated sine waves controlled by a rising edge of a step signal.

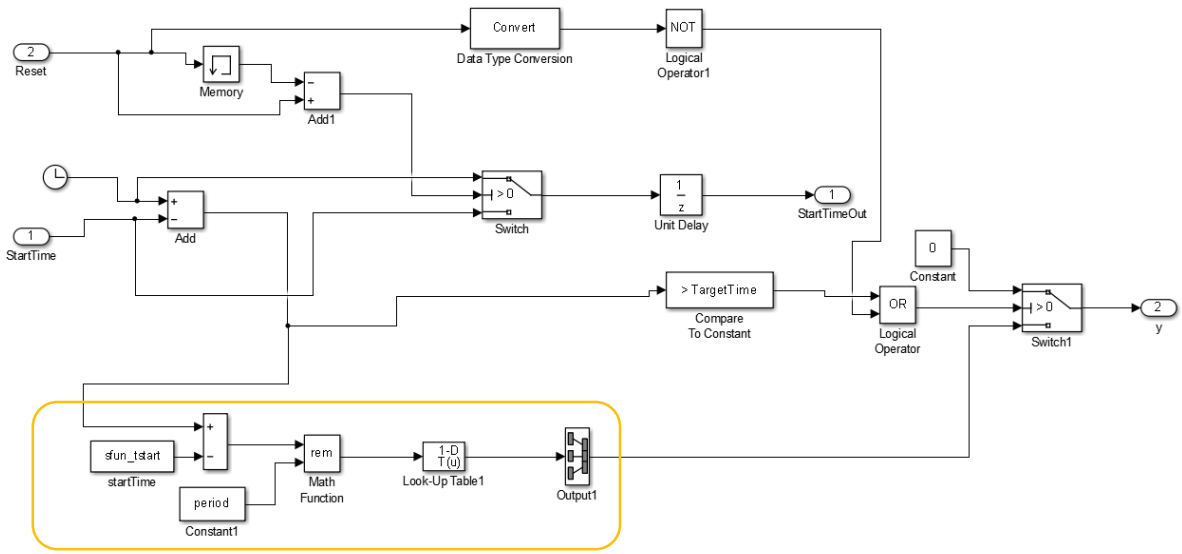

Figure 5.7 Ramp Signal Model in Simulink

The function in Look-Up Table is generated by breakpoints and table data set. Equation (5-7) shows the breakpoints t_{HW} and output data θ_{HW_ramp} in ramp signal generation.

$$\theta_{HW_ramp} = \sin(2\pi f \cdot t_{HW}) \tag{5-7}$$

where

f: Frequency of a single sinusoidal waveform

t_{HW}: Time vector from 0 to the period of the single sinusoidal waveform

θ_{HW_ramp}: Output handwheel angle values

Figure 5.8 shows a ramp signal which has six repeating 0.2Hz sinusoidal waveform ($f = 0.2$) within 30 seconds generated by the model in Figure 5.7.

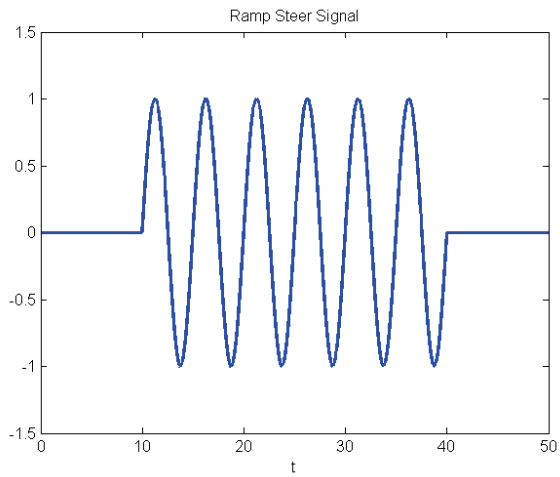

Figure 5.8 Generated Ramp Signal

- Single Lane Change Signal

Single Lane Change is changing back and forth on two lanes which is usually for passing a car in real life. So it is a typical testing method to examine the performance of a steering system. Figure 5.9 shows the driving path of single lane change.

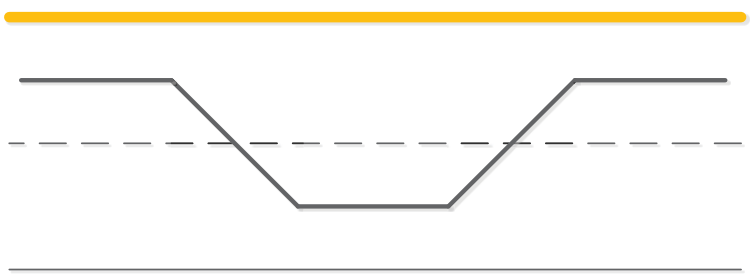

Figure 5.9 Driving Path of Single Lane Change

Similar to the generation of ramp signal, single lane change signal is also generated by Repeating Sequence block in Simulink. The following equation shows the calculation in Repeating Sequence block.

$$F_{Single}(t) = f_{Single}([t - t_{start}] - n \cdot T) \qquad (5\text{-}7)$$

where

t_{start}: Start time of the function to be repeated

n: Positive integer (0,1,2…n)

T: Time period for the repeating

t: Time

$f_{Single}(\,)$: Single lane change function

The Simulink model for generating single lane change signal is the same as the model for ramp signal, as shown in Figure 5.7. $f_{Single}(\,)$ in equation (5-7) is the single lane change function in Look-Up Table.

The calculation of break point HwAngle_time and output values HwAngle_output in the Look-Up Table for single lane change is referred to Figure A in Appendix A.

Figure 5.10 shows the generated single lane change waveform with 0.2Hz sinusoidal wave and period of 10 seconds. The single lane change signal is the continuous repeating of the single lane changes waveform which represents the steering angle input by driver when doing single lane change.

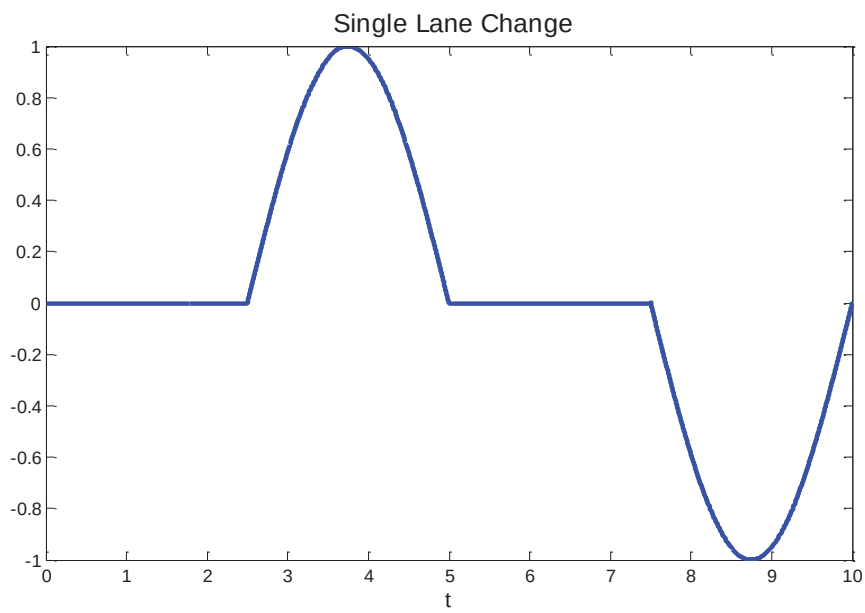

Figure 5.10 Single Lane Change Waveform

5.2.2 Ramp Steer Test

Ramp Steer Test is used to examine vehicle steering performance under sine maneuver input at maintained vehicle speed. In this test, a low frequency of 0.2Hz sinusoidal signal is used as a steering input to investigate the vehicle behavior at 20kph, 60kph and 100kph, respectively. For

each constant vehicle speed, parameters in PID controller need to be adjusted to make vehicle reach the command speed quickly and stably.

In the following analysis, two kinds of plots are generated for each constant vehicle speed to evaluate the behavior of the steering system. The first is in time domain, which consists of five subplots, including vehicle speed, steering angle, steering torque, motor speed, lateral acceleration and yaw rate. The second plot is a cross plot which consists of four subplots, steering torque, lateral acceleration and yaw rate with respect to steering angle, and steering torque with respect to lateral acceleration.

With the spread of power steering systems, there are increasing demands for not only vehicle performance, but also comfortable steering feeling. Evaluation of lateral acceleration and yaw rate are carried out to describe vehicle performance under steering maneuvers. The steering torque is considered as an indication of driver's steering feeling, which is an important factor of driving handling.

- Sinusoidal Signal: 0.2 Hz

  Vehicle Speed: 20kph

  Handwheel Angle gain: 150

  PID controller: KP: 5.5 KI: 0.05

Figure 5.11 and Figure 5.12 illustrate the steering behavior of the Integrated Mechanical Model with 0.2Hz steering input at maintained speed of 20kph. The maximum value of the simulated handwheel angle is referring to the real vehicle test. It is clear that all the variables in time plots follow the 0.2Hz sine pattern at constant speed 20kph. The cross plot shows a generally linear response between handwheel angle and handwheel torque, lateral acceleration, yaw rate.

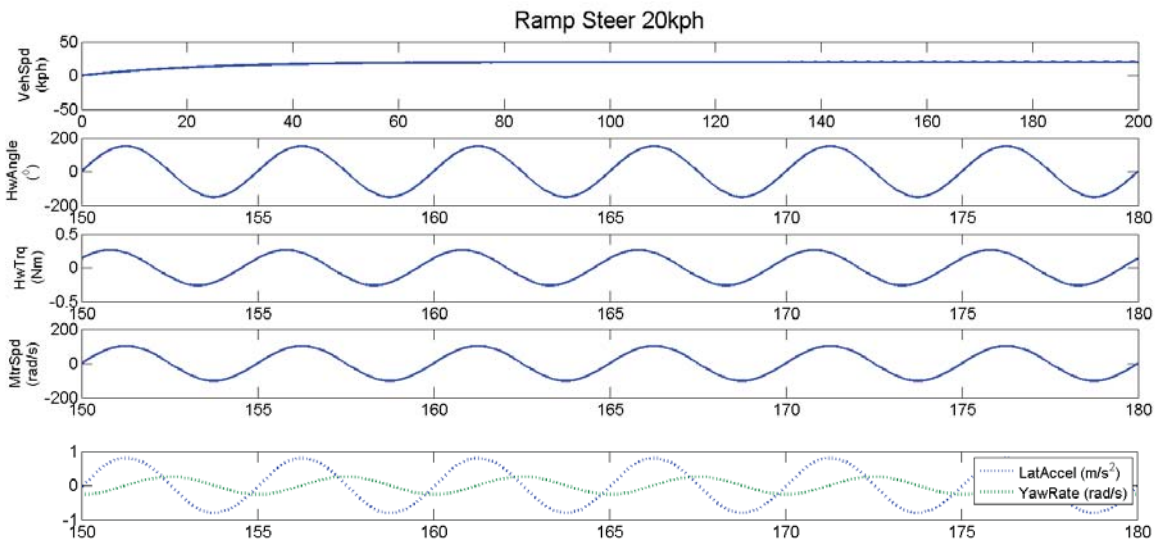

Figure 5.11 Ramp Steer Time Plots@20kph

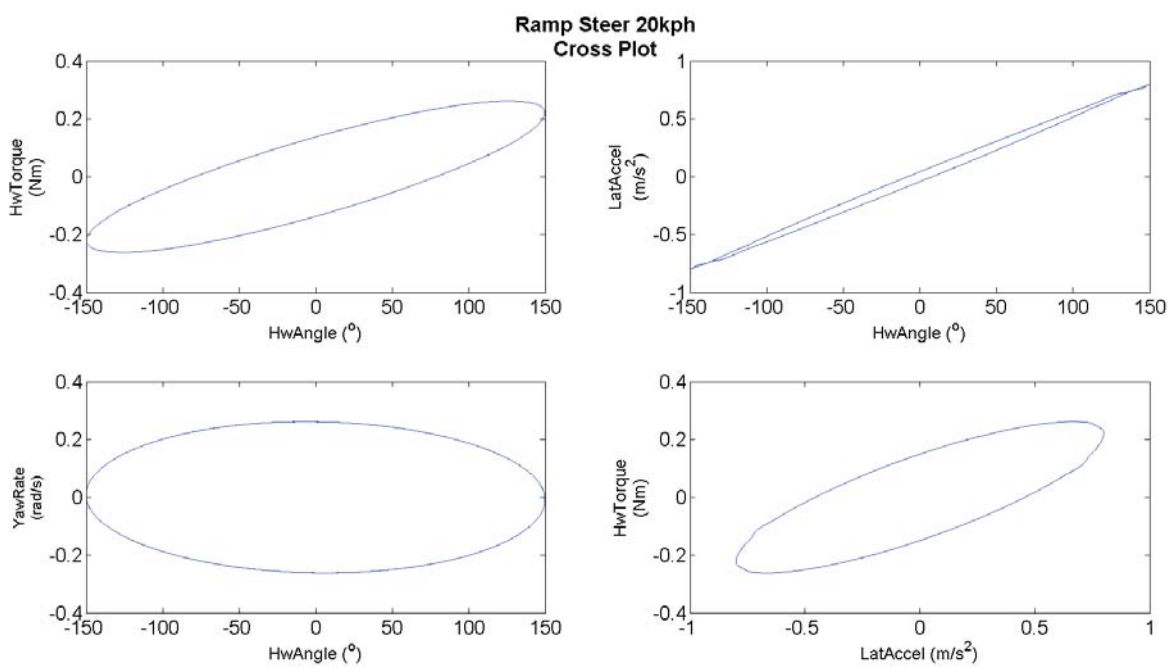

Figure 5.12 Ramp Steer Cross Plot @20kph

- Sinusoidal Signal: 0.2 Hz

  Vehicle Speed: 60kph

  Handwheel Angle gain: 40

  PID controller: KP: 3 KI: 0.3

Figure 5.13 and Figure 5.14 illustrate the test results for 0.2Hz sine maneuver at maintained speed 60kph. All variables are following the period of 0.2Hz sinusoidal input signal. However, there is obvious fluctuation on lateral acceleration and handwheel torque, which can be seen at related cross plot as well. Complicated steering at high vehicle speeds inevitably lead to change of lateral acceleration, which will have an impact on reaction force applied on steering rack and definitely influence driver's feeling represented by steering torque.

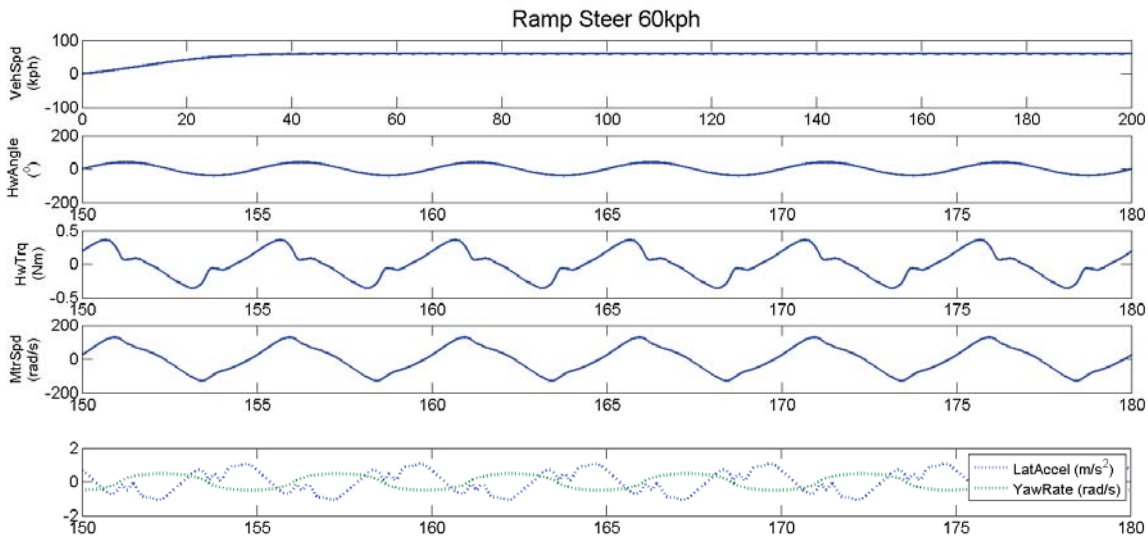

Figure 5.13 Ramp Steer Time Plots @60kph

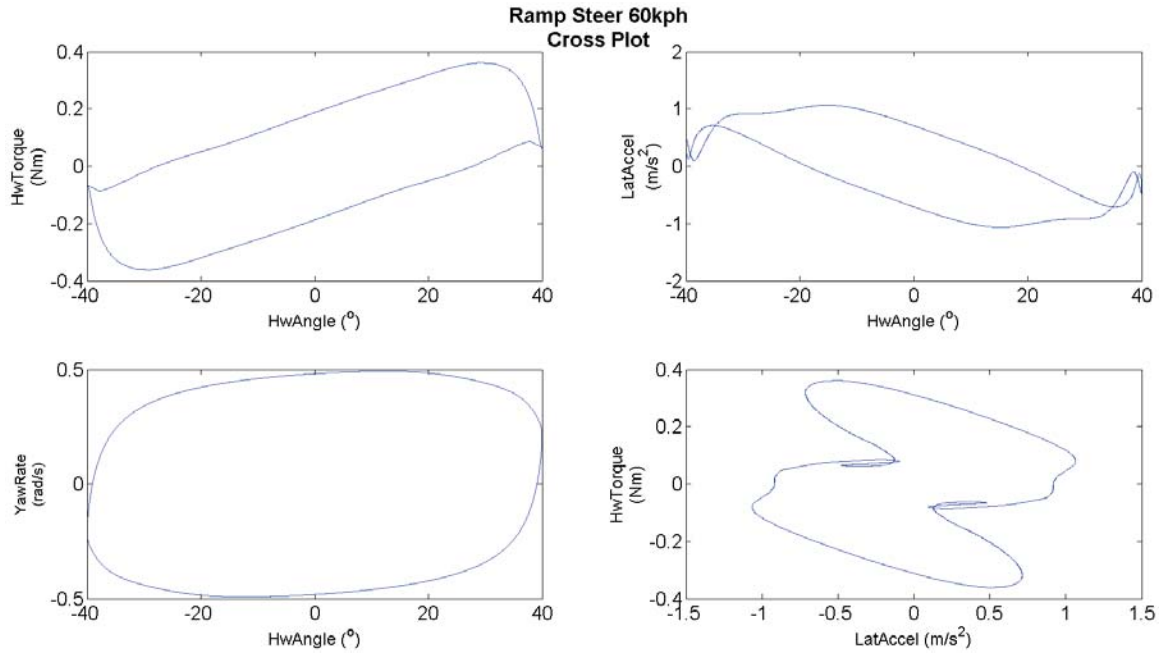

Figure 5.14 Ramp Steer Cross Plot @60kph

- Sinusoidal Signal: 0.2 Hz

  Vehicle Speed: 100kph

  Handwheel Angle gain: 20

  PID controller: KP: 3 KI: 0.3

Figure 5.15 and Figure 5.16 show 0.2Hz ramp steer test results at 100kph. It is obvious that the lateral acceleration increases as the increase of vehicle speed. The similar fluctuation in handwheel torque and lateral acceleration happens on vehicle at 100kph. The vehicle behaves more unstable at high speed and easy to be off balance.

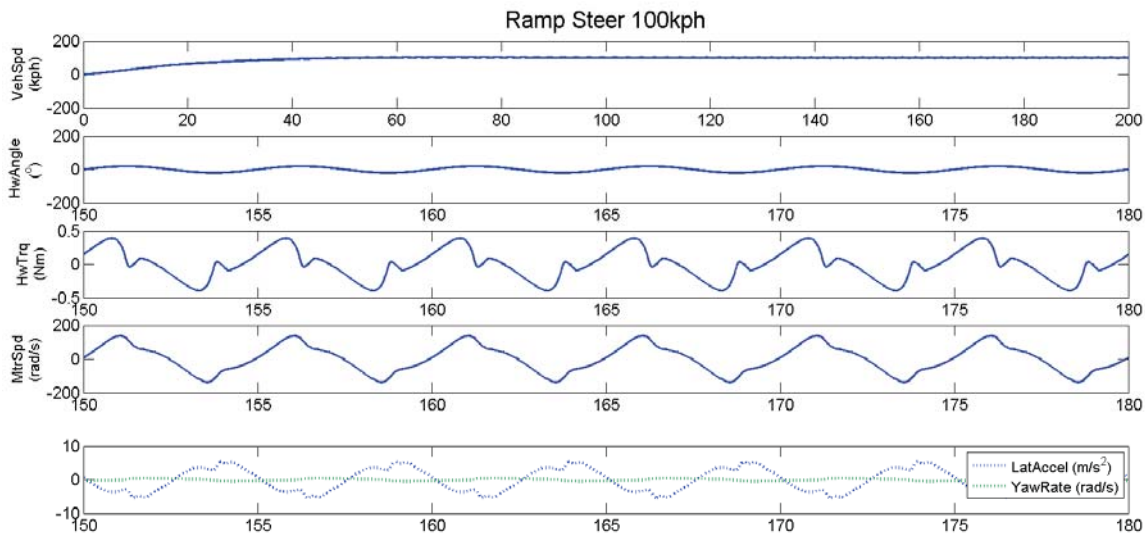

Figure 5.15 Ramp Steer Time Plots @100kph

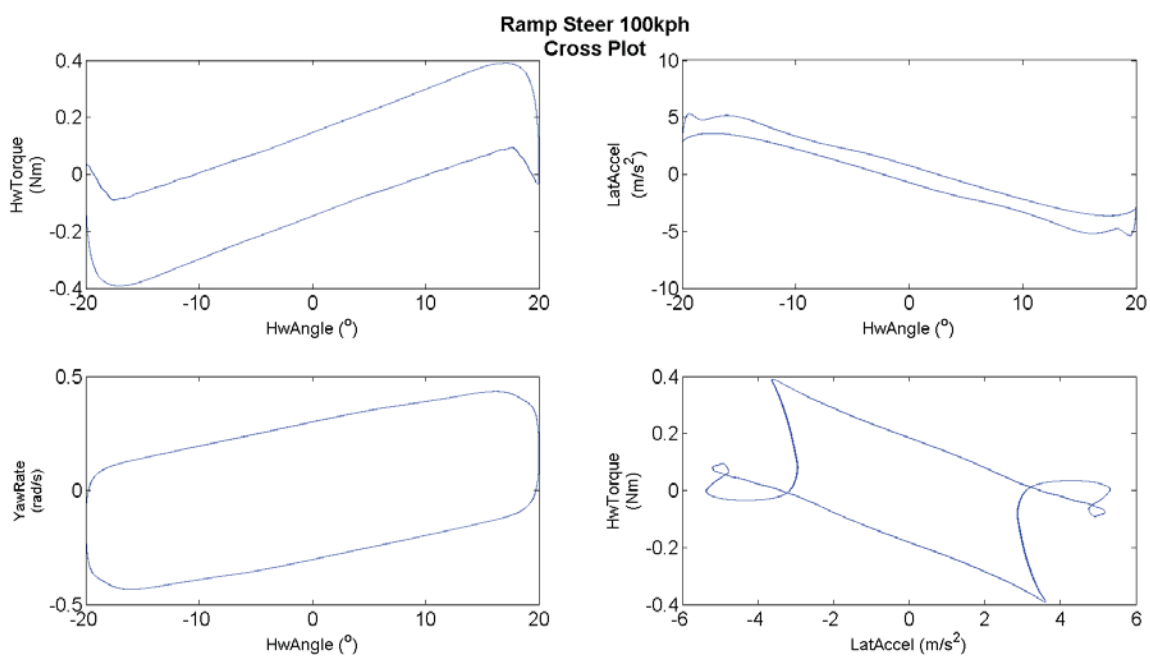

Figure 5.16 Ramp Steer Cross Plot @100kph

5.2.3 Single Lane Change Test

- Vehicle Speed: 20kph

  Handwheel Angle gain: 70

PID controller: KP: 5.5 KI: 0.05

Figure 5.17 and Figure 5.18 illustrate the Single Lane Change Test results at 20kph. All variables shown in time plots follow the steering input pattern at exactly the same frequency of 0.2Hz. The cross plots have an approximately linear response between handwheel angle and handwheel torque, lateral acceleration, yaw rate.

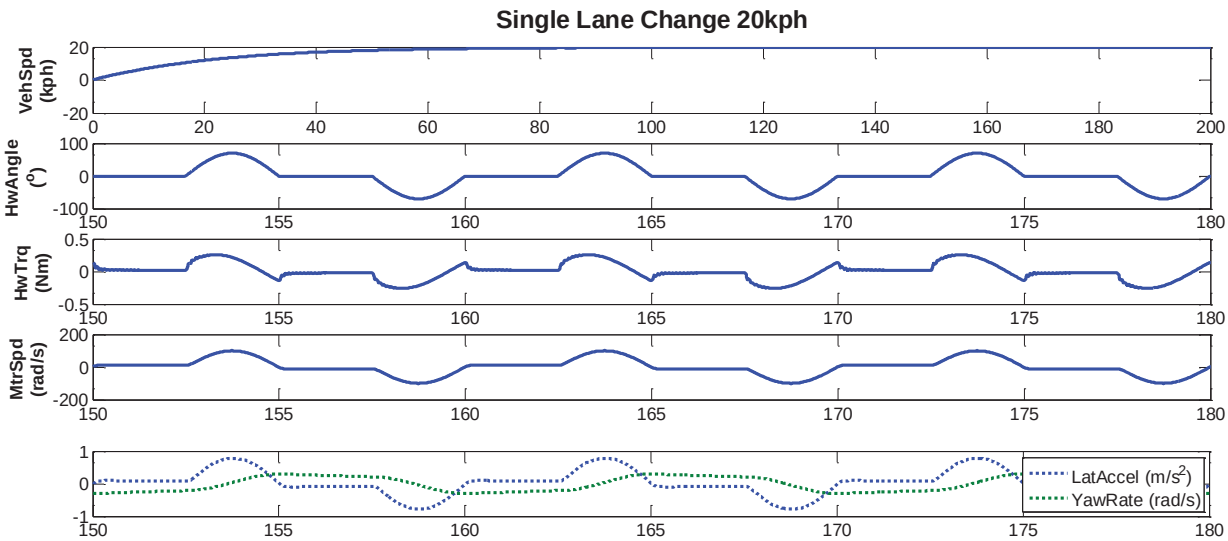

Figure 5.17 Single Lane Change Time Plots @20kph

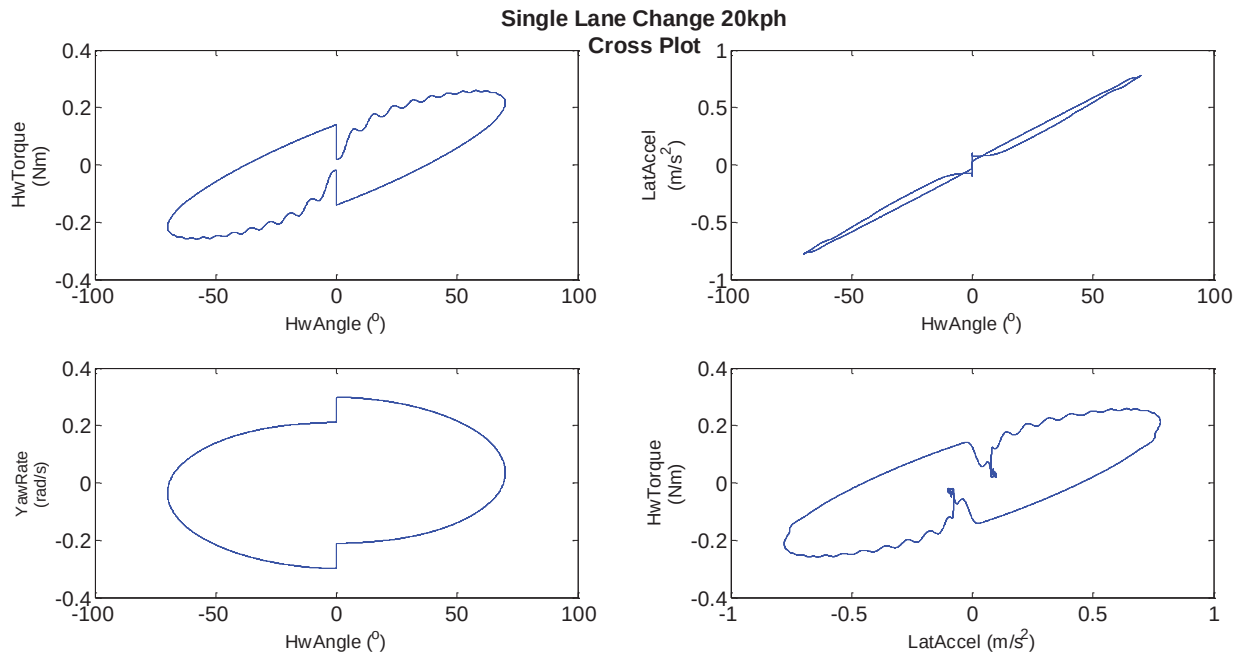

Figure 5.18 Single Lane Change Cross Plot @20kph

- Vehicle Speed: 60kph

  Handwheel Angle gain: 30

  PID controller: KP: 6 KI: 0.3

The test results for single lane change at maintained 60kph are shown in Figure 5.19 and Figure 5.20. The handwheel torque and lateral acceleration follow the steering input frequency, but they do have some oscillations over time, which can also be seen in cross plot.

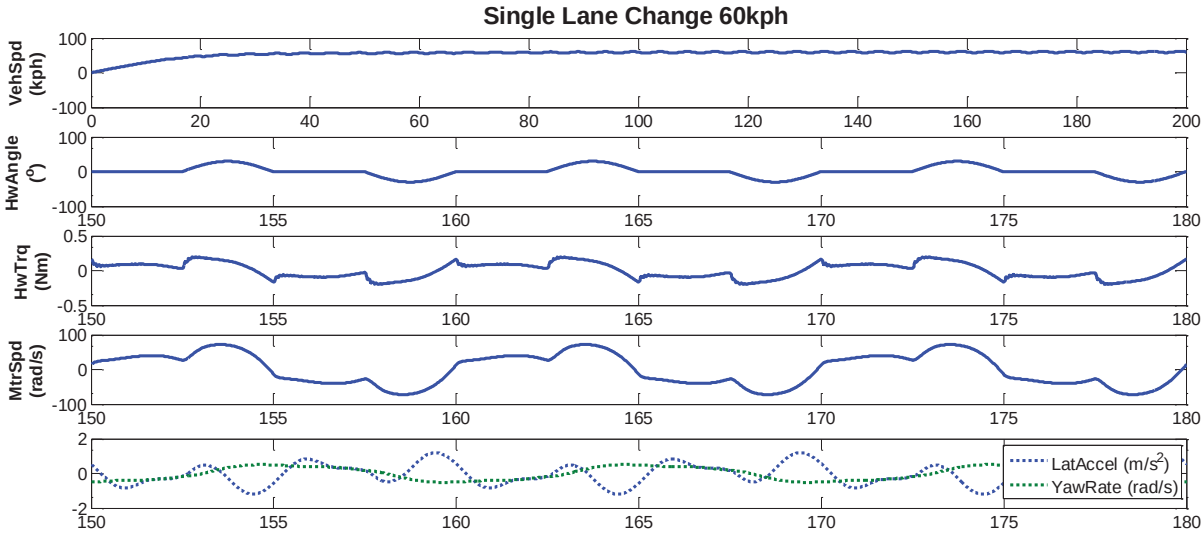

Figure 5.19 Single Lane Change Time Plots @60kph

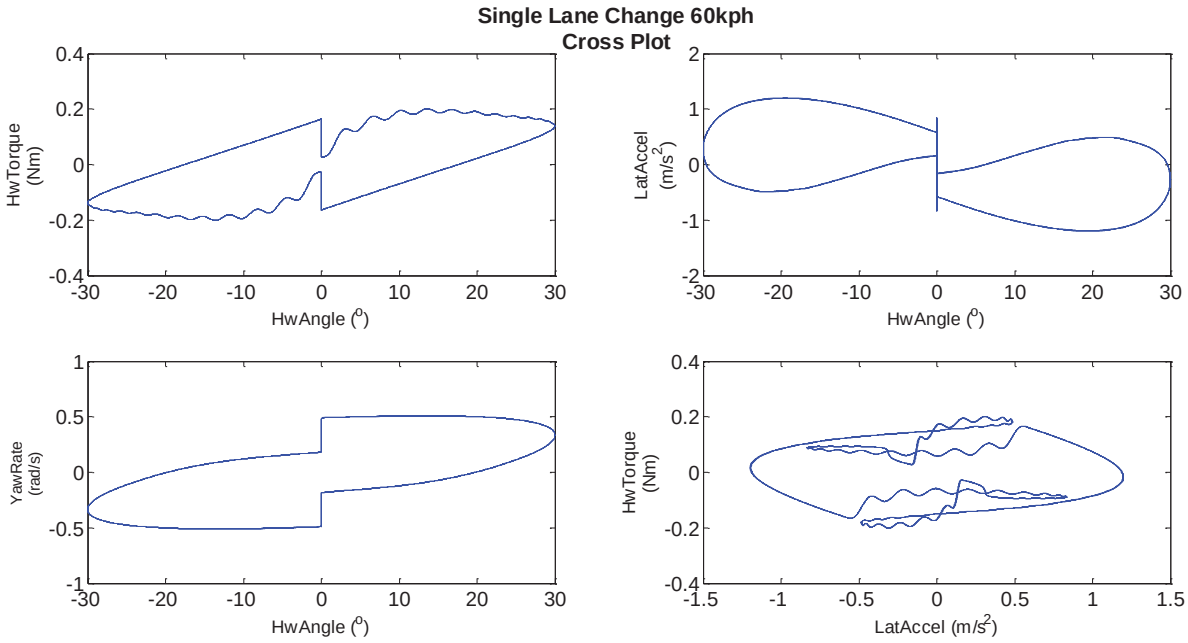

Figure 5.20 Single Lane Change Cross Plot @60kph

- Vehicle Speed: 100kph

  Handwheel Angle gain: 20

  PID controller: KP: 5 KI: 0.3

Figure 5.21 and Figure 5.22 give illustration about single lane change test results at 100kph. It is seen from the figure that handwheel torque and lateral acceleration behave a little oscillation over time.

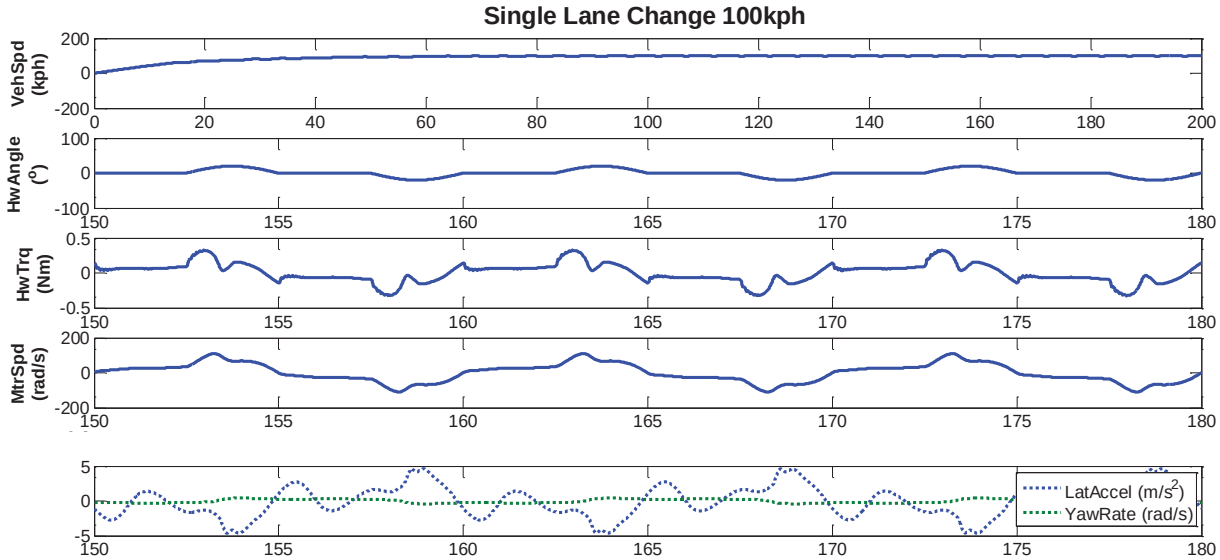

Figure 5.21 Single Lane Change Time Plots @100kph

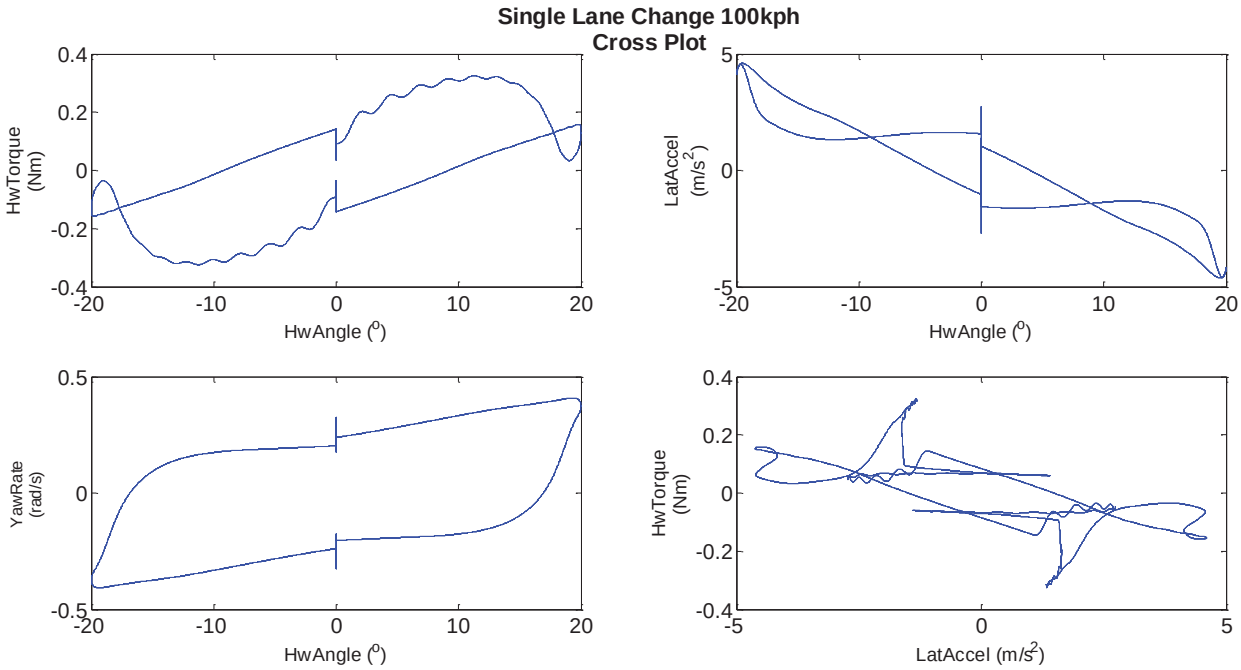

Figure 5.22 Single Lane Change Cross Plot @100kph

6 CONCLUSIONS

In this report, a dSPACE HIL system is developed for the testing of EPS ECUs to facilitate repetitive functional testing, cut down development cost, and enable early testing before production.

Sensors and Actuators are simulated, PWM Signal Measurement and CAN Module are configured to build to complete dSPACE HIL system for EPS ECUs testing. The set up system can be used on testing different vehicles different controller algorithms.

Reaction force is simulated by ASM Vehicle Dynamics Model to improve the reliability and accuracy of the HIL bench testing. The Integrated Mechanical Model, which is the combination of ASM Vehicle Dynamics Model and Nexteer Steering Model, simulate the driver on vehicle and calculate the steering torque which represents driver's feeling during steering. The model is validated via offline steering maneuver test, including ramp steering and single lane change. Specific testing cases are defined based on vehicle speed and steering input signals.

In the future, the dSPACE HIL Simulator is connected with EPS ECU with the whole EPS Model operating as plant model in real-time to test functionalities of the ECU, including base assist, active return, active damping, etc. In addition, the functionalities of the EPS ECU when interacted with other systems such as Electronic Stability Control (ESC) can be evaluated by means of combined test benches. [18] [19]

REFERENCES

1. Waeltermann, P., *Hardware-in-the-Loop: The Technology for Testing Electronic Controls in Automotive Engineering.* 6th Paderborn Workshop "Designing Mechatronic Systems", 2009.

2. Pfister, F., C. Reitze, and A. Schmidt, *Hardware-in-the-Loop - The technologie for development and test for vehicle control systems.*

3. Schuette, H. and P. Waeltermann, *Hardware-in-the-Loop Testing of Vehicle Dynamics Controllers - A Technical Survey.* SAE World Congress, 2005.

4. Grunewald, M.C. *Functional testing of an electric power steering using HiL simulations*. in *5th International Munich Chassis Symposium 2014*. 2014. Springer.

5. Lee, M.H., et al., *Development of a hardware in the loop simulation system for electric power steering in vehicles.* International Journal of Automotive Technology, 2011. 12(5): p. 733-744.

6. Kim, J.-H. and J.-B. Song, *Control logic for an electric power steering system using assist motor.* Mechatronics, 2002. 12(3): p. 447-459.

7. Segawa, M., et al., *Preliminary Study Concerning Quantitative Analysis of Steering System Using Hardware-in-the-Loop (HIL) Simulator*, 2006, SAE Technical Paper.

8. Lamberg, K. and P. Waeltermann, *Using HIL Simulation to Test Mechatronic Components in Automotive Engineering.* 2nd Congress on "Mechatronik im Automobil", 2000.

9. GmbH, d., *dSPACE Simulator EcoLine Hardware Installation and Configuration.* 2010.

10. GmbH, d., *Quad Power Performance boost for HIL simulation with new DS1006 Processor Board*, in *dSPACE Magazine*2010. p. 52-55.

11. GmbH, d., *DS2211 HIL I/O Board Features.* 2013(Release 2013-A).

12. GmbH, d., *DS5202 Electric Motor HIL Soluation Hardware and Software Installation and Configuration.* 2013(Soluation Version 13.1).

13. GmbH, d., *RTI CAN MultiMessage Blockset Reference For RTI CAN MultiMessage Blockset 2.9.* 2013(Relaease 2013-B).

14. GmbH, d., *RTI CAN MultiMessage Blockset Tutorial.* 2013(Release 2013-B).

15. GmbH, d., *ASM Vehicle Dynamics.* 2013.

16. GmbH, d., *ASM Vehicle Dynamics Reference For ASM Vehicle Dynamics Blockset 2.4 and ASM Vehicle Dynamics Operator Blockset 2.4.* 2013(Release 2013-A).

17. Yan, Q., et al., *Hardware in the Loop for a Dynamic Driving System Controller Testing and Validation.* SAE World Congress, 2005.

18. Himmler, A., P. Waeltermann, and M. Khoee-Fard, *Advancements in Hardware-in-the-Loop Technology in Support of Complex Integration Testing of Embedded System Software.* 2011. 1.

19. Waeltermann, P., H. Schuette, and K. Diekstall, *Hardware-in-the-Loop Testing of Distributed Electronic Systems.* 2004.

APPENDIX

A Single Lane Change waveform generation

```
% Single Lane Change waveform

period = 10;      % s
freq = 0.3;       % Hz
step_size = 0.001;

time1 = 0: step_size:period/2;
HwAngle_time = 0:step_size:period;
output1 = zeros(length(time1),1);

for ii = 1:length(time1)
if time1(ii)<period/4
    output1(ii) = 0;
else
    output1(ii) = -1*sin(2*pi*freq*time1(ii));
end
end

HwAngle_output = zeros(length(HwAngle_time),1);
HwAngle_output = [output1;-1*output1];
HwAngle_output(1) = [];
```

Figure A: Matlab code to generate Single Lane Change waveform