MBSD Lab #3

# Purposes

* Design a quarter-car physical model following the MAB modeling guidelines[[1]](#footnote-1).
* Design a skyhook control system following the MAB modeling guidelines.
* Validate the control system using the physical model and an opportune set of stimuli.

# Instructions

The system is organized into 3 models, like in Lab 2:

* **Harness.slx**, referencing models for the controller and plant and the test stimuli generation;
* **Controller.slx**, containing the controller (it operates at 1 KHz);
* **Plant.slx**, containing the plant model.

To develop the system, use the technical information in the file lectures/quarter car model and sh.docx available on the Dropbox of the course.

The harness.slx file shall contain test stimuli to simulate the road surface and to test the controller functionalities.

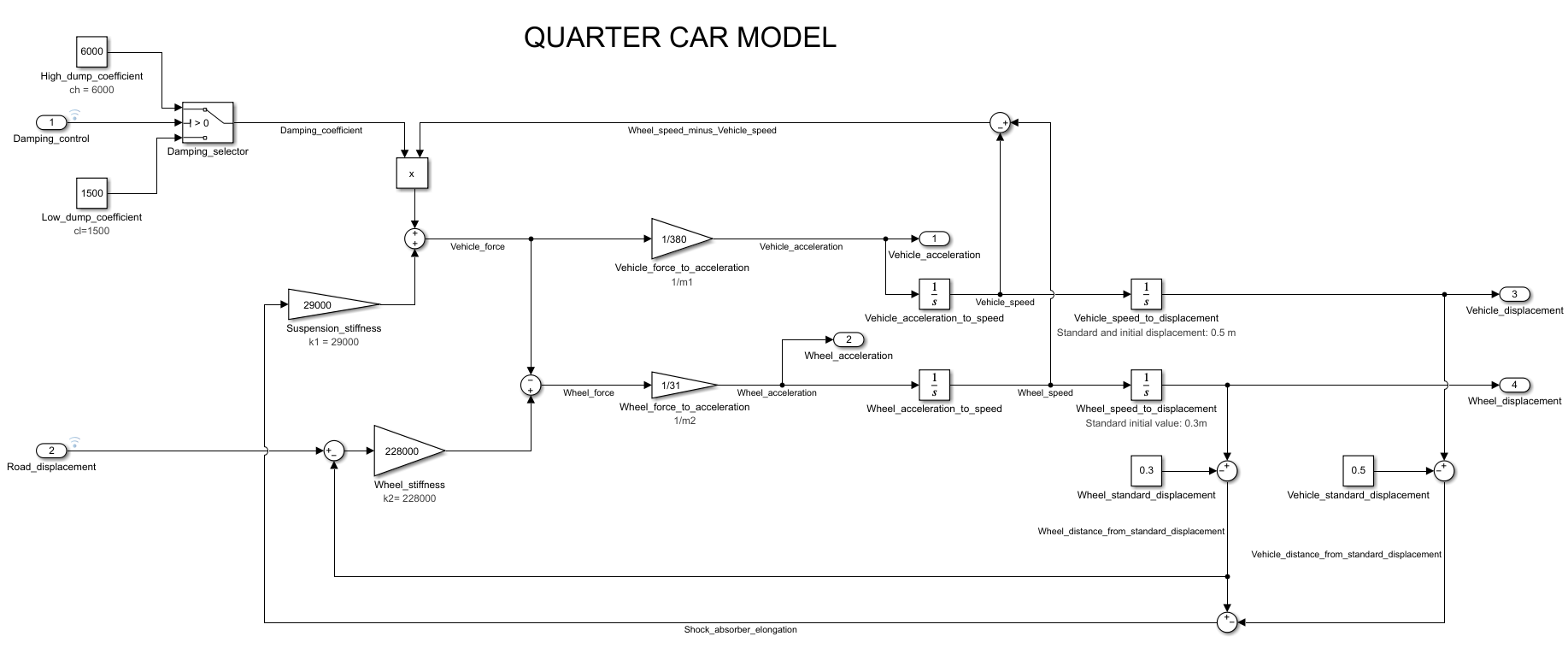
The controller shall provide an input to enable or enter the safe state with the .

Moreover, it provides an error checker that determines if the input signals are stuck to any value for more than 20 ms, indicating a malfunction of the sensors. This error checker is independent of the decision algorithm, but when it is triggered the controller shall enter the safe state.

The deliverable has to be provided as a .ZIP file up to **May 12th at 19:00.** It shall also contain a brief report explaining the design of the controller using the following template.Report template Lab 3

* Alessandro Fasiello ID: s297276
* Alice Giordano ID: s304003

# Plant model



The dynamic equation describing the suspension system is the following:

|  |  |
| --- | --- |
|  |  |

where , , and represent, respectively, the masses’ velocities and accelerations.

The previous equation can be split into two separate equations, yielding the vertical accelerations for the two masses:

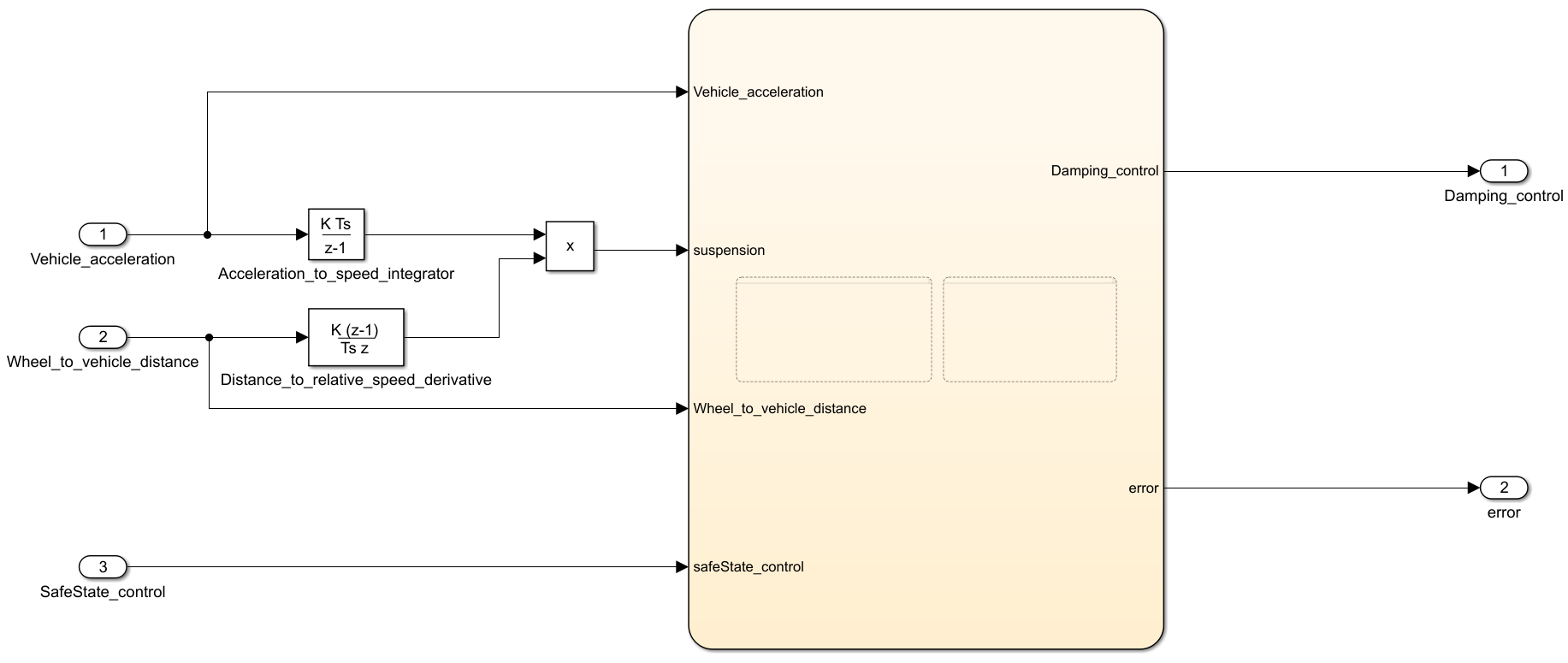
This control law has been implemented in Simulink using cascades of gains and integrators. A switch block has been used to actuate the damping selection based on the control input.

The Plant receives as input the damping control (Boolean) and the road displacement (double) while giving as output accelerations and displacements of both wheel and vehicle (double).

## Interfaces

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Unit\*** | **Type[[2]](#footnote-2)** | **Data Type[[3]](#footnote-3)** | **Dimension** | **Min** | **Max** |
| Damping control | N.A. | Input | Boolean | 1x1 | N.A. | N.A. |
| Road displacement | M | Input | Double | 1x1 | -0.2 | 0.2 |
| Vehicle acceleration | m/s2 | Output | Double | 1x1 |  |  |
| Wheel acceleration | m/s2 | Output | Double | 1x1 |  |  |
| Vehicle displacement | m | Output | Double | 1x1 |  |  |
| Wheel displacement | m | Output | Double | 1x1 |  |  |

# Controller implementation

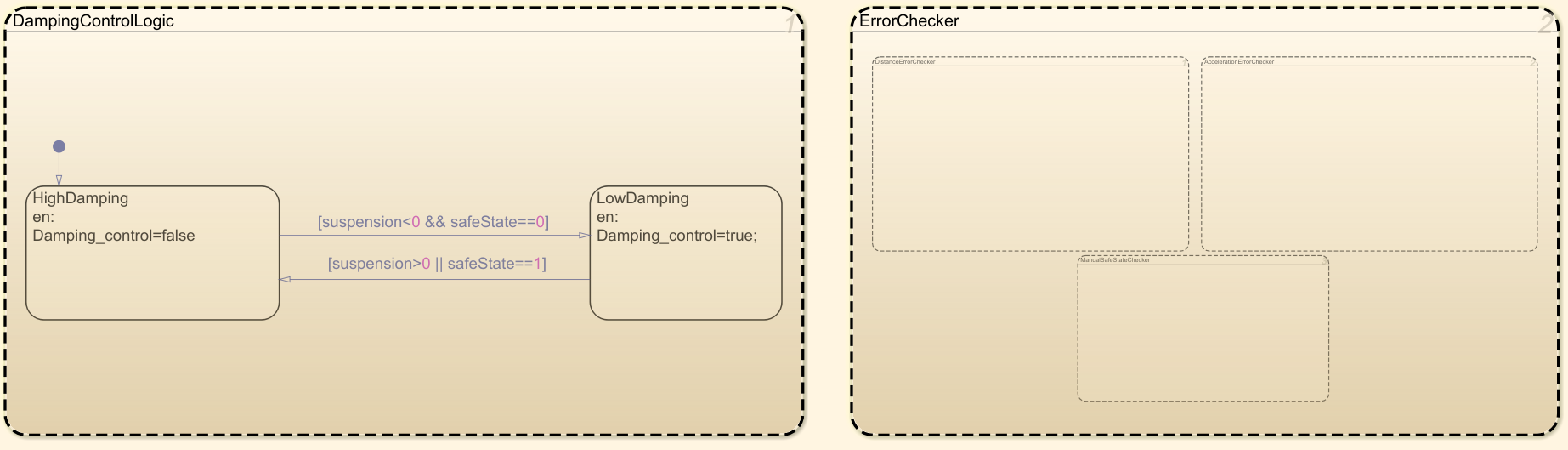


*[screenshot of the Simulink controller implementation]*

*Provide a brief description of the Controller functionalities and its interfaces (considering only the external interfaces, i.e., inports and outports of the controller.slx file).*

The given inputs Vehicle\_acceleration and Wheel\_to\_vehicle\_distance has been respectively integrated and derivated and then multiplied in order to obtain the suspension input of stateflow chart.

The last input is SafeState\_control, used to manually activate the SafeState mode, disabling the damping control.



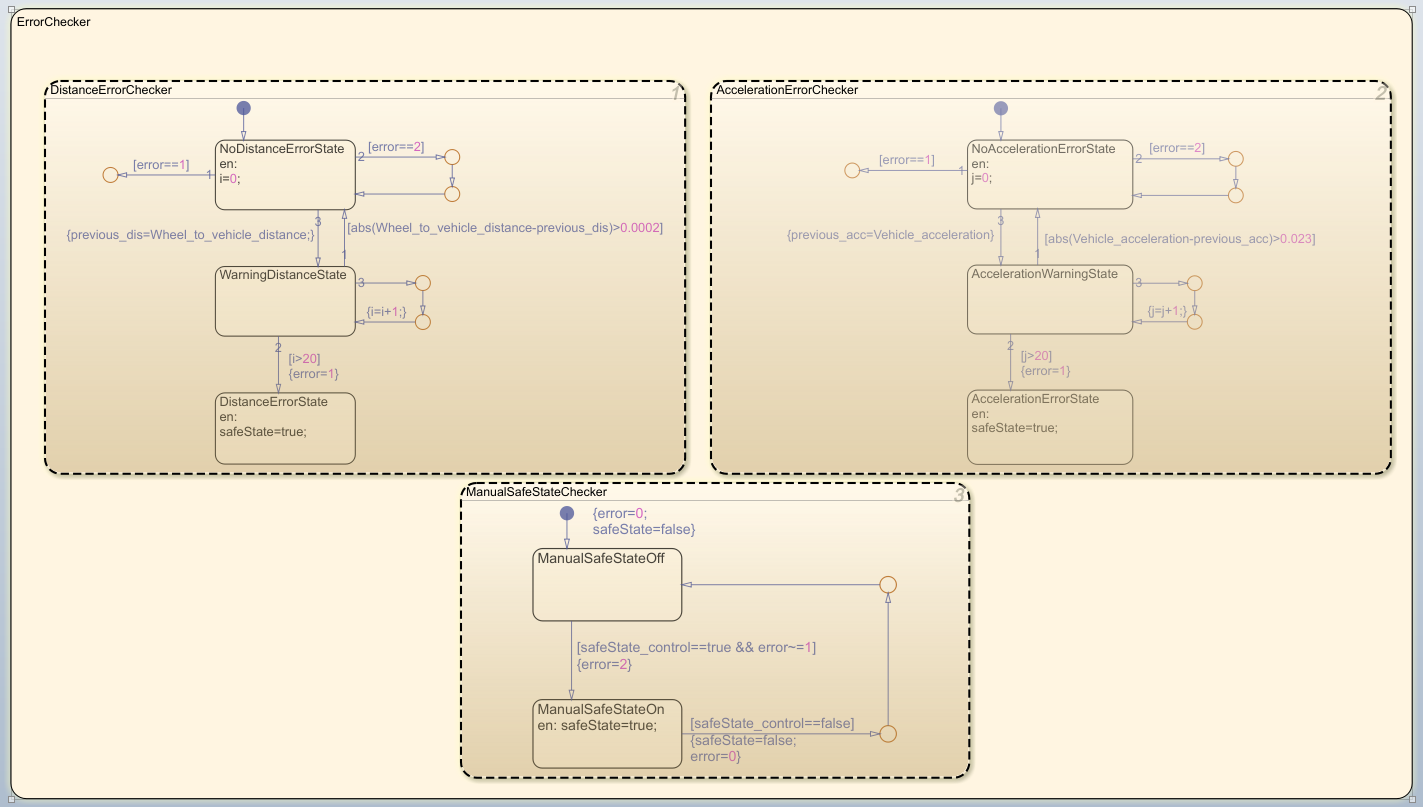
In the Stateflow chart we can find two parallel states that carry out the damping control logic and the error checking.

Immagine che contiene testo

Descrizione generata automaticamente

The first is simply composed of two exclusive states that enable the highest or the lowest value of damping based on the suspension data, that can be described by the following formula:

but remains in HighDamping state (or returns to it) if the safeState have been enabled by the ErrorChecker.



The Error checker is divided into three parallel states. The first one performs the control on the wheel\_to\_vehicle\_distance variable, the second does it on the Vehicle\_acceleration, both controlling that those values do not remain constant for more than 20 milliseconds and determining an error code equals to 1.

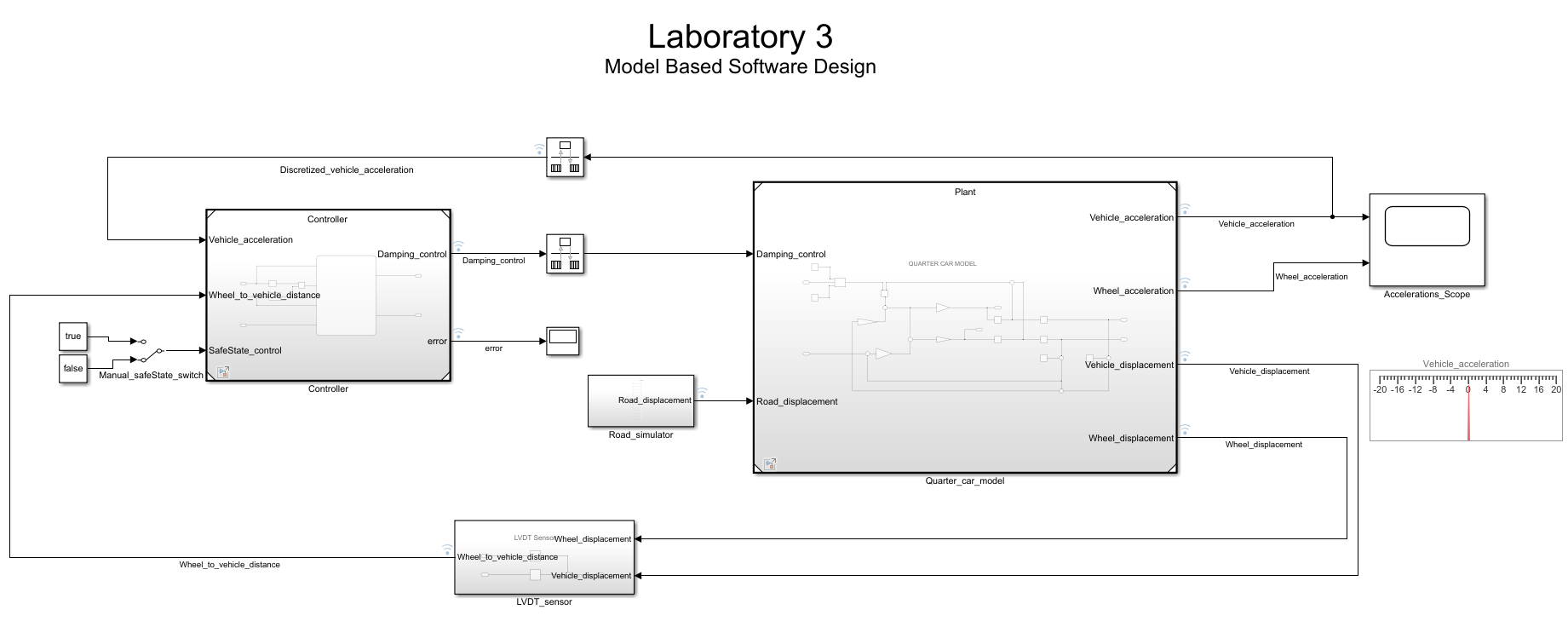
The third one simply enable the safe state based on the manual input, controlling an error code equals to 2.

The controller is a periodic task with a rate of 1 KHz. The solver is fixed-step.

## Interfaces

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Unit\*** | **Type[[4]](#footnote-4)** | **Data Type[[5]](#footnote-5)** | **Dimension** | **Min** | **Max** |
| Vehicle\_acceleration | m/s^2 | input | double | 1 | N.A. | N.A. |
| Wheel\_to\_vehicle\_distance | m | input | double | 1 | N.A. | N.A. |
| SafeState\_control | N.A. | input | boolean | 1 | N.A. | N.A. |
| Damping\_control | N.A. | output | boolean | 1 | N.A. | N.A. |
| error | N.A. | output | Uint8 | 1 | 0 | 2 |

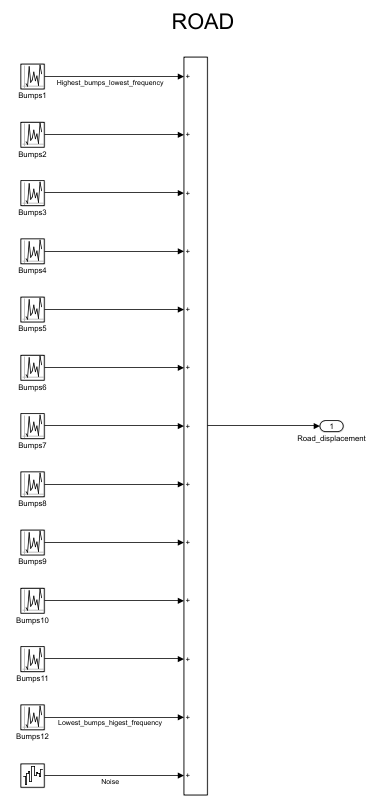
# Harness

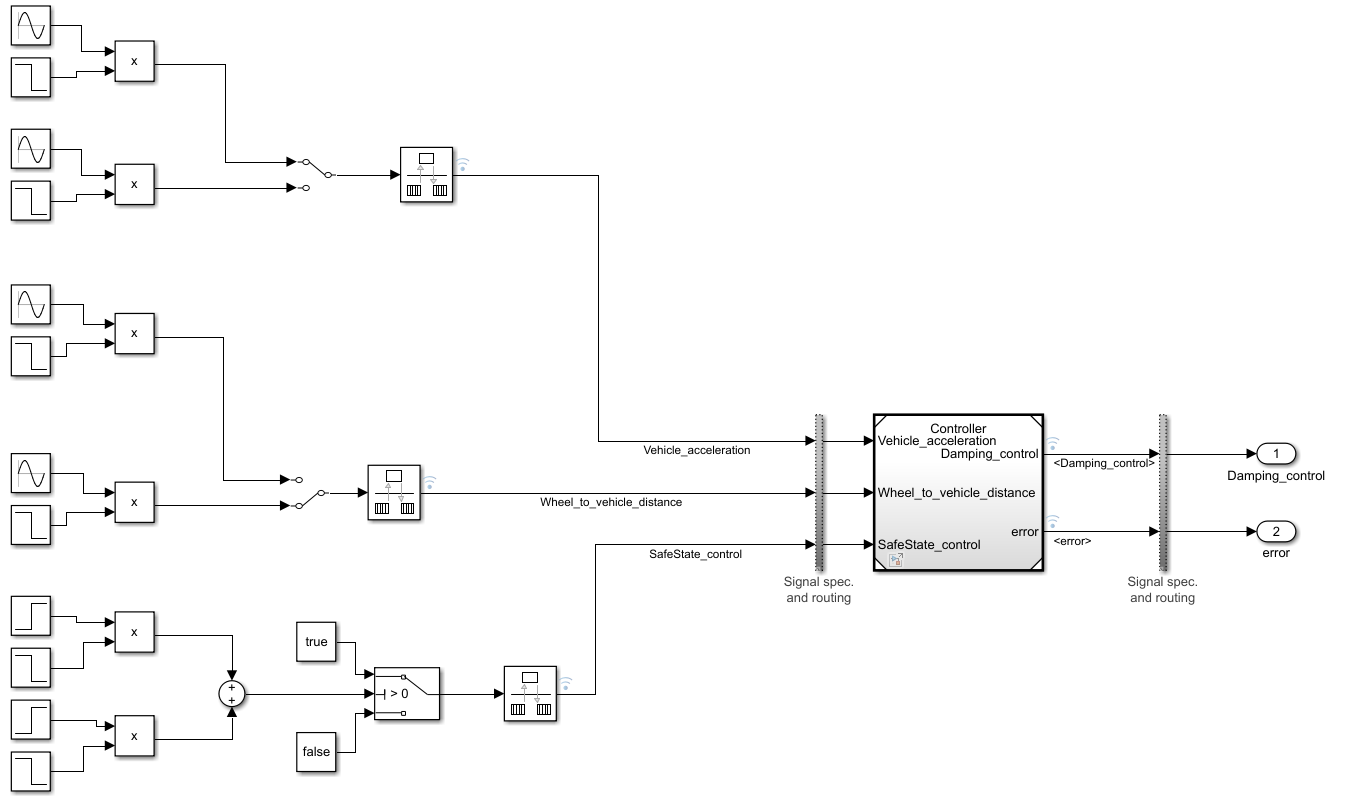


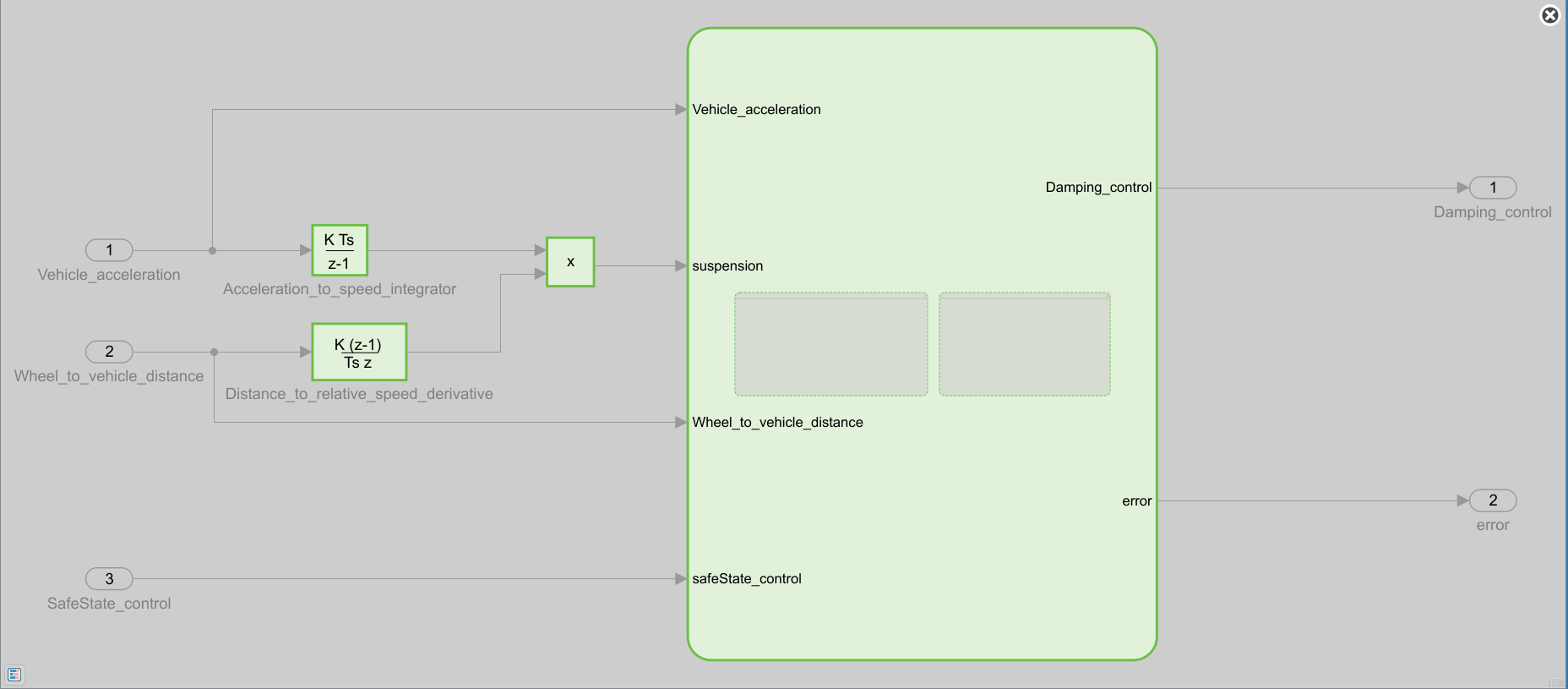
*[screenshot of the harness]*

## Test stimuli

*Provide a brief description of the stimuli set chosen to test the controller functionality.*







*Compare the performances in terms of vertical accelerations with the controller enabled or disabled.*

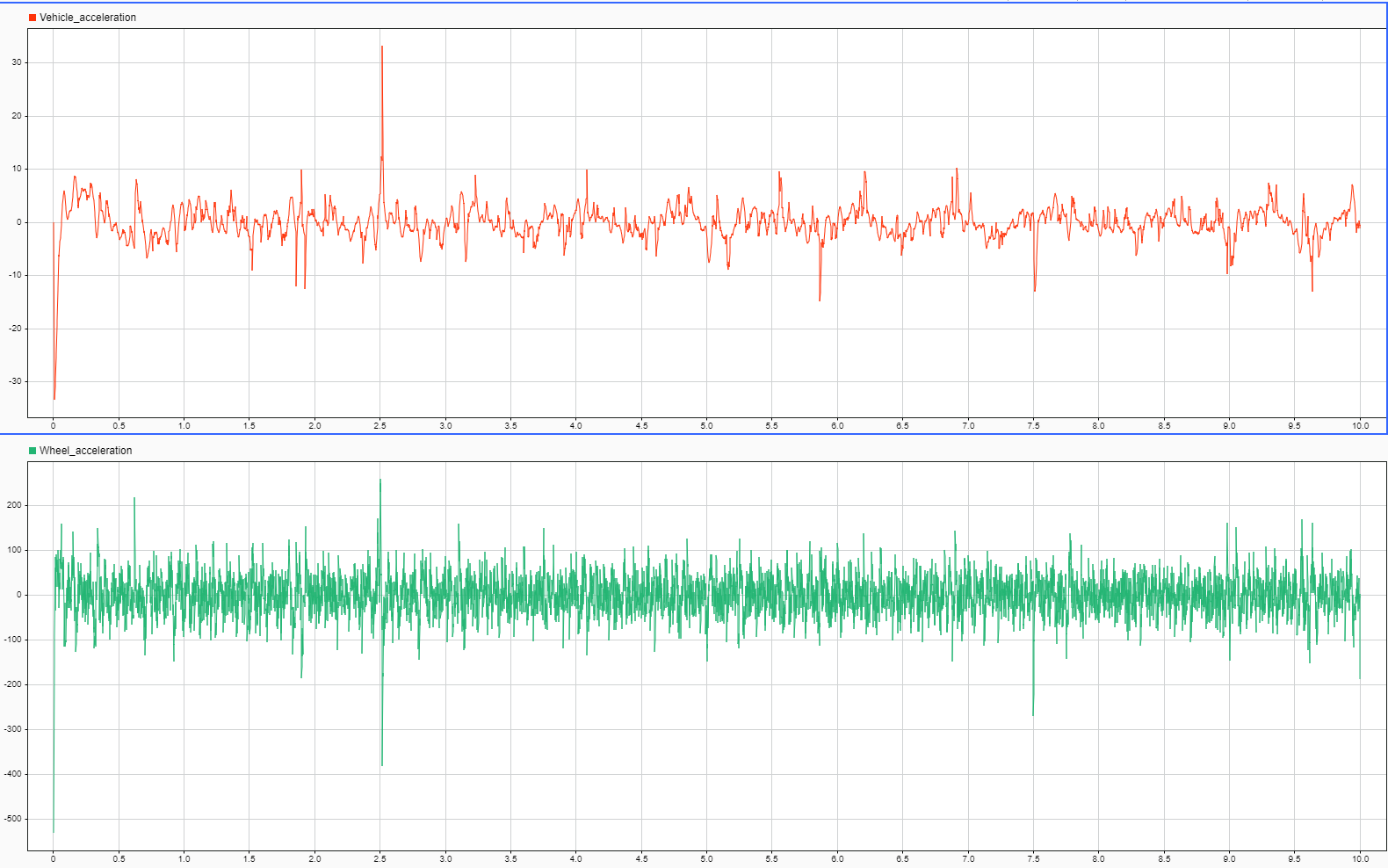
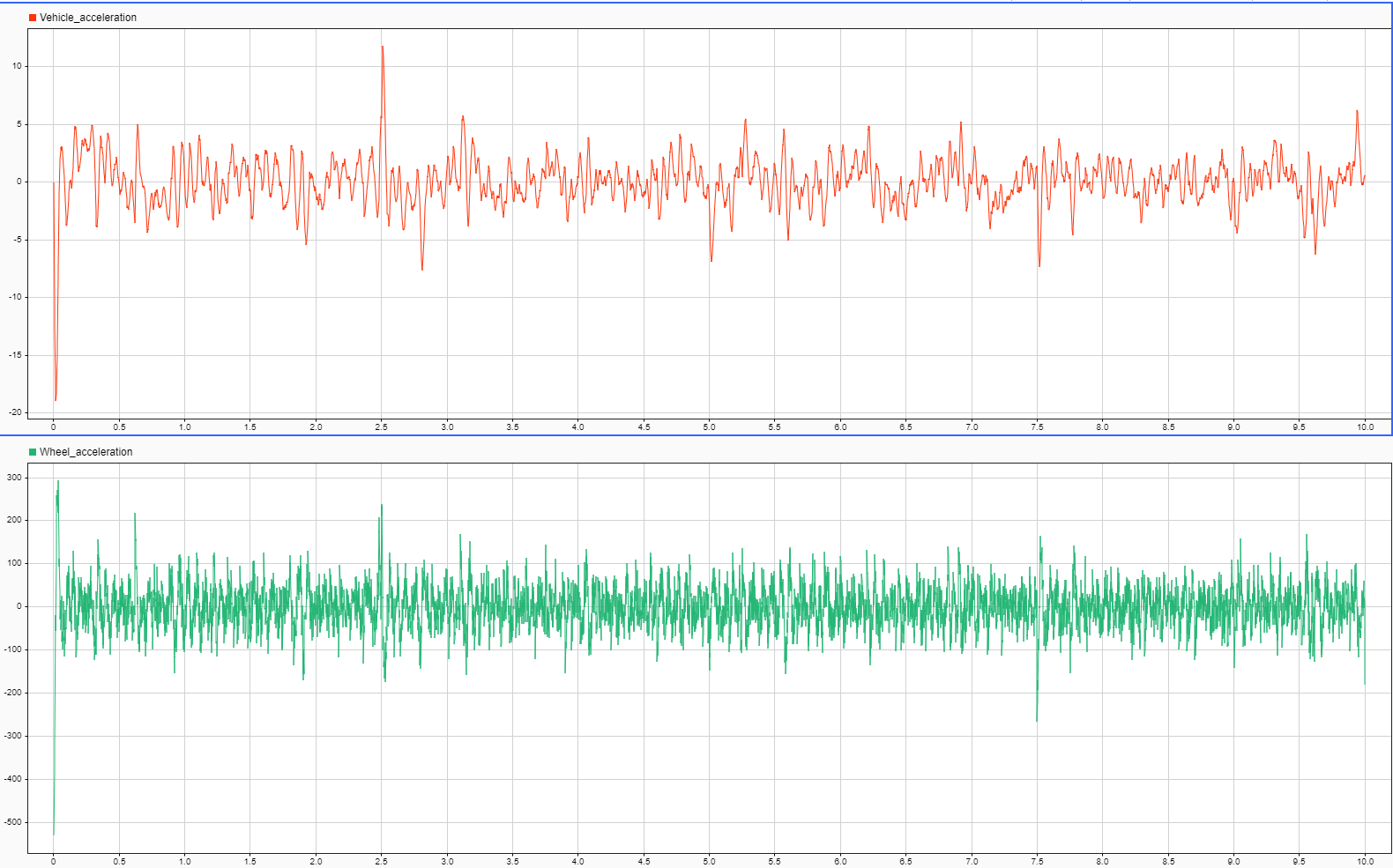


Figure 1: Damping control ON



1. <https://it.mathworks.com/content/dam/mathworks/mathworks-dot-com/solutions/mab/mab-control-algorithm-modeling-guidelines-using-matlab-simulink-and-stateflow-v5.pdf> [↑](#footnote-ref-1)
2. Input/Output/Global [↑](#footnote-ref-2)
3. Double, single, Boolean, uint8, etc.. [↑](#footnote-ref-3)
4. Input/Output/Global [↑](#footnote-ref-4)
5. Double, single, Boolean, uint8, etc.. [↑](#footnote-ref-5)