MBSD Lab #2 A.Y. 2022/23

# Purposes

* Implement the “one pedal controller” as a Simulink model.

To implement the system, follow the description already provided in the Laboratory 1 document considering also eventual assumptions already stated in this latter document.

The safety mechanisms are not required in this laboratory.

The Simulink project has to be split into 3 files:

* **Harness.slx**, containing reference models for the controller and plant and test stimuli generation
* **Controller.slx**, containing the controller (to be developed)
* **Plant.slx**, containing the car longitudinal physical model.

The longitudinal physical model and the test stimuli generators shall demonstrate the effectiveness of the implemented controller with respect to the expected functionalities.

Consider the plant model description as an example of the comments to be inserted in the Controller Software Unit description report.

Templates of these files are available alongside this document.

The deliverable, composed of the 3 Simulink models and a PDF file obtained by filling the following pages of this document (please delete this first page), has to be provided as a .ZIP file up to **May 14th at 23:59 through “Consegna Elaborati” of the Portale della Didattica.** It shall also contain a brief report explaining the design of the controller, using the following template. It is sufficient that only one of the group members uploads it.

# Model-Based Software Design, A.Y. 2022/23

# Laboratory 2 Report

## Components of the working group (max 2 people)

* Simone Bergadano, S303053
* Pietro Vignini, S317465

## External interfaces of the plant

|  |  |  |
| --- | --- | --- |
| **Name** | **Direction** | **Type** |
| **Requested\_Torque\_Nm** | Input | CAN |
| **Vehicle\_Speed\_km\_h** | Output | CAN |
| **Automatic\_Transmission\_Selector** | Input (from the driver to the controller) | CAN {P, R, N, D, P} |
| **Selected mode/errors** | Output (to the driver) | CAN |

## Equations of the plant

The plant considered in this model is the so-called *Vehicle Longitudinal Dynamics*.

Considering:



* the vehicle acceleration, expressed in []
* the vehicle longitudinal speed, expressed in []
* the vehicle mass, expressed in []
* the longitudinal force applied to the vehicle center of gravity, expressed in []
* the longitudinal force applied to the wheel on the terrain, expressed in []
* the longitudinal force applied to the vehicle center of gravity due to the frictions with air and terrain, expressed in []
* the moment of inertia of each one of the wheels, expressed in []
* the radius of the wheel, expressed in []
* is the angular speed of the wheel, expressed in [
* is the angular speed of the engine/electrical motor, expressed in [
* is the angular speed of the engine/ electrical motor, expressed in [
* is the angular acceleration of the wheel, expressed in [
* is the frontal surface of the car, expressed in []
* is the automobile drag coefficient
* is the average density of air at sea level in standard conditions à
* is the gearbox reduction ratio
* is the final drive reduction ratio
* is the total power train reduction ratio.

An extremely simplified model can be obtained as follow:

where is the vehicle acceleration, is its mass, is the longitudinal force applied to its center of gravity by the effects of the torque applied on the wheels, and is the sum of the friction forces on the vehicle due to wheel-terrain and vehicle-air interactions.

Considering that the torque is equally split between the two wheels (valid only on straight tracks)

the absence of slipping:

and considering the moment of inertia of the wheels , we can define the following equation, given that à .

The drag force that limits the maximum speed of the vehicle is equal to:

where:

and, as usually modeled:

By substituting the (2) equation in (1), and by integrating both sides, we obtain:

and, by substituting (3) in (6):

Remember that the integrator block of Simulink requires an initial condition corresponding to the vehicle's longitudinal speed at the beginning of the simulation, . A possible configuration of the integration block is shown in Figure 2.

During the model development, put all the needed gain to obtain as an output of the physical model a speed expressed in km/h.

To simulate the slope of the terrain, it is possible to add the gravity force as follows:

With the gravity acceleration on Earth.

Reasonable values for an electric compact car can be:

* The torque T (at the wheel) can vary in the range

Chart, line chart

Description automatically generated

Figure 1 Graph showing drag forces of tires (in orange) and air (in blue) at various speeds. It is possible to observe that, as imposed in equation (5), at 50 km/h. Below this speed, the tire drag is dominant, after that, the air drag is dominant. Moreover, it is possible to see the top speed of the car (around 230 km/h) when , with

With those values, the top speed on level ground reachable by the car is about 230 km/h, where the drag forces equal the traction force (3200 N).

Considering the reverse direction, the maximum speed reachable with a limitation of -60 Nm is about 45 km/h.

Graphical user interface, application, Teams

Description automatically generated

Figure 2 Settings window for the Integrator block of Simulink

Use these values (with a certain tolerance, for example, 10 %) to saturate the integrator block.

To make the model more realistic, it is possible to compute the torque request at the engine/motor. A typical ratio value for transmission of an electric car with a single gear can be around .

All the initialization parameters of the model are automatically loaded model by a callback of the function **init\_fn** as shown in Figure 3.

Graphical user interface, text, application

Description automatically generated

Figure 3 init\_fn callback configuration in the harness model properties.

## Description of the whole system

*Draw the I/O block diagram of the plant and of the controller, showing how they interact to each other.*

*Immagine che contiene schizzo, diagramma, disegno, testo

Descrizione generata automaticamente*

Figure 4 The Vehicle Longitudinal Dynamics model (Plant), its controller and the Driver model with all the needed I/O interfaces.

The controller provides the torque request through the TorqueRequest\_Nm variable to the Longitudinal Dynamics model of the Vehicle (Plant), which in turn returns the instantaneous speed of the vehicle in the variable Vehicle\_Speed\_km\_h.

The controller also displays the current automatic transmission mode through the AutomaticTransmissionState variable.

The instantaneous speed is used both by the Driver model and by the Controller. The former provides to the controller the position of the throttle pedal as ThrottlePedalPosition, the information about the brake pedal being pressed in the BrakePedalPressed variable and the selected automatic transmission mode in the variable AutomaticTransmissionSelectorState.

# Controller SW Unit specifications

*Provide a brief description of the Controller functionalities and its interfaces.*

## Interfaces

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Unit\*** | **Type** | **Data Type** | **Dimension** | **Min** | **Max** |
| BrakePedalPressed | N/A | Input | Boolean | 1x1 | 0 *(false)* | 1 *(true)* |
| ThrottlePedalPosition | N/A | Input | Single[[1]](#footnote-3) | 1x1 | 0 | 1 |
| AutomaticTransmissionSelectorState | N/A | Input | Enum  {P,R,N,D,B} | 1x1 | 0 | 4 |
| Vehicle\_Speed\_km\_h | km/h | Input | Single2 | 1x1 | -60 | 240 |
| AutomaticTransmissionState | N/A | Output | Enum  {P,R,N,D,B} | 1x1 | 0 | 4 |
| TorqueRequest\_Nm | Nm | Output | Single2 | 1x1 | -80 | 80 |

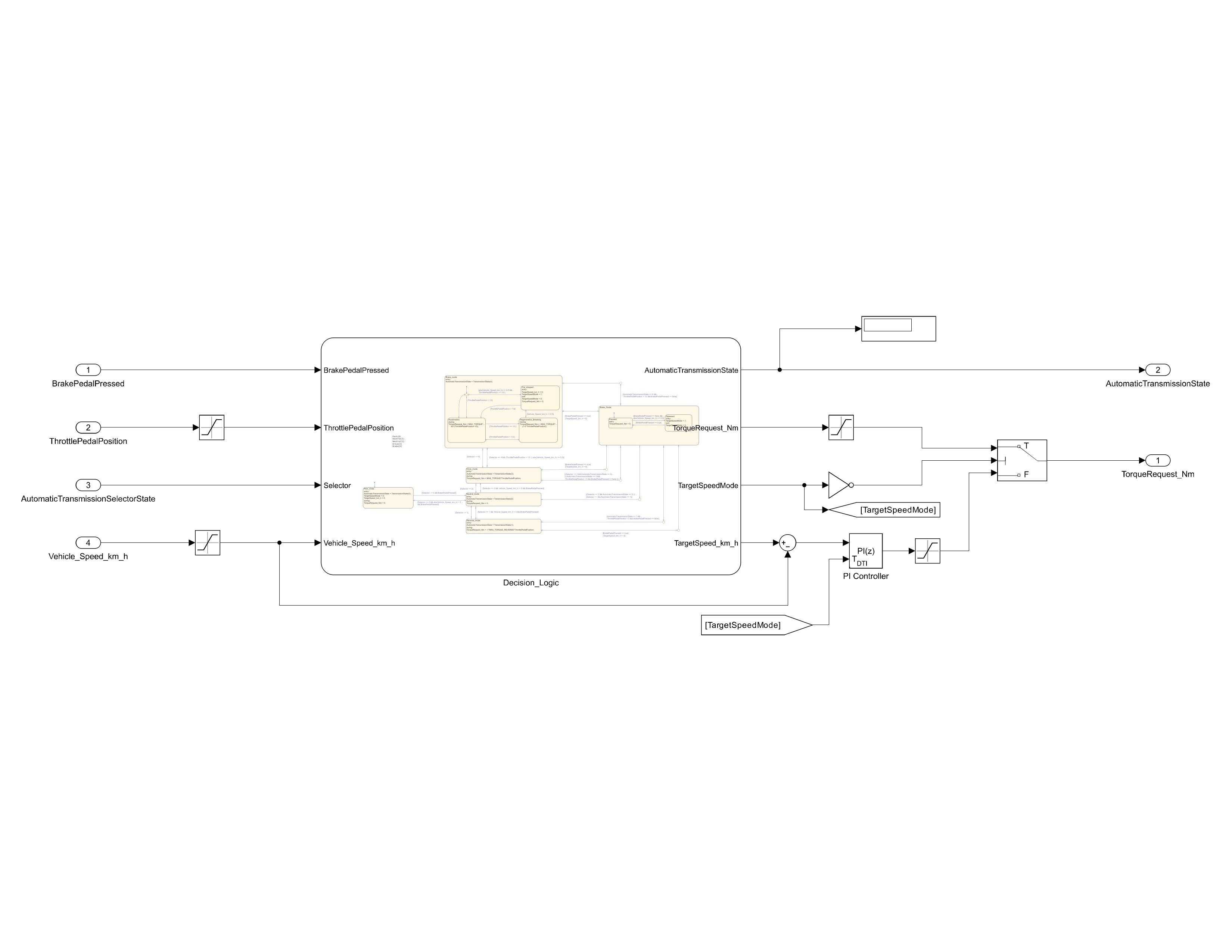
**

Figure 5 Simulink model of the Controller.

The closed-loop control implemented with a PI controller (Kp = 100, Ki = 0.1) allows to regulate the car to the target speed (TargetSpeed\_km\_h) imposed by the Decision\_Logic. The PI controller compute the desired torque request based on the speed error, given by the difference between the current speed and the desired one. Its activity is regulated through the local Boolean variable TargetSpeedMode. There are two possible scenarios where the PI controller is activated by the Decision\_Logic:

* To completely stop the car (TargetSpeed\_km\_h = 0) when the vehicle is in B mode and is already at low speed (Vehicle\_Speed\_km\_h <= 0.5), in order to prevent that, during regenerative braking, the negative torque request causes the vehicle to move in the reverse direction.
* To make the car move at low speed (TargetSpeed\_km\_h = 5 in D and B modes, TargetSpeed\_km\_h = -5 in R mode) when the brake pedal is pressed and then released, while waiting for the driver to take control of the car with the throttle pedal.

*Draw the Finite State Machine (FSM) representing the controller logic:*

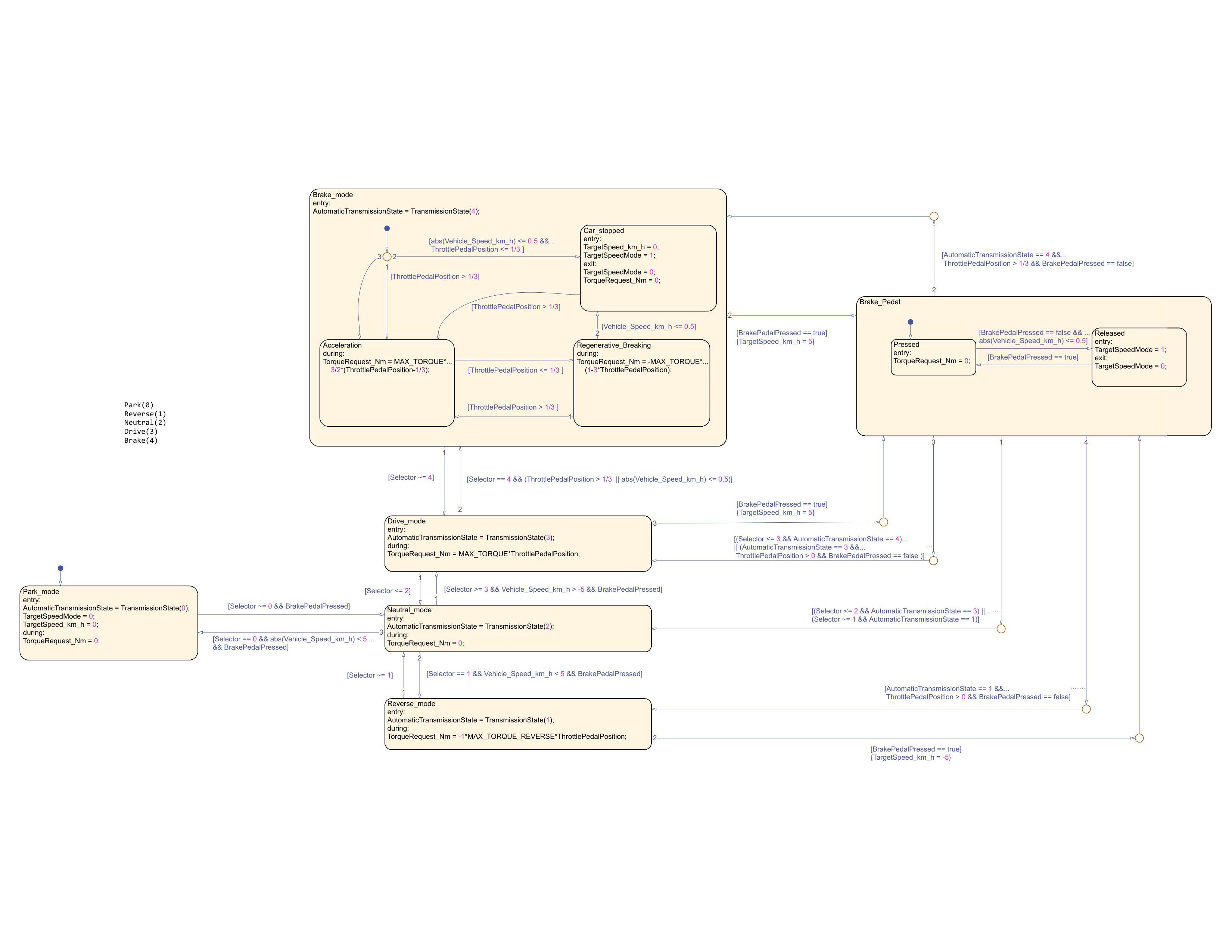


Figure 6 Finite State Machine (FSM) of the Decision\_Logic on Stateflow.

*Comment on the design choices of the FSM, which are not trivial to be understood just by analyzing the controller logic.*

* The input variable AutomaticTransmissionSelectorState has been renamed Selector inside the Stateflow chart for the sake of brevity.
* The Brake\_mode state can be reached from the Drive\_mode state only if that mode has been chosen by the driver through the selector and only if the ThrottlePedalPosition > 1/3 or abs(Vehicle\_Speed\_km\_h) <= 0.5. The latter condition allows switching to B mode also when the vehicle is (almost) stationary but no throttle pedal is being pressed. In this way the car can start directly in B mode.
* The Car\_stopped substate inside the Brake\_mode state completes the stopping of the car regardless of the road slope during the regenerative breaking by activating the TargetSpeedMode with target speed 0 km/h, preventing the reverse motion of the car due to the negative torque of the regenerative breaking. The system moves into that state when in B mode and with Vehicle\_Speed\_km\_h <= 0.5 and ThrottlePedalPosition <= 1/3.
* The Brake\_Pedal state has been introduced for two reasons: the first one is to assure that when the brake pedal is pressed no torque request is provided to the engine, even if for some reason the ThrottlePedalPosition > 0. This works as a brake-throttle override mechanism. The second reason is to allow the car to move after it has reached a full stop when the break pedal is pressed and then released. This is handled with the Released substate.
* The Released substate activates the TargetSpeedMode with target speed 5 or -5 km/h depending on the current driving mode (B, D or R). The system can enter this state only after the brake pedal has been pressed (to enter the Brake\_Pedal state) and then released, with the condition that the car has to be stationary (Vehicle\_Speed\_km\_h <= 0.5) when this happens.
* The system can return to the Brake, Drive or Reverse mode states from the Brake\_Pedal state only if it was already in that state prior to the brake pressing (AutomaticTransmissionState == 4, 3 or 1 respectively), the brake pedal is no longer pressed, and the driver is accelerating. The only exception to this is the condition (Selector <= 3 && AutomaticTransmissionState == 4) that allows the possibility to switch from the Brake\_mode to the Drive\_mode even during breaking. If the brake is not released the system will instantly return to the Brake\_pedal\_pressed state after updating the AutomaticTransmissionState to 3.
* The system can go from the Brake\_Pedal state to Neutral\_mode state in any moment if the Selector variable has changed to 2 or to a value that requires the passage in Neutral. This is possible only if the system was in Drive or Reverse before entering the Brake\_Pedal pressed: [(Selector <= 2 && AutomaticTransmissionState == 3) || (Selector ~= 1 && AutomaticTransmissionState == 1)].

If instead the system is in Brake\_mode state while the brake pedal is pressed, and the Selector goes to Neutral or Reverse, the system will firstly move to Drive and then to Neutral or Reverse depending on the conditions. The direct jump from Brake\_mode to Neutral through the Brake\_Pedal state is not allowed to respect the B --> D --> N sequence.

* When TargetSpeedMode switches to true, the requested torque no longer depends on the pedal position but is calculated to reach the speed set in TargetSpeed\_km\_h; the integrative action in addition to the proportional action gives greater stability to the controller and guarantees its correct operation even in the event of sloping road.

1. Single precision floating point number [↑](#footnote-ref-3)