Brake Torque Control on Ford CD4 Platforms

This document illustrates the differences between the CMD_TORQUE and CMD_TORQUE_RQ brake control modes. Both control modes allow the user to request a brake torque value, but CMD_TORQUE relies solely on an open-loop lookup table to convert brake torque commands to pedal position commands, whereas CMD_TORQUE_RQ actively closes the loop on the torque request signal transmitted by the vehicle. Dataspeed recommends using CMD_TORQUE_RQ instead of CMD_TORQUE in all cases for reasons that are illustrated below.

Users can command brake torque directly by selecting the CMD_TORQUE mode (CMD_TYPE = CMD_TORQUE), and setting PCMD_TORQUE to the desired brake torque level. The logic in the ADAS Kit then internally maps this commanded brake torque to a commanded brake-pedal position (PCMD_PEDAL), as shown in Figure 1. This mapping, however, solely uses an open-loop lookup table to make this conversion. (See Figure 2).

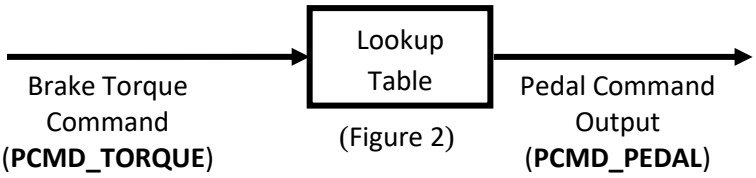

Figure 1. Open-loop brake torque mode.

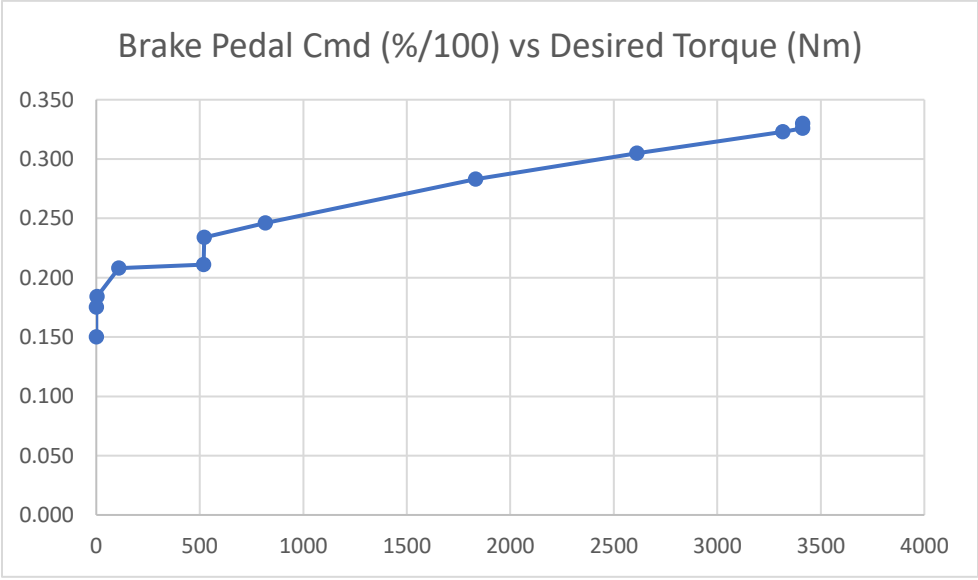

Figure 2. Open-loop lookup table that maps desired braking torque to brake pedal position.

Attenuation and Poor Accuracy in Open-Loop

Figure 3 shows an example of open-loop brake torque control behavior for the **CMD_TORQUE** mode. The blue line is the user's drive-by-wire brake command, and the red line is the brake torque request reported by the vehicle. Because of the open-loop nature of this command mode, the accuracy of the resulting brake torque request value is not guaranteed.

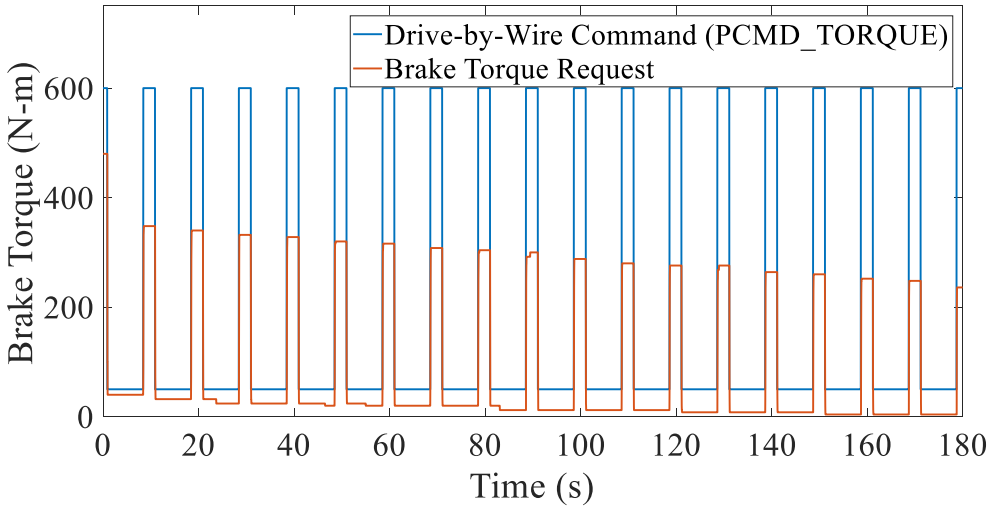

Figure 3. Attenuating torque output in open-loop brake torque mode.

(CMD_TYPE = CMD_TORQUE)

Moreover, this example also shows unexpected behavior for this vehicle platform. During long periods of slowly-changing, nonzero braking input, the amount of brake torque that is applied diminishes slowly over time, despite being issued a fixed torque command. This issue clearly cannot be accounted for by the sole use of an open-loop lookup table. (This slow decay is the result of action being taken by the vehicle itself, and not by the Dataspeed ADAS Kit.)

Improved Performance in Closed-Loop

The CMD_TORQUE_RQ command mode addresses the observed shortcomings of the open-loop CMD_TORQUE command mode. Figure 4 shows the results from running a similar test with CMD_TYPE set to CMD_TORQUE_RQ.

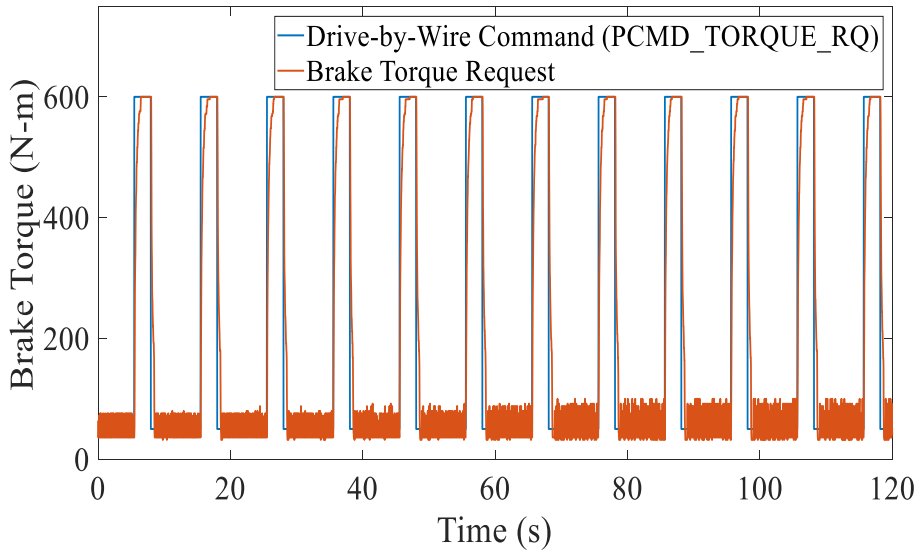

Figure 4. More accurate control in closed-loop brake torque mode.

(CMD_TYPE = CMD_TORQUE_RQ)

The architecture of the closed-loop brake controller behind the new CMD_TORQUE_RQ mode is illustrated in Figure 5. The same lookup table used in the CMD_TORQUE mode provides a feed-forward pedal position component that is added to the feedback component to produce the actual pedal position command sent to the vehicle. The feedback component is computed by applying PI control to the error signal defined as the difference between the user's brake torque command and the brake torque request as reported by the vehicle.

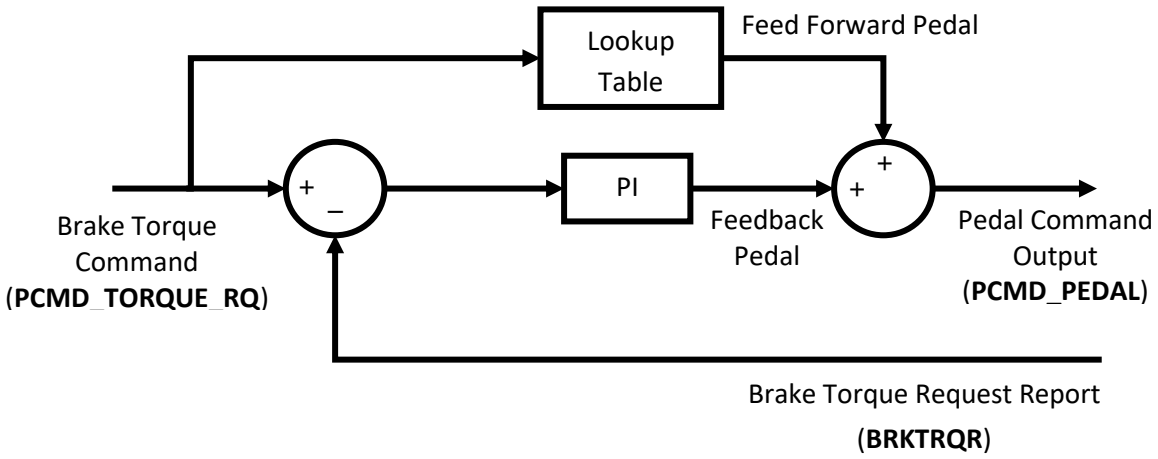

Figure 5. Closed-loop brake torque mode.

(CMD_TYPE = CMD_TORQUE_RQ)

Chattering at Low Torque Command Values

As seen in Figure 4, when the user requests a small level of brake torque, the closed-loop controller causes high-frequency changes in the brake torque request. This behavior is attributed to the nearly flat region in the brake torque output expressed as a function of pedal position, found between roughly 100 and 500 Nm. This nonlinearity causes a range of brake torque values to be unattainable using pedal commands. Therefore, if the user commands a brake torque value in this unattainable range, the closed-loop controller naturally pulses the command such that the average brake torque is close to the desired value. In practice, this chattering behavior is high enough in frequency and low enough amplitude to not be appreciably felt by passengers. It has also been shown to greatly improve braking performance in the affected torque range.