



## **Real-Time FPGA/CPU-Based Simulation of a Full-Electric Vehicle Integrated with a High-Fidelity Electric Drive Model**

Downloaded from: <https://research.chalmers.se>, 2025-12-05 03:11 UTC

Citation for the original published paper (version of record):

Jiang, B., Sharma, N., Liu, Y. et al (2022). Real-Time FPGA/CPU-Based Simulation of a Full-Electric Vehicle Integrated with a High-Fidelity Electric Drive Model. *Energies*, 15(5). <http://dx.doi.org/10.3390/en15051824>

N.B. When citing this work, cite the original published paper.

## Article

# Real-Time FPGA/CPU-Based Simulation of a Full-Electric Vehicle Integrated with a High-Fidelity Electric Drive Model

Bowen Jiang , Nimananda Sharma, Yujing Liu, Chuan Li and Xiaoliang Huang

Department of Electrical Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden; sharman@chalmers.se (N.S.); yujing.liu@chalmers.se (Y.L.); chuanl@student.chalmers.se (C.L.); xiaoliang.huang@chalmers.se (X.H.)

\* Correspondence: bowen.jiang@chalmers.se; Tel.: +46-729856515

**Abstract:** Real-time simulations refer to the simulations of a physical system where model equations for one time-step are solved within the same time period as in reality. An FPGA/CPU-based real-time simulation platform is presented in this paper, with a full-electric vehicle model implemented in a central processing unit (CPU) board and an electric drive model implemented in a field programmable gate arrays (FPGA) board. It has been a challenge to interface two models solved with two different processors. In this paper, one open-loop and three closed-loop interfaces are proposed. Real-time simulation results show that the best method is to transmit electric machine speed from the vehicle model to the electric derive model, with feedback electric machine torque calculated in FPGA. In addition, a virtual vehicle testing tool (CarMaker) is used when building the vehicle model, achieving more accurate modeling of vehicle subsystems. The presented platform can be used to verify advanced vehicle control functions during hardware-in-the-loop (HIL) testing. Vehicle anti-slip control is used as an example here. Finally, experiments were performed by connecting the real-time platform with a back-to-back electric machine test bench. Results of torque, rotor speed, and d&q axis currents are all in good agreement between simulations and experiments.



**Citation:** Jiang, B.; Sharma, N.; Liu, Y.; Li, C.; Huang, X. Real-Time FPGA/CPU-Based Simulation of a Full-Electric Vehicle Integrated with a High-Fidelity Electric Drive Model. *Energies* **2022**, *15*, 1824. <https://doi.org/10.3390/en15051824>

Academic Editor: Federico Barrero

Received: 11 February 2022

Accepted: 25 February 2022

Published: 1 March 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** real-time simulation; hardware-in-the-loop (HIL); battery electric vehicle (BEV)

## 1. Introduction

Vehicle electrification has become more and more important in recent years, especially when people put more attention on achieving low-carbon transportation and reducing harmful exhaust emissions. Battery electric vehicles (BEVs), as a gradually maturing solution, utilize energy storage systems (ESSs), DC/AC inverters, and electric machines (EMs) to produce traction force, replacing traditional powertrain systems consisting of internal combustion engines and fuel tanks [1]. Accordingly, many new technologies are being investigated to enhance the overall performance of BEVs. Offline simulation is the first step to develop a new vehicle function [2]. In offline simulation, the time limit on calculating the target system model is relatively low, because only the logic and feasibility of the new function need to be studied in this step [3]. On the contrary, real-time simulation aims at testing the maneuverability and effectiveness of the new function, and it is commonly used in hardware-in-the-loop (HIL) testing [4,5]. During HIL testing, it is of particular importance to coordinate the frequency of the function control signals and the real-time model simulation time-step [6].

Real-time models can be uploaded and run on central processing units (CPUs), field programmable gate arrays (FPGAs), etc. Due to serial execution inside one CPU core, CPU-based models need more time to be solved. As a result, their time-steps are around tens of microseconds [7]. Studies of most vehicle level behaviors do not need to be smaller than this time dimension. Similarly, vehicle level control functions also do not go to very high frequency. Therefore, CPUs are sufficient to run vehicle models as well as their

control functions. One FPGA board consists of a huge amount of logic elements, whose interconnection is reconfigured to develop functions. This design makes the logic gate parallel execution possible [8]. Thus, FPGA-based real-time models can be solved with extremely fast speeds, and their time-steps can be reduced to even less than ten nanoseconds. This feature shows great advantages in terms of running electric drive models, consisting of electric machine, inverter, etc. [9,10]. This is because active components inside inverters are operated with very high frequencies up to tens of kHz [11]. To study such high frequency behaviors, FPGA-based models with even smaller time-steps are necessary.

As is discussed, the vehicle model and electric drive model are normally solved with different processors. Therefore, a solution needs to be found in order to conduct real-time HIL testing of vehicle functions while considering electric drive performance. Some research has been carried out before to deal with this issue. The first method is to conduct offline vehicle simulation beforehand and save simulated EM torque and speed profiles. These profiles work as fixed inputs to the electric drive model [12,13]. This method excludes the vehicle model from real-time simulation. The electric drive model runs independently, but the real-time interaction between the electric drive and other vehicle subsystems cannot be studied. The second method is to replace the real-time electric drive model with a physical testing bench. The vehicle model and electric drive controller are compiled and uploaded to real-time simulators, such as Opal-RT [14], Typhoon [15], and dSPACE [16]. During the real-time HIL testing, the vehicle model is solved with CPU. High frequency electric drive signals can be observed from the testing bench by using oscilloscopes. However, the construction costs of the testing bench are high, and this method is also not conducive to verifying the general applicability of vehicle control functions under different electric drive parameters. The third method is to transmit control signals from the vehicle model to the electric drive model via analog to digital converters [17–19]. For example, in reference [19]’s method, the vehicle model is solved with DSP 100, while the electric drive model is solved with Typhoon HIL 402. PWM control signals of the inverter are sent from DSP to Typhoon during the real-time simulation, but no signal is fed back from the electric drive model to the vehicle model. Therefore, the dynamics of electric machine torque/speed do not affect the vehicle model. Additionally, both the second and the third methods have simplified vehicle models. As a result, nonlinear tire features, driver behaviors, and complex driving cycles are difficult to model accurately in these cases. Virtual vehicle testing tools (e.g., CarMaker, CarSim) are needed when more advanced vehicle functions are tested [2].

To solve the difficulties mentioned above, a new HIL testing platform is proposed in this paper, which integrates a CPU-based full-electric vehicle model and an FPGA-based electric drive model. The novelty of this platform can be summarized in two points. First, CarMaker from IPG Automotive is used to carry out vehicle modeling, and XSG from Xilinx is selected to perform electric drive modeling. The integration of these two simulation tools on conducting vehicle real-time HIL testing has not been carried out before. This method is able to verify advanced vehicle functions due to the high-fidelity vehicle model built in CarMaker. At the same time, electrical performance can also be observed by solving the high-fidelity XSG electric drive model. The vehicle anti-slip control is used to verify the proposed platform when nonlinear tire features and driver behaviors are considered. This would be very hard if only the simplified vehicle model is used. Second, one open-loop and three closed-loop interfaces are built between the vehicle model and the electric drive model. Results show that the best method is to transmit electric machine speed from the vehicle model to the electric drive model, and feedback the electric machine torque calculated by FPGA. In this way, the interaction between the electric drive and other vehicle subsystems is also considered during the real-time simulation.

The rest of this paper is arranged as below. Section 2 introduces full-electric vehicle and electric drive models. Section 3 proposes one open-loop and three closed-loop interfaces between the two models. Section 4 verifies the proposed real-time HIL platform with vehicle anti-slip control. In Section 5, experiment results obtained from a back-to-back

electric machine test bench are compared with the real-time simulation results, before some conclusions are finally presented in Section 6.

## 2. Vehicle and Electric Drive Modeling

### 2.1. CPU-Based Full-Electric Vehicle Modeling

A full-electric vehicle model consists of the modeling of the vehicle body, suspension system, steering system, brake system, tires, and electric powertrain system [20]. Based on different ways of torque transmission, there are front-wheel driven, rear-wheel driven, and four-wheel driven vehicles. In this paper, a distributed rear-driven powertrain system is selected, which has an EES, two DC/AC inverters, and two electric machines connected to the rear-left and rear-right wheels through two gear groups, as shown in Figure 1.

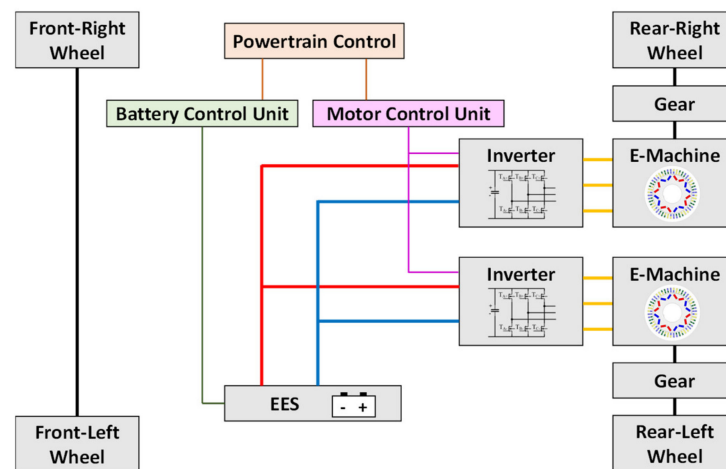


Figure 1. Distributed rear-drive full-electric powertrain.

In the longitudinal direction of the vehicle's driving, there are traction force ( $F_{trac}$ ) applied on the rear-driven wheels, rolling resistance force ( $F_{roll}$ ) applied on all four wheels, aerodynamic force ( $F_{aero}$ ) applied on the vehicle body, and gradient force ( $F_{grad}$ ) applied on the vehicle center of gravity (COG). The longitudinal acceleration of the vehicle ( $a_x$ ) can be determined by the above forces according to:

$$m \cdot a_x = F_{trac} - F_{roll} - F_{aero} - F_{grad} \quad (1)$$

where  $m$  is the total mass of the vehicle.

If we take one electric machine as an analysis object and have the same traction force on the left and right wheels, the electric machine acceleration can be calculated as below:

$$\frac{d\omega_{EM}}{dt} \cdot \left( (J_{EM} + J_{g1}) \cdot \eta_t + \frac{J_{g2} + J_w}{i_g^2} \right) = T_{em} \cdot \eta_t - \frac{F_{trac} \cdot r}{2 \cdot i_g} \quad (2)$$

where  $\omega_{EM}$  is the rotation speed of the electric machine;  $J_{EM}$  is one electric machine rotation inertial;  $J_w$  is one wheel rotation inertial;  $J_{g1}$  is the rotation inertial of the gear that has the same speed with the electric machine;  $J_{g2}$  is the rotation inertial of the gear that has the same speed with the wheel;  $i_g$  is the transmission ratio of the gear group;  $r$  is the wheel radius;  $T_{em}$  is the electromagnetic torque provided by one electric machine; and  $\eta_t$  is the transmission efficiency.

The relationship between the rotation acceleration of the electric machine and the longitudinal acceleration of the vehicle is shown below:

$$\frac{d\omega_{EM}}{dt} = \frac{a_x \cdot r}{i_g} \quad (3)$$

By substituting Equations (2) and (3) into Equation (1), Equation (4) is obtained, which describes the required electromagnetic torque provided by each of two electric machines to have a certain vehicle longitudinal acceleration.

$$T_{em} = \frac{r}{2 \cdot i_g \cdot \eta_t} \cdot (\gamma \cdot m \cdot a_x + F_{roll} + F_{aero} + F_{grad}) \quad (4)$$

The expression of the coefficient  $\gamma$  will be given in Equation (5). Since the electromagnetic torques given by two electric machines are not only used to accelerate the vehicle body, but also all of the rotation components (including rotors of electric machines, gear groups, and wheels), the coefficient  $\gamma$  can be understood as a ratio to convert both linear and rotation masses into one linear mass. The rotation inertial of two non-driven wheels shall also be considered when performing this conversion.

$$\gamma = 1 + \frac{2 \cdot \eta_t \cdot i_g^2 \cdot (J_{EM} + J_{g1})}{m \cdot r^2} + \frac{2 \cdot J_{g1} + 4 \cdot J_w}{m \cdot r^2} \quad (5)$$

Vehicle longitudinal acceleration is positively correlated with the traction force according to Equation (1). The traction force is mainly affected by three factors: vertical forces applied on driven wheels, road/tire conditions, and wheel slip ratio ( $\lambda$ ). It is common to have the vertical force vary among four different wheels. If the vehicle is driven in a straight trajectory, there is no load transfer between the left and the right wheels due to zero lateral acceleration. However, load transfer can happen between the front and the rear wheels due to longitudinal acceleration, road gradient, and the position of the vehicle COG. In our case, the vertical force ( $F_{rz}$ ) applied on one real driven wheel is calculated in Equation (6):

$$F_{rz} = \frac{1}{2} \cdot m \cdot \left( g \cdot \frac{l_f \cdot \cos(\theta) - h \cdot \sin(\theta)}{L} + a_x \cdot \frac{h}{L} \right) \quad (6)$$

where  $L$  is the longitudinal distance between the front and the rear axles;  $l_f$  is the longitudinal distance between the vehicle COG and the front axle;  $h$  is the height of vehicle COG; and  $\theta$  is the road slop angle.

The second factor, road/tire conditions, determines the friction coefficient. The friction coefficient, together with the vertical force calculated in Equation (6), decides the maximum traction force the road can provide to the driven wheels. The third factor, wheel slip ratio, describes the sliding between the tire surface and the road surface, as shown in Equation (7):

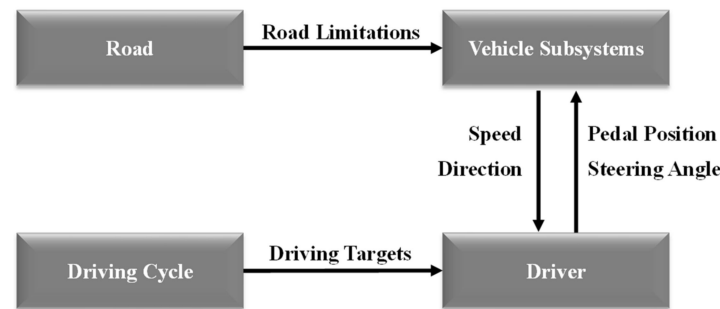
$$\lambda = \frac{r \cdot \omega_w - V_x}{|r \cdot \omega_w|} \quad (7)$$

where  $\omega_w$  is the rotation speed of the wheel, and  $V_x$  is the vehicle longitudinal speed.

The traction force changes with the wheel slip ratio. When the vehicle is driven on a normal asphalt road, the traction force reaches its maximum value when the wheel slip ratio is around 0.3 and will go down afterward [21]. If the electromagnetic torque keeps increasing after the traction force reaches its maximum value, the electric machine, as well as the wheel, will be accelerated rapidly according to Equation (2). The wheel slip ratio will also increase fast according to Equation (7). This may result in vehicle acceleration performance reduction and the loss of vehicle control, since much less lateral force can be applied to the tire surface. However, when the anti-slip function is used on driven wheels, the wheel slip ratio is continuously monitored. The reference torque of the electric machine can be reduced if the wheel slip ratio is detected to exceed its limit value.

Besides various vehicle subsystems, the vehicle model also includes the modeling of the driver, road, and driving cycle. As described in Figure 2, the driving cycle provides driving targets (direction, speed, etc.) to the driver, which are used to determine the pedal position and steering angle inputs to the vehicle subsystems. At the same time, the road can provide driving limitations to the vehicle, such as maximum traction and lateral force

based on road conditions. Vehicle maneuvering results are simulation outputs and are sent back to the driver to realize the close loop control.



**Figure 2.** The complete vehicle model consists of the modeling of vehicle subsystems, driver, road, and driving cycle.

## 2.2. FPGA-Based Electric Drive Modeling

Two permanent magnet synchronous machines (PMSM) and insulated-gate bipolar transistor (IGBT) inverters are modeled. They provide power/torque sources connected to the left-rear and right-rear sides of vehicle wheels. The PMSM electrical equation in the d&q phase is shown as below:

$$\begin{bmatrix} L_{di} & 0 \\ 0 & L_{qi} \end{bmatrix} \cdot \begin{bmatrix} \frac{di_d}{dt} \\ \frac{di_q}{dt} \end{bmatrix} = \begin{bmatrix} u_d \\ u_q \end{bmatrix} - R_s \cdot \begin{bmatrix} i_d \\ i_q \end{bmatrix} - \omega_{EM} \cdot \begin{bmatrix} -L_{qa} \cdot i_q \\ L_{da} \cdot i_d + \Psi_m \end{bmatrix} \quad (8)$$

where  $u_d$ ,  $u_q$ ,  $i_d$ , and  $i_q$  are the instantaneous d&q axis voltages and currents;  $R_s$  is the stator phase resistance;  $L_{di}$ ,  $L_{qi}$ ,  $L_{da}$ , and  $L_{qa}$  are the d&q axis incremental and apparent inductances; and  $\Psi_m$  is the magnet flux linkage.

Under the no-load condition, the magnet flux linkage in Equation (8) is equal to the d axis flux linkage. By integrating the current derivatives, instantaneous d&q axis currents can be obtained. Park transformation is used to transform voltages/currents between the d&q phase and the three-phase. According to Equation (9) and Equation (10), the d&q axis incremental and apparent inductance can be calculated from the d&q axis flux linkages, which can be obtained from finite element method (FEM) simulations of the reference electric machine.

$$L_{di} = \frac{\partial \psi_d^{(i_q=\text{constant})}}{\partial i_d}; \quad L_{qi} = \frac{\partial \psi_q^{(i_d=\text{constant})}}{\partial i_q} \quad (9)$$

$$L_{da} = \frac{\Psi_d}{I_d}; \quad L_{qa} = \frac{\Psi_q}{I_q} \quad (10)$$

The incremental inductance equals the apparent inductance when the iron-core is not saturated. Otherwise, the incremental inductance will be smaller. However, in our electric drive model, these two inductances are always assumed to be the same. The electromagnetic torque can be calculated from the apparent inductances, magnet flux linkage and currents, as shown in Equation (11).

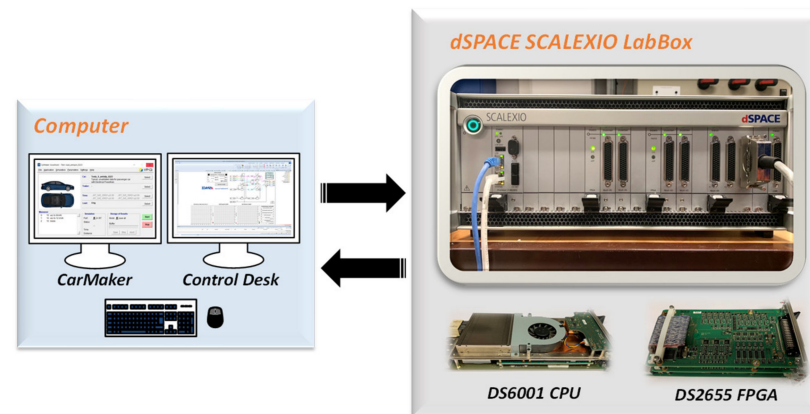
$$T_{em} = \frac{3}{2} \cdot p \cdot (\Psi_m \cdot i_q + (L_{da} - L_{qa}) \cdot i_d \cdot i_q) \quad (11)$$

A two-level voltage source converter with IGBT switches is modeled as the inverter in the electric drive model. An ideal switch model is utilized here. In other words, the switching transients and the losses of the IGBT are not included in modeling. The focus of the paper is to study the torque/speed interface between the vehicle model and the electric drive model. Therefore, the simplification of the switch and inverter modeling has little impact on the final results.



### 3. System Signal Interfaces

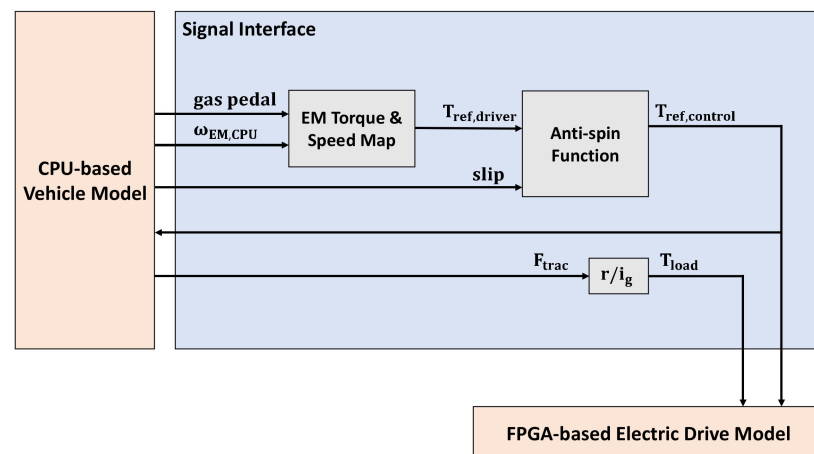
In this work, CarMaker from IPG Automotive is used to carry out the vehicle modeling, and XSG from Xilinx is used to carry out the electric drive modeling. Vehicle control functions and signal interfaces are implemented together with the vehicle model. As shown in Figure 3, a DS6001 board and a DS2655 board from dSPACE were selected as CPU and FPGA processors to run the vehicle model and electric drive model respectively in real-time. Control Desk from dSPACE provides access to tune model parameters and observe model variables. In the following subsections, interfaces between the vehicle model and the electric drive model are described.



**Figure 3.** The real-time HIL system used in this work.

#### 3.1. Open-Loop Interface

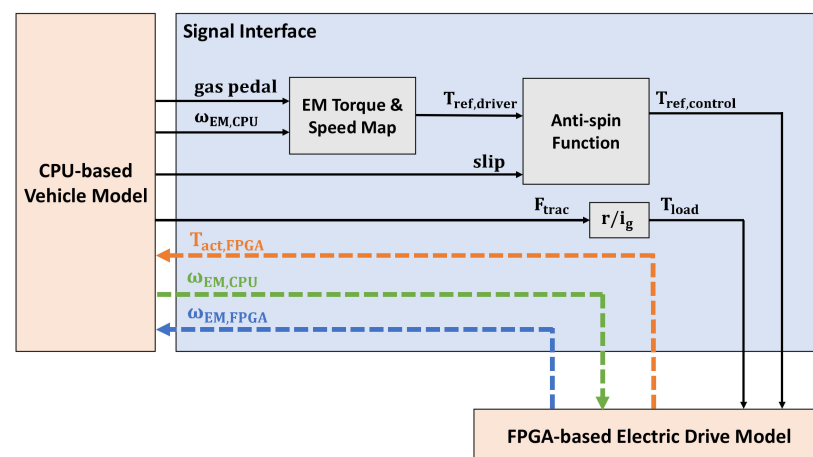
Figure 4 describes an open-loop interface between the vehicle model and the electric drive model. Gas pedal position and electric machine speed ( $\omega_{EM,CPU}$ ) signals are obtained from the vehicle model. They are used together with the operation map of the electric machine to determine the driver requested torque ( $T_{ref,driver}$ ). Driven wheel slip ratios are calculated in the vehicle model according to Equation (7) and sent to the anti-slip function block. Anti-slip control algorithms are used to find out the adjusted reference torque ( $T_{ref,control}$ ) based on the intents of the driver and slip conditions of the driven wheels. The adjusted reference torque is fed back to the vehicle model. It is also an input to the electric drive model. Another input is load torque ( $T_{load}$ ) applied on the electric machine shaft. When there is no brake action in the vehicle model, the load torque applied on the electric machine shaft is calculated from the traction force ( $F_{trac}$ ) according to Equation (2), considering the tire radius and the transmission ratio. When brake actions are performed (not shown in Figure 4), the brake pedal position and the regeneration rate are used to determine the adjusted reference torque, which has the opposite sign compared to the propulsion mode. The sum of the traction force given by the road surface and the brake force given by the mechanical brake system, which can both be obtained from the vehicle model, is used to calculate the load torque. In the electric drive model, the maximum torque per ampere (MTPA) method is used to find reference d&q currents, and electromagnetic torque is obtained according to Equation (11). This electromagnetic torque, together with the load torque from the vehicle model, is used to calculate rotor speed in the electric drive model. If the inertia and damping coefficients in the two models are the same, they should ideally have the same rotor speed. However, due to the bandwidth of torque control, the electromagnetic torque will not exactly follow the reference torque from the vehicle model. This torque inconsistency may make the rotor speed deviate in the electric drive model. In the open-loop interface, no signal is transmitted from the FPGA-based electric drive model to the CPU-based vehicle model, so the two models are always running parallelly.



**Figure 4.** Open-loop interface between the vehicle model and the electric drive model.

### 3.2. Closed-Loop Interface

There are three ways to implement closed-loop interfaces. The first one is torque closed-loop interface. As explained in the last section, an electromagnetic torque ( $T_{act,FPGA}$ ) is calculated in the electric drive model. In the torque closed loop, this electromagnetic torque is sent back to the vehicle model (orange dashed line in Figure 5). It is then used to solve the vehicle model equations. Traction force on driven wheels is calculated using this electromagnetic torque as well and converted into the electric machine load torque. By using the torque closed loop, the load torque sent to the electric drive model is related to the electromagnetic torque calculated inside the model. In addition, the mechanical performance of the electric machine, such as the torque ripple, can affect vehicle simulation. Since the actual torque is fed back to the vehicle model, the rotor speed deviation will be much smaller compared to the open-loop interface. The remaining deviation can be due to the sample delay in the feedback of the electromagnetic torque.



**Figure 5.** Closed-loop interface between the vehicle model and the electric drive model.

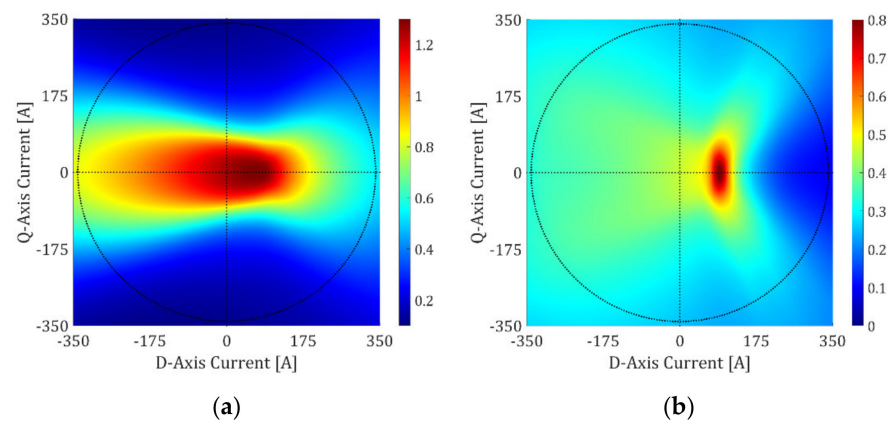
The second and third closed-loop interfaces can be referred to as speed closed-loop interfaces. They are both based on the torque closed-loop interface described above. The former one has the electric machine speed transmitted from the vehicle model to the electric drive model (green dashed line in Figure 5). The rotor speed calculation in the electric drive model is bypassed. Instead, the received electric machine speed from the vehicle model ( $\omega_{EM,CPU}$ ) is used. The closed loop in this interface is complete, which means that both torque and speed in the two models are always the same.



The latter one, on the contrary, sends rotor speed from the electric drive model ( $\omega_{EM,FPGA}$ ) to the vehicle model (blue dashed line in Figure 5). The rotor speed calculation is kept in the electric drive model, unlike before. As explained, the rotor speeds inside the two models may not be exactly the same due to the sample delay in torque feedback. Therefore, in this interface, the rotor speed from the electric drive model may cause wheel speed deviation in the vehicle model. This is because the load torque is indirectly dependent on driven wheel speed via the slip ratio. The deviation of the wheel speed can cause the vehicle model to detect wheel slip inadvertently. The load torque, which is an input to the electric drive model, in turn, affects the rotor speed calculation. Hence, oscillations in the speed and torque may occur in this interface.

#### 4. Real-Time Simulation Results

A PMSM with hairpin windings was selected as the reference electric machine in this work. The d&q axis inductances were calculated using the finite element method, and are shown in Figure 6.



**Figure 6.** Inductance maps of the electric machine: (a) d-axis (b) q-axis.

A medium-size full-electric car (such as Tesla Model S) was selected as the reference vehicle. Some key parameters of the reference vehicle and the reference electric drive are listed in Table 1.

**Table 1.** Parameters of the reference vehicle and the reference electric drive.

Parameter	Value	Unit
Car overall weight	1745	kg
Wheel radius	0.343	m
Wheel rotation inertial	2.673	kg·m <sup>2</sup>
Gear ratio	7	-
EM peak power (30 s)	60	kW
EM peak torque (30 s)	190	Nm
EM maximum rotor speed	9000	rpm
EM rotation inertial	0.018	kg·m <sup>2</sup>
EM pole pair number	4	-
EM stator resistance	26.4	mΩ
Inverter switching frequency	5	kHz
DC-link voltage	360	V

##### 4.1. Open-Loop Real-Time Simulation Results

The real-time simulation results of the open-loop signal interface are shown in Figure 7. The first subplot presents actual vehicle speed from the vehicle model. It is followed by the torque subplot, where the adjusted reference torque ( $T_{ref,control}$ ), electromagnetic torque ( $T_{act,FPGA}$ ), and load torque ( $T_{load}$ ) are shown. The third subplot compares the rotor speed

from the electric drive model ( $\omega_{EM,FPGA}$ ) and the vehicle model ( $\omega_{EM,CPU}$ ). The final subplot shows the reference and the actual d&q axis currents in the electric drive model.

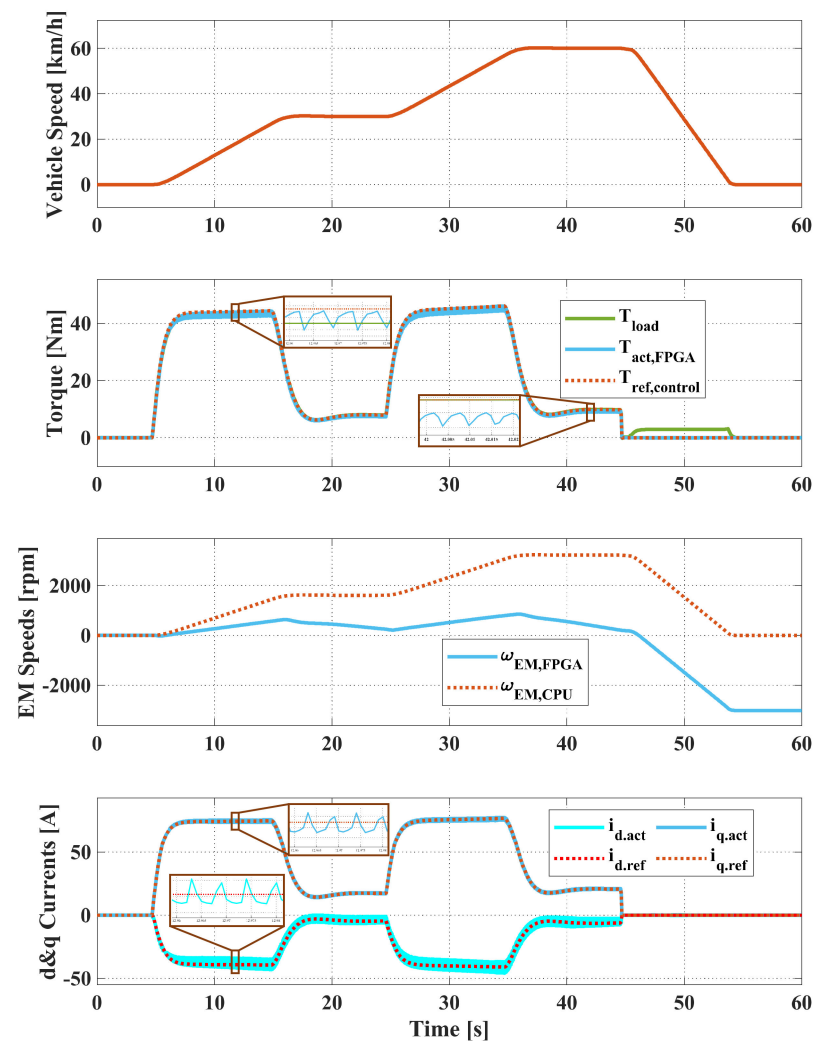


Figure 7. Open-loop real-time simulation results.

A speed-based driving cycle was used. The vehicle was first accelerated to 30 km/h in 10 s, followed by cruising for 10 s. Then, it was accelerated again to 60 km/h in a similar way. Finally, it was braked to a standstill within 10 s. The vehicle speed shown in Figure 7 followed this driving cycle closely. The adjusted reference torque was higher than the load torque during vehicle acceleration. This difference consisted of the losses in the transmission and rotation inertia. When the vehicle started to brake, the load and the reference torque dropped to zero due to the release of the gas pedal. The reference torque was kept at zero during the braking phase because regeneration was not used in this work. However, the non-zero load torque was caused by brake torque, which is not shown in the figure. The electromagnetic torque calculated from the electric drive model follows the reference torque. However, it is a little smaller, which can be explained by the oscillations of actual d&q currents compared to the reference values.

In Figure 7, subplot 3, the rotor speed in the vehicle model follows the vehicle speed, but the rotor speed in the electric drive model diverges. Two rotor speeds from two models were both calculated according to Equation (2), but in the vehicle model, the reference torque, instead of the electromagnetic torque, was used. In Figure 7, subplot 2, the first zoomed-in part shows that the difference between the reference torque and the load torque is bigger than the average difference between the electromagnetic torque and the load

torque. Thus, the rotor acceleration inside the vehicle model is bigger than that in the electric drive model. Similarly, in the second zoomed-in part of the same subplot, it can be observed that the reference torque and the load torque are identical as expected, since the vehicle speed is maintained unchanged at this moment. However, the average electromagnetic torque is lower than the load torque, which makes the rotor speed inside the electric drive model decrease.

#### 4.2. Closed-Loop Real-Time Simulation Results

The simulation results of three closed-loop interfaces are shown in Figures 8–10. The figures are plotted in the same order as described previously. In addition, the same speed-based driving cycle was used. The vehicle speed shown in Figure 8 also followed the drive cycle closely. The electromagnetic torque, during acceleration, was always higher than the load torque. This is because the electromagnetic torque was fed back to the vehicle model, which was used to calculate the load torque. As a result, the rotor speeds in both models followed the vehicle speed. However, the rotor speed in the vehicle model was slightly smaller. The sampling frequency of the CPU-based vehicle model is much smaller compared to the FPGA-based electric drive model. Therefore, the average electromagnetic torque was different in the two models, which can cause two rotor speeds to differ. This issue can be solved by using the second closed-loop interface, as shown in Figure 9. The rotor speeds from two models are almost identical, because of the speed feedback from the vehicle to the electric drive model. There will only be one sample delay caused by data transfer, which is negligible.

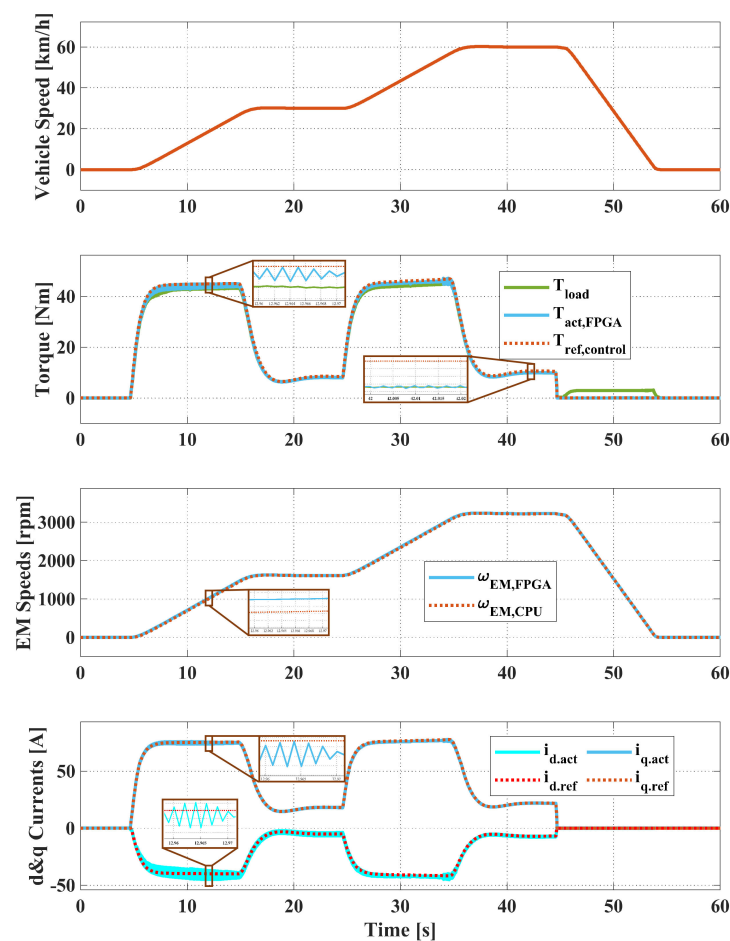
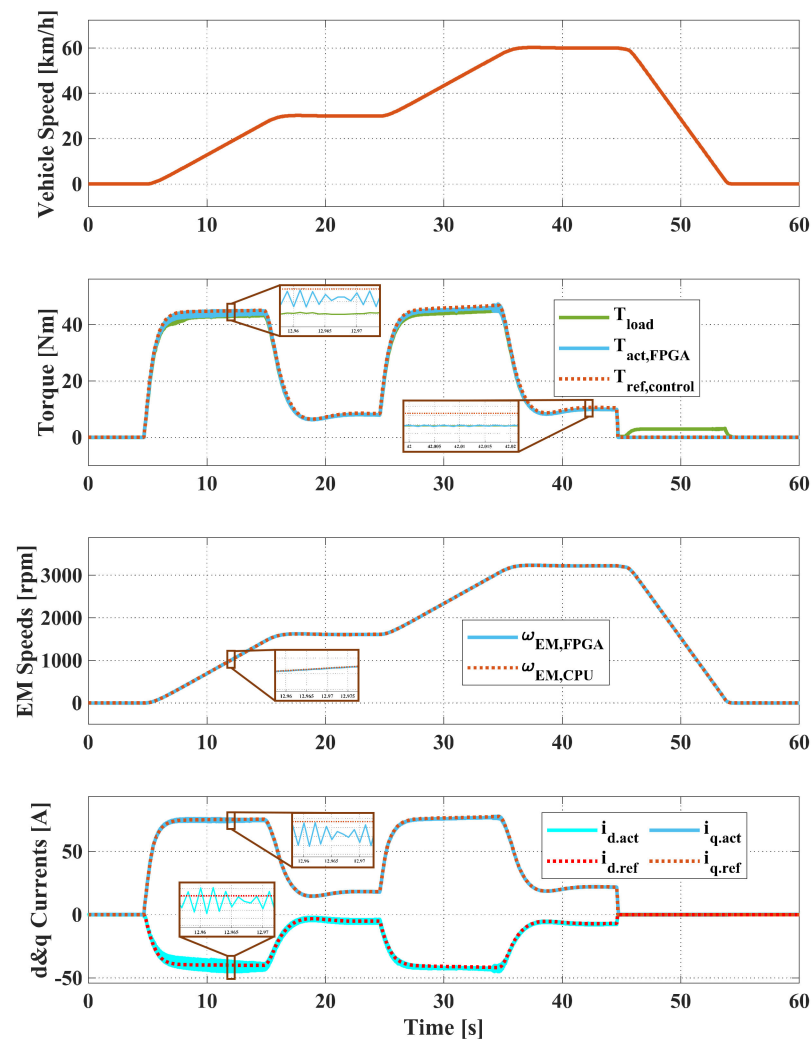
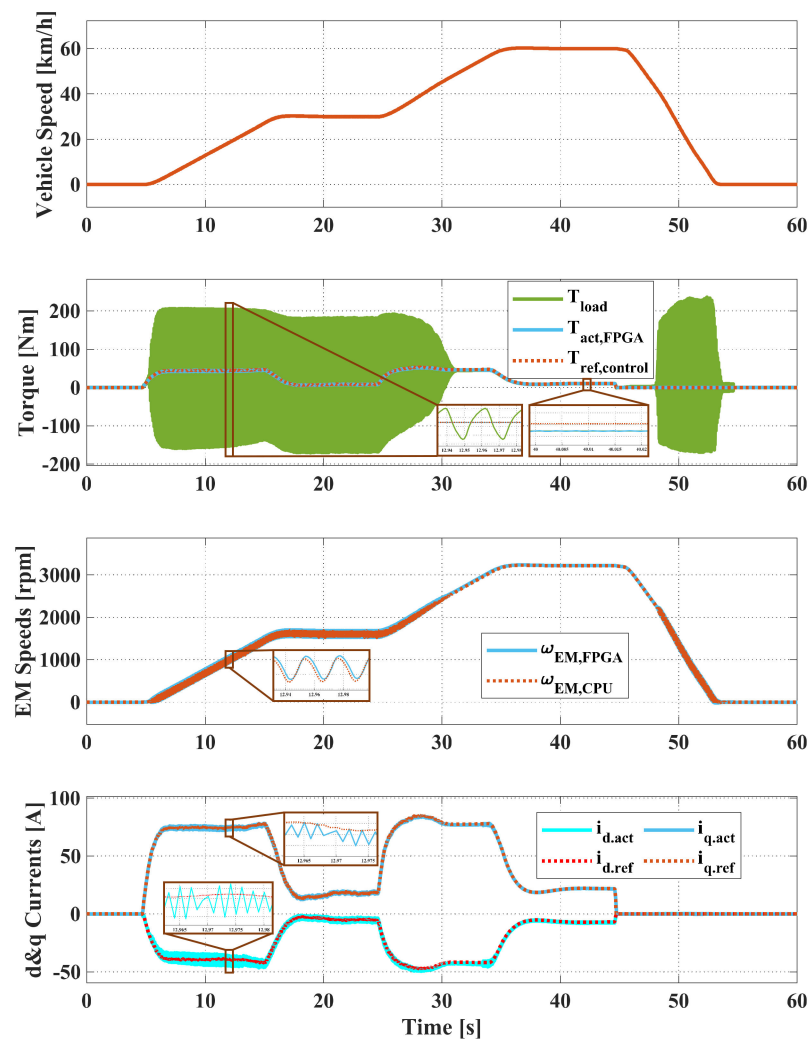


Figure 8. Torque closed-loop real-time simulation results.



**Figure 9.** Speed closed-loop real-time simulation results. Electric machine speed is sent from the vehicle model to the electric drive model.

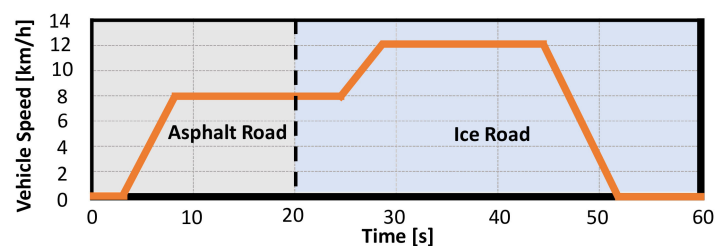
Figure 10 presents results of the closed-loop interface where the rotor speed from the electric drive model was fed back to the vehicle model. Oscillations can be observed in the load torque and the rotor speeds. It was seen in Figure 8 that the rotor speed from the electric drive model was higher than the one calculated in the vehicle model. When this bigger rotor speed was fed back to the vehicle model, the wheel slip calculated according to Equation (7) was also higher. As explained before, the load torque is related to the wheel slip. The increased load torque will reduce the rotor speed in the next time-step. Consequently, the wheel slip will decrease, which will cause the observed oscillations. However, these oscillations do not result in oscillations in the vehicle speed because the vehicle speed is estimated considering both driven and non-driven wheels. The non-driven wheel speeds calculated inside the vehicle model are not affected by the feedback of rotor speed. Furthermore, at low vehicle speeds, a relatively smaller error in the rotor speed can cause a bigger wheel slip. This is why the load torque and the rotor speed oscillations are not observed when the vehicle speed is higher than 50 km/h.



**Figure 10.** Speed closed-loop real-time simulation results. Electric machine speed is sent from the electric drive model to the vehicle model.

#### 4.3. Anti-Slip Real-Time Simulation Results

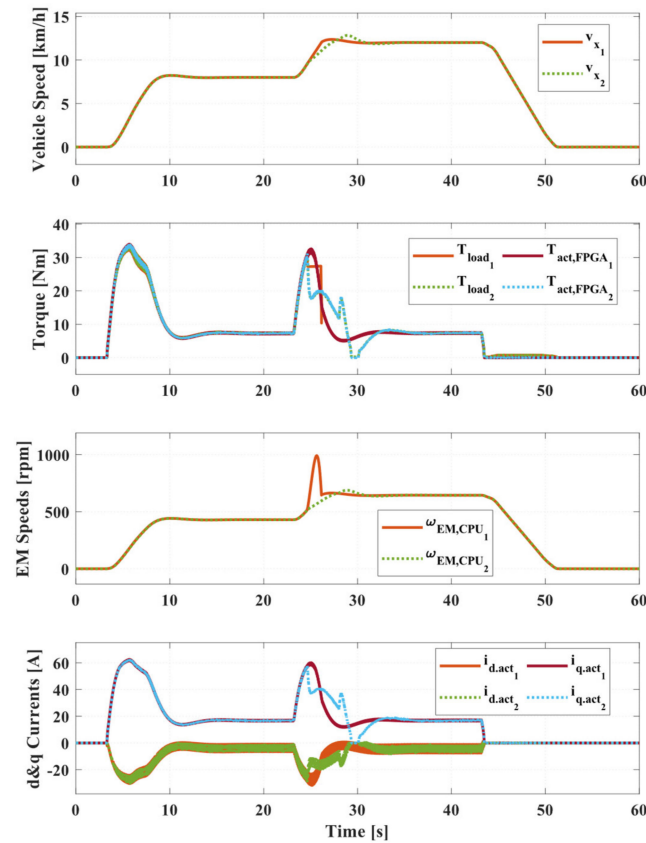
The driving cycle used in the anti-slip real-time simulation is described in Figure 11. The vehicle was accelerated from a standstill on a normal asphalt road to 8 km/h. It cruised at this speed and entered an ice road with a friction coefficient of 0.1. Then the vehicle was accelerated to 12 km/h, followed by constant speed driving. Finally, the vehicle was decelerated to zero speed. The closed-loop interface with rotor speed feedback from the vehicle model to the electric drive model was used.



**Figure 11.** Driving cycle used in anti-slip real-time simulation.

The simulation results are shown in Figure 12. The first subplot presents vehicle speeds from the vehicle model. It is followed by a torque subplot, where electromagnetic

and load torque are shown. The third subplot compares the results to the rotor speed from the vehicle model. The final subplot shows the actual d&q axis currents. The subscripts “1” and “2” used in the legends refer to results without and with the anti-slip function enabled, respectively.



**Figure 12.** Anti-slip real-time simulation results. Subscripts “1” and “2” indicate without and with anti-slip function, respectively.

When the vehicle is accelerating on the ice road, the load torque firstly increases with the reference torque. If the anti-slip function is not activated, the load torque will drop suddenly when the maximum traction force is reached. Afterward, the maximum load torque is maintained, while the reference torque keeps increasing. This will cause rotor/wheel speed to increase rapidly, which can be observed through the orange line in subplot 3. In this case, the driven wheel slip also increases largely according to Equation (7), which means the vehicle is losing control since the available lateral force on driven wheel tires will be much lower. The load torque comes back to the normal range when the reference torque is reduced and the vehicle starts cruising. If the anti-slip function is activated, a part of the reference torque is removed when big wheel slips are detected. Correspondingly, the load torque follows the reduced reference torque and does not exceed its maximum value. As a result, the driven wheels can follow the vehicle speed. A small overshoot of the vehicle speed at the end of the acceleration period can be observed, which was caused by the release of the anti-slip control. The d&q currents in subplot 4 are also correlated to the corresponding electromagnetic torque.

## 5. Experiment Verification

### 5.1. Experimental Setup

The FPGA-based electric drive model was verified using a back-to-back electric machine test bench, as shown in Figure 13. The test bench has two identical PMSMs driven by two IGBT-based two-level voltage source inverters. The parameters of PMSMs, inverters,



and DC power supply were the same as the parameters used for modeling, which are listed in Table 1. The two PMSMs were connected via a shaft. One worked as a traction machine, while the other worked as a load machine. The DC links of the two inverters were connected together to the same DC power supply. This connection allowed the power to recirculate in the system. Hence, only the losses in the experimental setups needed to be supplied, putting less requirement on the DC power supply.

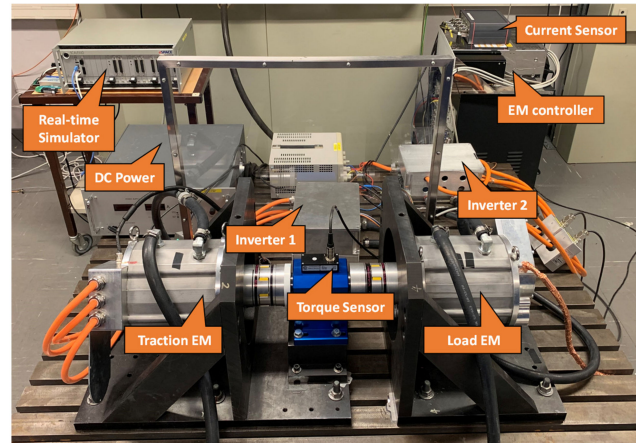


Figure 13. Experimental setup of back-to-back electric machine test bench.

A cascaded control strategy was used, which has the speed/torque control in the outer loop and the field-oriented current control in the inner loop. The speed controller bandwidth was 20 rad/s, and the current controller bandwidth was 400 rad/s. As with the real-time simulations, the MTPA method was used to find the reference d&q axis currents based on the reference torque. One digital torque sensor was installed, which enabled the shaft torque to be recorded during experiments.

During experiments, the real-time simulation was also running in parallel. The closed-loop interface with the speed feedback from the vehicle model to the electric drive model was used, as shown in Figure 14. The performance of one FPGA-based electric drive model was compared with results from the test bench. The reference torque from the vehicle model was used for torque control of the traction machine, while the electric machine speed from the vehicle model was used for speed control of the load machine. Both signals from the real-time simulator were sent to the machine controllers in the test bench via A/D channels. The same driving cycle as described in the anti-slip real-time simulation was used. However, only the tests when the anti-slip function was not activated was performed here.

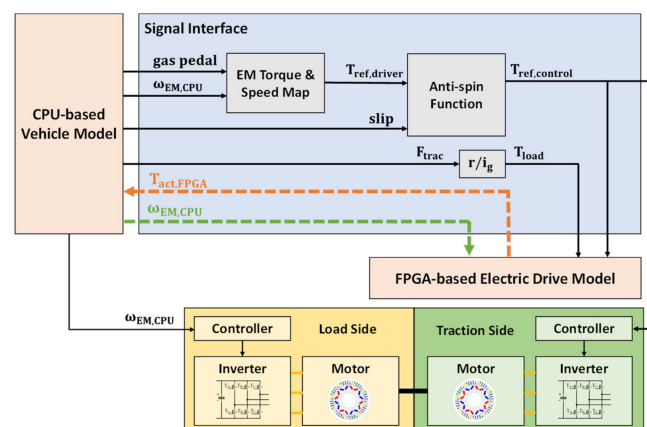


Figure 14. Experimental interface between the real-time simulator and the test bench.



## 5.2. Experimental Results

The results shown in Figure 15 present the comparison of torque, rotor speed, and d&q currents between the real-time simulation and the experiment. The electromagnetic torque from the electric drive model is compared with the shaft torque in the bench. It can be observed that the experimental torque is much smaller at the start of the acceleration. This is due to the limitation of speed control implemented in the load machine, which does not allow a low speed ramp-up when torque is applied on the shaft. Therefore, a reduction of the reference torque is applied during the initial stage of acceleration. This reduction can also reflect on the d&q currents at the beginning phase in subplot 3. Furthermore, since the mechanical losses of the electric machine are not modeled, the real torque on the shaft is smaller even after the starting period. When calculating the current references using MTPA, the same FEM model of the machine was used in both the experiment and the simulation. Therefore, the d&q current results match well after 8 s. In subplot 2, the electric machine speeds in the experiment and the simulation are quite close during steady-state and deceleration. However, during dynamics, a speed mismatch was caused by the speed controller of the load machine

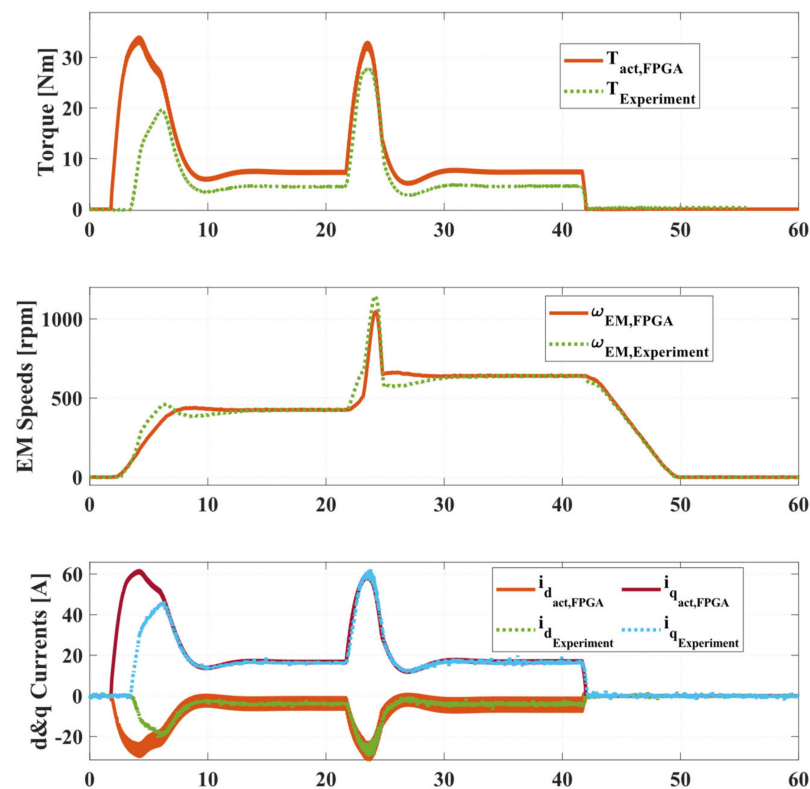


Figure 15. Comparison of electric machine performance between experiment and simulation.

## 6. Conclusions

A real-time simulation platform is proposed in this paper. The platform includes a CPU-based vehicle model and an FPGA-based high-fidelity electric drive model. Both the performance of vehicle control functions and electric quantities inside the electric drive can be analyzed simultaneously. A detailed investigation was carried out to interface two models solved with two types of processors. One open-loop and three different closed-loop interfaces were presented in this paper. It was observed that the open-loop interface causes electric machine speed to diverge in the two models. When using the closed-loop interface with only the torque signal, the speeds between the two models follow each other but are slightly different. On the other hand, when using the closed-loop interface with speed feedback from the FPGA-based electric drive model to the CPU-based vehicle model, large

oscillations in torques and speeds occur at low vehicle speeds. On the contrary, speed feedback from the CPU-based vehicle model to the FPGA-based electric drive model was found to be the best closed-loop interface. The vehicle model is built with CarMaker from IPG Automotive, and the electric drive model is built with XSG from Xilinx. It is the first time that these two simulation tools are integrated together to conduct full-electric vehicle real-time simulation. The proposed platform can be used to perform HIL testing of advanced vehicle control functions, and it was demonstrated by implementing the vehicle anti-slip function in this paper. Finally, a back-to-back electric machine test bench was operated to represent one electric drive performance. The comparison of torque, speed, and d&q currents between all experimental and simulation results shows good agreement.

**Author Contributions:** Conceptualization, B.J., N.S. and Y.L.; Data curation, B.J. and N.S.; Formal analysis, B.J.; Funding acquisition, Y.L.; Investigation, B.J., N.S. and Y.L.; Methodology, B.J., N.S. and Y.L.; Project administration, Y.L.; Resources, B.J., N.S. and X.H.; Software, B.J., N.S. and C.L.; Supervision, Y.L.; Validation, B.J. and N.S.; Visualization, B.J.; Writing—original draft, B.J.; Writing—review and editing, B.J., N.S. and Y.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work has received funding from the European Union’s Horizon 2020 research and innovation program under grant agreement No. 769989. The opinions expressed in this document reflect only the authors’ view and reflect in no way the European Commission’s opinions. The European Commission is not responsible for any use that may be made of the information it contains.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Khaligh, A.; Li, Z. Battery, Ultracapacitor, Fuel Cell, and Hybrid Energy Storage Systems for Electric, Hybrid Electric, Fuel Cell, and Plug-In Hybrid Electric Vehicles: State of the Art. *IEEE Trans. Veh. Technol.* **2010**, *59*, 2806–2814. [\[CrossRef\]](#)
2. Sundaravadivelu, K.; Shantharam, G.; Prabakaran, P.; Raghavendra, N. Analysis of vehicle dynamics using co-simulation of AVL-CRUISE and Car, Maker in ETAS RT environment. In Proceedings of the 2014 International Conference on Advances in Electrical Engineering (ICAEE), Vellore, India, 9–11 January 2014; pp. 1–4.
3. Lee, J.S.; Choi, G. Modeling and hardware-in-the-loop system realization of electric machine drives—A review. *CES Trans. Electr. Mach. Syst.* **2021**, *5*, 194–201. [\[CrossRef\]](#)
4. Tavana, N.R.; Dinavahi, V. Real-Time Nonlinear Magnetic Equivalent Circuit Model of Induction Machine on FPGA for Hardware-in-the-Loop Simulation. *IEEE Tran. Energy Convers.* **2016**, *31*, 520–530. [\[CrossRef\]](#)
5. Mudrov, M.; Ziuzev, A.; Nesterov, K.; Valtchev, S. Status and development prospects of real-time systems (RTS) in the electric drive. In Proceedings of the 2020 XI International Conference on Electrical Power Drive Systems (ICEPDS), St. Petersburg, Russia, 4–7 October 2020; pp. 1–5.
6. Mojlish, S.; Erdogan, N.; Levine, D.; Davoudi, A. Review of Hardware Platforms for Real-Time Simulation of Electric Machines. *IEEE Trans. Transp. Electr.* **2017**, *3*, 130–146. [\[CrossRef\]](#)
7. Saad, H.; Ould-Bachir, T.; Mahseredjian, J.; Dufour, C.; Dennetiere, S.; Nguefeu, S. Real-Time Simulation of MMCs Using CPU and FPGA. *IEEE Trans. Power Electron.* **2013**, *30*, 259–267. [\[CrossRef\]](#)
8. Navabi, Z. *Embedded Core Design with FPGAs*; McGraw-Hill: New York, NY, USA, 2007.
9. Tola, S.; Sengupta, M. Real-time simulation of an induction motor in different reference frames on a FPGA platform. In Proceedings of the 2012 IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES), Bengaluru, India, 16–19 December 2012; pp. 1–6.
10. Chen, H.; Sun, S.; Aliprantis, D.C.; Zambreno, J. Dynamic simulation of electric machines on FPGA boards. In Proceedings of the 2009 IEEE International Electric Machines and Drives Conference, Miami, FL, USA, 3–6 May 2009; pp. 1523–1528.
11. Williamson, S.S.; Emadi, A.; Rajashekara, K. Comprehensive Efficiency Modeling of Electric Traction Motor Drives for Hybrid Electric Vehicle Propulsion Applications. *IEEE Trans. Veh. Technol.* **2007**, *56*, 1561–1572. [\[CrossRef\]](#)
12. Awadallah, M.; Tawadros, P.; Walker, P.; Zhang, N. Hardware-in-the-Loop Simulation for the Design and Testing of Motor in Advanced Powertrain Applications. In Proceedings of the 2018 IEEE 27th International Symposium on Industrial Electronics (ISIE), Cairns, QLD, Australia, 13–15 June 2018; pp. 817–824.

13. Queiroz, J.; Carvalho, S.; Barros, C.; Barros, L.; Barbosa, D. Embedding an Electrical System Real-Time Simulator with Floating-Point Arithmetic in a Field Programmable Gate Array. *Energies* **2021**, *14*, 8404. [[CrossRef](#)]
14. Stunda, M.; El Baghdadi, M.; Ribickis, L.; Hegazy, O. Induction Machine HiL Setup Implementation and IFOC Testing Based on the OP4510 Realtime Computer. In Proceedings of the 2020 IEEE 61th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON), Riga, Latvia, 5–7 November 2020; pp. 1–5.
15. Moldovan, T.; Inte, R.; Nemes, R.-O.; Ruba, M.; Martis, C. Typhoon HiL Real-Time Validation of Permanent Magnet Synchronous Motor's Control. In Proceedings of the 2021 9th International Conference on Modern Power Systems (MPS), Cluj-Napoca, Romania, 16–17 June 2021; pp. 1–6.
16. Etzold, K.; Fahrbach, T.; Klein, S.; Scheer, R.; Guse, D.; Klawitter, M.; Pischinger, S.; Andert, J. Function Development With an Electric-Machine-in-the-Loop Setup: A Case Study. *IEEE Trans. Transp. Electr.* **2019**, *5*, 1419–1429. [[CrossRef](#)]
17. Abdelrahman, A.S.; Algarny, K.S.; Youssef, M.Z. A Novel Platform for Powertrain Modeling of Electric Cars With Experimental Validation Using Real-Time Hardware in the Loop (HiL): A Case Study of GM Second Generation Chevrolet Volt. *IEEE Trans. Power Electron.* **2018**, *33*, 9762–9771. [[CrossRef](#)]
18. Kali, Y.; Saad, M.; Bouchama, A.; Dehbozorgi, R.; Paquin, J.-N.; Gregoire, L.-A.; Belanger, J.; Rodas, J. HiL Simulation of On-line Parameters Estimation and Current Control of a Six-Phase Induction Machine using OPAL-RT Technologies. In Proceedings of the 2020 IEEE Power & Energy Society General Meeting (PESGM), Montreal, QC, Canada, 2–6 August 2020; pp. 1–5.
19. Yang, B.; Guo, L.; Ye, J. Real-time Simulation of Electric Vehicle Powertrain: Hardware-in-the-Loop (HiL) Testbed for Cyber-Physical Security. In Proceedings of the 2020 IEEE Transportation Electrification Conference & Expo (ITEC), Chicago, IL, USA, 23–26 June 2020; pp. 63–68.
20. Ruan, J.; Song, Q. A Novel Dual-Motor Two-Speed Direct Drive Battery Electric Vehicle Drivetrain. *IEEE Access* **2019**, *7*, 54330–54342. [[CrossRef](#)]
21. Lee, C.; Hedrick, K.; Yi, K. Real-Time Slip-Based Estimation of Maximum Tire–Road Friction Coefficient. *IEEE/ASME Trans. Mechatron.* **2004**, *9*, 454–458. [[CrossRef](#)]