



RT Box

DEMO MODEL

Sensorless Vector Control for Permanent Magnet Synchronous Machine

Cascaded speed and current control with an observer based position and speed estimation

Last updated in RT Box Target Support Package 2.1.5

www.plexim.com

- ▶ Request a PLECS and PLECS Coder trial license
- ▶ Get the latest RT Box Target Support Package
- ▶ Check the PLECS and RT Box documentation

1 Overview

This RT Box demo model features a drive system with a permanent magnet synchronous machine (PMSM). The following sections describe in detail the implementation of the power stage and controls using the PLECS Electrical and Control domains. This demo model has the following features:

- The drive outputs a nominal torque of 0.11 Nm and is fed by a 24 V DC voltage source.
- Explanation how to convert data sheet parameters of a PMSM to use in the PLECS PMSM model.
- The implementation of a rotor position and speed observer for sensorless field oriented control strategy.
- A closed-loop cascaded controller with inner d- and q- axis current controller and outer speed controller.
- The model is split into two distinct subsystems called “Plant” and “Controller”. This allows the same model to be used in an offline simulation in PLECS and for real-time simulations on the PLECS RT Box. These subsystems can then be independently built on the PLECS RT Box either for Hardware-in-the-loop (HIL) testing of an external controller, or for rapid control prototyping (RCP). The two RT Boxes are connected for a complete closed-loop simulation.

The following sections provide a detailed description of the model and instructions on how to simulate it.

The execution time represents the actual time it takes to execute one calculation step of the PLECS model on the RT Box hardware. The chosen discretization step sizes and average execution times for each subsystem in this demo model are shown in Fig. 1.

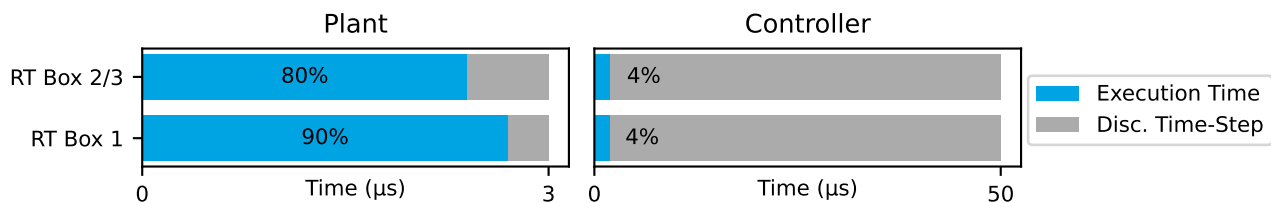


Figure 1: Performance overview for the execution on two RT Boxes

1.1 Requirements

To run this demo model, the following items are needed (available at www.plexim.com):

- Two PLECS RT Boxes and one PLECS and PLECS Coder license
- The RT Box Target Support Package
- Follow the step-by-step instructions on configuring PLECS and the RT Box in the Quick Start guide of the RT Box User Manual.
- Three 37 pin Sub-D cables to connect the boxes front-to-front.

Note that this demo model is targeted at two RT Boxes application, with one running the Plant and the other running the Controller. In this way, the execution time of each real-time target is minimized. Besides, the setup can easily transition to a HIL or RCP test later on.

However, if the user has only one RT Box available, please check the corresponding models targeted for one RT Box application. In this case, two 37 pin Sub-D cables are still needed to connect in front Analog Out interface with Analog In interface, and Digital Out interface with Digital In interface.

- For RT Box 2 and 3, by default the multi-tasking feature is enabled in this demo. “Controller” part is circled with a Task frame block, and runs in one core. The rest of the circuit on the schematic belongs to the “Base task”, and runs in another core. In this way the computational effort is split onto different cores. Please check the default setting under **Scheduling** tab of the **Coder options...** window.

- For RT Box 1, single-tasking should be used which runs both the Plant and the Controller on the same one core of the RT Box 1.

2 Model

The top level schematic of the demo model is depicted in Fig 2. The model is split into two subsystems: “Plant” and “Controller”. Both subsystems are enabled for code generation by right-clicking on the component and selecting **Enable code generation** check box from the **Subsystem + Execution settings...** menu. This allows the model to be simulated and the controls to be verified offline in PLECS before a real time simulation on the RT Box target.

The inherit delays of the closed-loop control are modelled in a simplified way in the offline simulation using a Delay block (z^{-1}) with sample time $T_{s_control}$.

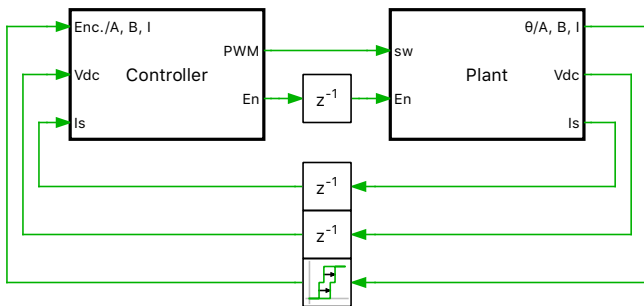


Figure 2: Top level schematic of the demo model sensorless speed controlled PMSM

2.1 Plant

The power circuit includes an PMSM and a three-phase full bridge voltage source inverter (VSI). The DC voltage source V_i with $V_{dc} = 24\text{ V}$ supplies the VSI, which is represented by three Mosfets Half Bridge power modules.

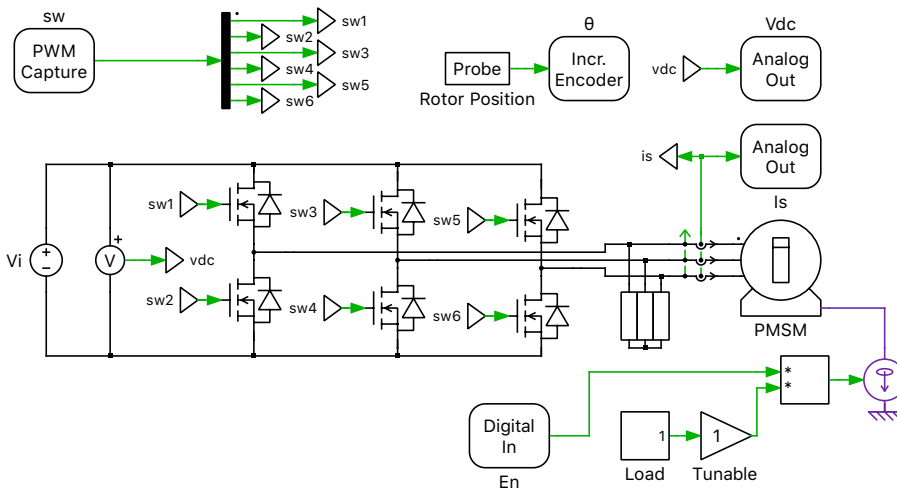


Figure 3: Power circuit of the PMSM drive system

The six switching signals are fed into the subsystem by a PWM Capture block from the PLECS RT Box component library. The measurements of the DC voltage and the AC current output by Analog Output blocks. The rotor angular position is converted by the Incremental Encoder block into digital orthogonal pulses, which can be measured outside of “Plant” subsystem.

The rotor inertia absorbs energy when the motor is accelerating and thus provides a load when starting up or changing speed. For this purpose the value from data sheet of Tab. 1 was used in the PMSM block, as internal mechanical load. For closed loop startup, the load cannot be higher than 40 %, otherwise the observer will not be able to track the speed in time and the control will become unstable. To demonstrate a load step, a load profile alternates between 20 % and 40 % of the nominal motor torque. The load is not applied until the controller is switched on.

Machine Parameters

The machine parameters are listed in Tab. 1 which were taken from the data sheet. The following parameters must be converted in order to use them in the PLECS model:

- Line to line quantities must be converted in line to neutral quantities such as stator resistance R and stator inductance L_d and L_q . The conversion factor is $\sqrt{3}$.
- The permanent magnet flux linkage ψ can be calculate from line to neutral back EMF voltage e_{ln} . Which is the peak voltage induced by the permanent magnets in each phase per unit rotational speed. The relation can be expressed as follows:

$$e_{ln} = p\psi\omega \quad (1)$$

In Tab. 1, the back EMF voltage e_{ln} was measured at a speed of 1000 revolution per minute (rpm). However, the PLECS model parameters are in radians per second (rad/s) which requires a conversion.

Note The machine model is chosen as Rotor reference frame. This configuration allows the generation of code for this model, however, it does not allow the simulation of the machine in generator mode together with dead-times on the inverter half-bridges.

Table 1: Specifications of applied PMSM BLWS232D-36V-4000

Number of pole pairs	2
Rated Power (W)	40
Rated Torque (Nm)	0.1
Rated Speed (rpm)	4000
Line to Line Resistance (Ω)	2.4
Line to Line Inductance (mH)	4.39
Rotor Inertia (Nms ²)	0.0000074852
Back EMF Voltage (V/krpm)	4.5

2.2 Controls

In the controller subsystem, the DC-link voltage and the stator currents are measured by Analog In blocks. The mechanical angular speed of the rotor is obtained from the Quadrature Encoder Counter block, which converts the orthogonal digital pulses.

The initial mechanical speed reference value is set at 2000 rpm. In order to guarantee a robust start-up process of the observer, it is always started at a speed of 2000 rpm and regulated to the desired speed after the start-up. The speed of 500 rpm cannot be undercut in connection with the attached

load in order to guarantee stable operation of the observer. Otherwise the observer will not be able to track the speed in time and the control will become unstable.

For the startup scenario a Finite State Machine is used. The system is first started with a current ramp in current control mode in order to align the position of the machine. After the current ramp and a short settling time, the speed control is started. If the drive system is operated with encoder, this startup scenario is omitted. The observer is already set to an initial speed of 500 rpm for a better transient response during startup. For robust system behavior, an abrupt speed change is limited with a rate limiter.

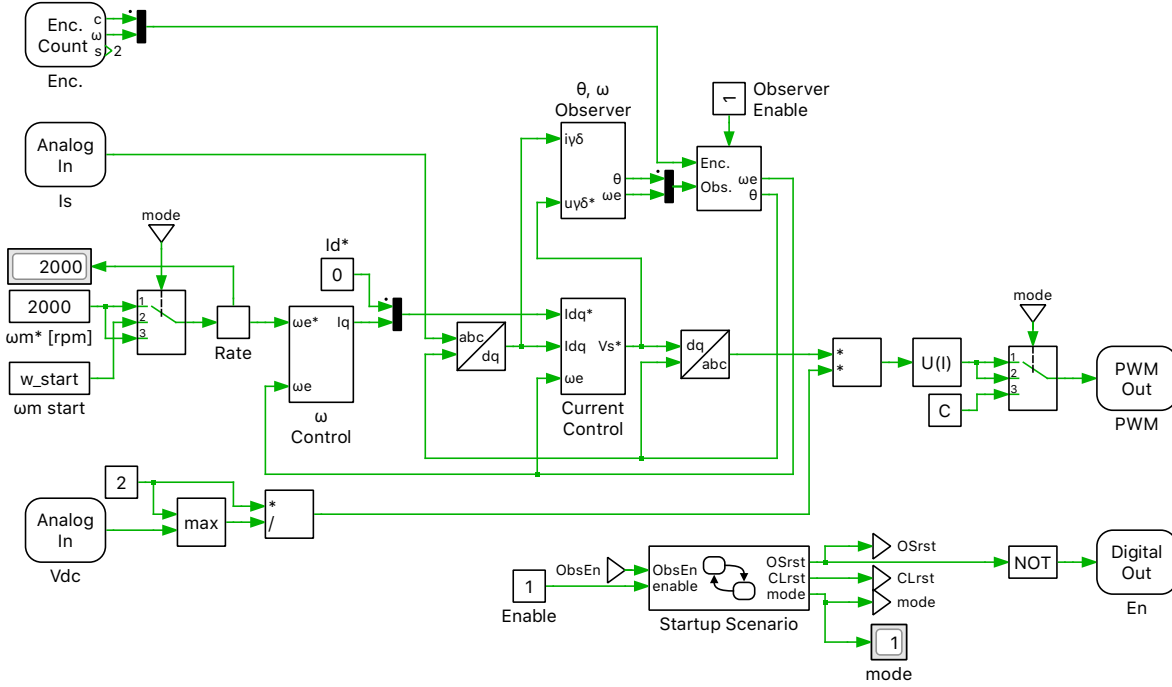


Figure 4: Controller model of the sensorless rotor-field oriented control

Rotor-field oriented control is applied to the drive system and the basic structure is shown in Fig 4, where the stator current is regulated in the dq-frame. The speed and rotor position can be calculated by either using the Observer subsystem or Encoder Counter target block. This selection is made by the “Selector” block, where the observer can be enabled or disabled. A higher-level controller is implemented for speed control, whereby a speed reference in rpm can be set. The complete controller can be enabled or reset via “Enable”. In this case, the model continues to run in open loop with a fixed modulation index. In further investigations where the “Plant” RT box is replaced by real hardware, this Digital Out is needed for enabling and disabling.

The voltages are the control variables and are represented in the dq-frame which can be derived from Eq. 2 while $L_{sd} = L_{sq}$. The currents are controlled with a PI controller, which is explained in the following.

$$\begin{bmatrix} u_d \\ u_q \end{bmatrix} = \begin{bmatrix} R_s & -\omega L_{sq} \\ \omega L_{sd} & R_s \end{bmatrix} \cdot \begin{bmatrix} i_d \\ i_q \end{bmatrix} + \begin{bmatrix} L_{sd} & 0 \\ 0 & L_{sq} \end{bmatrix} \cdot \begin{bmatrix} \frac{d}{dt} i_d \\ \frac{d}{dt} i_q \end{bmatrix} + \begin{bmatrix} 0 \\ \omega \psi_f \end{bmatrix} \quad (2)$$

PI current controller

The d and q axis currents which are controlled by a PI controller are included in subsystem “Current Control”, which is shown in Fig 5. The proportional and integral gains are designed following the “Optimum Magnitude” method, which is described in more detail in the “Boost Converter” demo model of



Rotor position and speed observer

In the observer model the estimated rotating γ - δ frame is used, which differs from the d-q reference frame by the position error θ_e .

The block diagram illustrates the control system for a two-link robot arm. It consists of the following main components and signal flows:

- Reference Inputs:** $iy\delta$ and $uy\delta^*$.
- Feedforward Path:** $iy\delta$ is multiplied by a gain K . $uy\delta^*$ is multiplied by a gain $*$ and then by a derivative block $\frac{d}{dt}$. These signals are summed to produce the reference velocity $\dot{\theta}_e$.
- Observer Block (Blue Dashed Box):** This block contains two parallel observer structures. The top observer uses $\dot{\theta}_e$ and the estimated position $\hat{\theta}$ to calculate the estimated velocity $\dot{\theta}$. The bottom observer uses $\dot{\theta}_e$ and the estimated velocity $\dot{\theta}$ to calculate the estimated position $\hat{\theta}$. Both observers use gain matrices R_s and L_s .
- Control Signal:** The estimated position $\hat{\theta}$ is compared with the reference position θ_e to produce the position error $\theta_e - \hat{\theta}$. This error is passed through a sign block and a gain block ω_0 to produce the control signal ω .
- Plant Dynamics:** The control signal ω is integrated by a $1/s$ block to produce the final output θ . The estimated position $\hat{\theta}$ is also used to calculate the estimated error ω_e .

erence speed is negative. This is necessary during start-up of the drive to ensure correct tracking of the observer. All integrators of the observer can be reset by the “OSrst” signal.

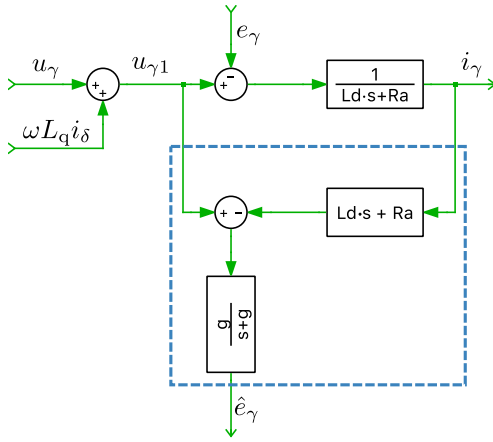


Figure 7: Equivalent block diagram of least-order observer

Algorithm

The machine model used for the sensorless control algorithm is given in $\gamma - \delta$ coordinates including the extended EMF [1]:

$$\begin{bmatrix} u_\gamma \\ u_\delta \end{bmatrix} = \begin{bmatrix} R_s & -\omega L_{sq} \\ \omega L_{sd} & R_s \end{bmatrix} \cdot \begin{bmatrix} i_\gamma \\ i_\delta \end{bmatrix} + \begin{bmatrix} L_{sd} & 0 \\ 0 & L_{sq} \end{bmatrix} \cdot \begin{bmatrix} \frac{d}{dt} i_\gamma \\ \frac{d}{dt} i_\delta \end{bmatrix} + \begin{bmatrix} e_\delta \\ e_\gamma \end{bmatrix} \quad (3)$$

where

$$\begin{bmatrix} e_\delta \\ e_\gamma \end{bmatrix} = E_{ex} \begin{bmatrix} -\sin \theta_e \\ \cos \theta_e \end{bmatrix} + (\hat{\omega} - \omega) L_s \begin{bmatrix} -i_\delta \\ i_\gamma \end{bmatrix}. \quad (4)$$

Assuming the error between estimated speed $\hat{\omega}$ and actual speed ω is sufficiently small, Eq. 4 can be simplified and the second term is eliminated, resulting in Eq. 5.

$$\begin{bmatrix} \hat{e}_\delta \\ \hat{e}_\gamma \end{bmatrix} = E_{ex} \begin{bmatrix} -\sin \hat{\theta}_e \\ \cos \hat{\theta}_e \end{bmatrix} \quad (5)$$

By transforming Eq. 5 position error $\hat{\theta}_e$ can be estimated in two ways, both schemes are implemented in the configurable subsystem “ θ_e error”:

Scheme A:

$$\hat{\theta}_e = \tan^{-1} \left(-\frac{\hat{e}_\gamma}{\hat{e}_\delta} \right) \quad (6)$$

Scheme B:

$$\hat{\theta}_e \approx \left(-\frac{\hat{e}_\gamma}{\hat{E}_{ex}} \right) \quad (7)$$

A PI compensator $G_e(s)$ is used to compensate the estimated speed $\hat{\omega}$ and estimated position $\hat{\theta}$, so that the position error $\hat{\theta}_e$ becomes zero. During acceleration the estimated position error can be reduced using a 3. order instead of a 2. order compensator. Both are implemented in a configurable subsystem.

$$G_e(s) = K_1 - \frac{K_2}{s} + \frac{K_3}{s^2} \quad (8)$$

The coefficients are designed with a natural frequency of $\omega_n = 70 \text{ rad/s}$ and a damping factor of $\zeta_n = 1.5$ for the 2. order compensator according to the following rule: $K_1 = 2\zeta_n\omega_n$, $K_2 = \zeta_n\omega_n^2$. The 3. order compensator is designed according to the following rule: $s^3 + K_1s^2 + K_2s + K_3 = (s + \omega_n)(s^2 + 2\zeta_n\omega_ns + \omega_n^2)$. The cutoff frequency of the low-pass filter for the speed estimator is selected as 1000 rad/s .

Speed Controls

To control the drive speed a *outer* speed control loop is provided. The time constants of the *outer* control loop is dictated by mechanical time constants, which is larger on comparison to the *inner* current control loop. The proportional-integral speed controller can be described as follows

$$\frac{T_e^*}{\Delta\omega_m} = K_p \left(1 + \frac{1}{\tau_i s} \right) \quad (9)$$

T_e^* defines the reference torque and $\Delta\omega_m$ is defined as $\Delta\omega_m = \omega_m^* - \omega_m$. The controller is represented in terms of a proportional gain K_p and integral time constant τ_i . According to [2] page 117 ff. for the closed loop transfer function of the speed controller the bandwidth is effectively defined by the bandwidth frequency ω_B^{sp} , while the proportional gain is given as

$$K_p = \omega_B^{sp} J \quad (10)$$

The inertia J of the total mechanical load must be taken into account additionally for designing the control parameters. For this purpose, the total combined inertia must be estimated.

According to [2] $\omega_B^{sp} = 100 \text{ rad/s}$ is a typical value for the effective speed control bandwidth in high-performance drives. In this demo model, however, a lower value of $\omega_B^{sp} = 70 \text{ rad/s}$ was assumed due to the additional delay of the observer. The value ω_B^{sp} , together with the inertia J , fully defines the proportional gain of the controller.

For a damping factor of $\zeta = 1$ and the already defined K_p , the integral time constant τ_i is determined as follows:

$$\tau_i = \frac{4}{\omega_B^{sp}} \cdot \quad (11)$$

The manipulated variable of the speed controller is the reference torque T_e^* , which in turn is the set-point of the inner current controller loop. Therefore a conversion from mechanical torque T_e to i_q current has to be done regarding Eq. 12. The values for flux linkage Ψ_m of the permanent magnets and number of rotor permanent magnet pole pairs p are needed additionally.

$$T_e = \frac{3}{2} \Psi_m i_q p \quad (12)$$

To perform this conversion the gain block with the coefficient K is used.

$$i_q^* = \frac{T_e^*}{K} \quad (13)$$

3 Simulation

This model can run both in offline mode on a computer or in real-time mode on the PLECS RT Box. For the real-time operation, two RT Boxes (referred to as “Plant” and “Controller”) need to be set up as demonstrated in Fig. 8 using three DB37 cables.

Please follow the instructions below to run a real-time model on two RT Boxes:

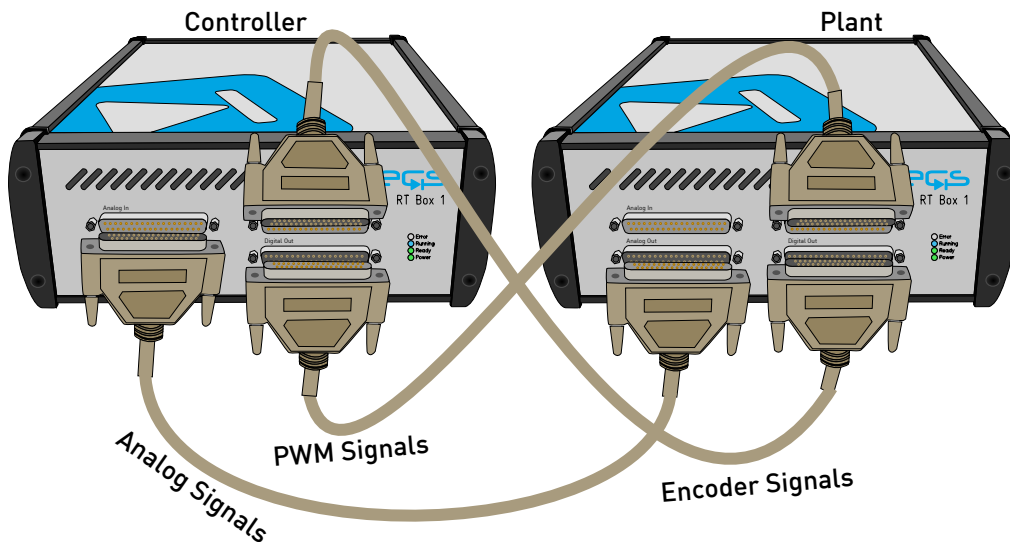


Figure 8: Hardware configuration for the real-time operation of the demo model

- From the **System** tab of the **Coder options...** window, select the “Plant” and **Build** it onto the “Plant” RT Box. Then, select “Controller” and **Build** it onto the “Controller” RT Box. If this sequence is not followed, the control will become unstable and the controller must be reset via “Enable” to force a restart.
- Once the models are uploaded, from the **External Mode** tab of the **Coder options...** window, **Connect** to both RT Boxes and **Activate autotriggering**.

3.1 Transient Scenarios

The load profile alternates between 0.04 Nm and 0.06 Nm with a nominal torque of 0.1 Nm, which is shown in Fig 9. The required torque can be compensated through the current controller in approx. 0.05 s. The drop in speed, which is approx. 200 rpm, can be compensated from the speed controller within 0.5 s.

Step change of speed reference

The stator current, rotational angle and speed are shown in the scope of the “Controller” subsystem. To observe a transient behavior of the system, e.g. a step change of the speed reference from 1000 rpm to 2000 rpm, please further follow the scenario below:

- Make sure that the External Mode and **Activate autotriggering** of both RT Boxes are enabled.
- Switch the **Trigger channel** parameter to [Observer] in the **External Mode** tab of the “Controller” subsystem’s **Coder options...** window.
- Setup the **Trigger level** parameter to be 1500 and **Trigger delay [steps]** to be -4000.
- Open the controller subsystem and change the Constant block “ ω_m^* ” from the default value of 1000 to 2000.

The step change will be captured by the scope in the “Controller” subsystem, as shown in Fig. 10. A comparison between the measured angle and speed from the encoder and the estimated angle and speed from the observer is shown. Nevertheless, the measured and estimated speed match each other very well. When the speed reference changes rapidly, the position estimation error and the speed estimation error are caused. However, this error is very small and has very little effect on the motor control. At speeds below 1000 rpm, a larger speed ripple becomes visible. In general, the observer works better for higher speeds, but this effect is additionally amplified in lower speed ranges by a blanking time.

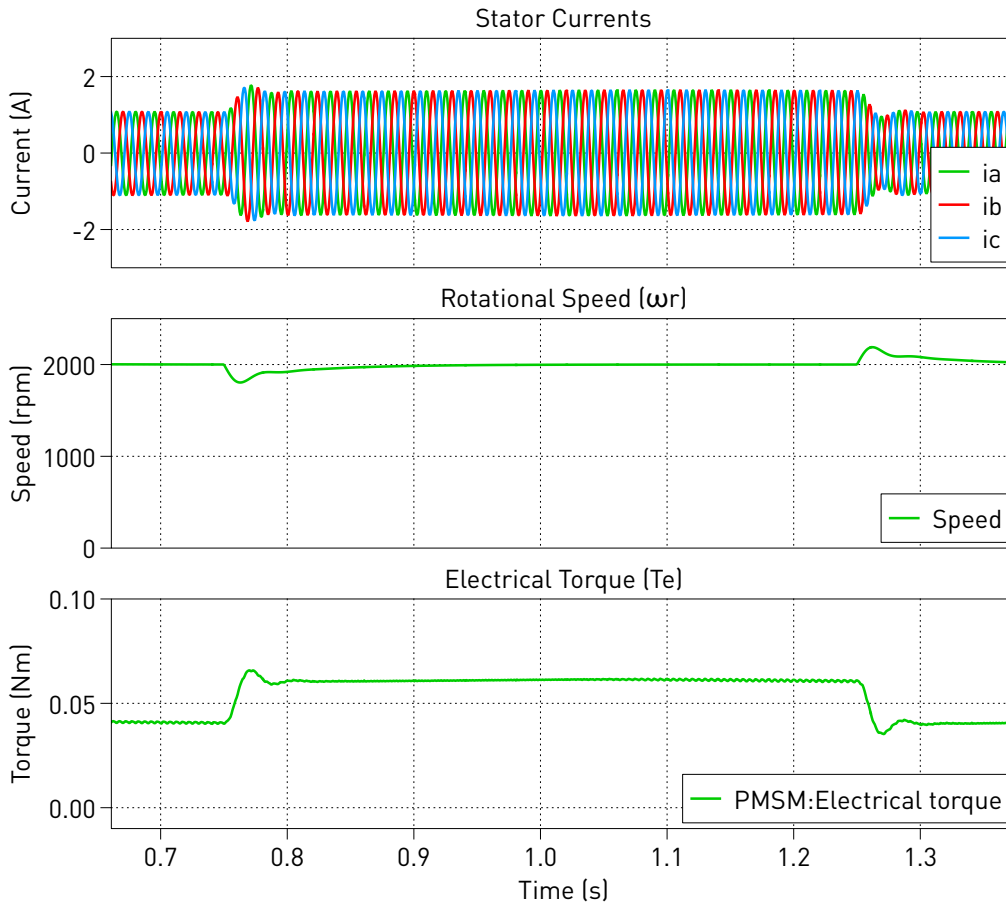


Figure 9: Transient response of the speed controller for a load torque step from 0.01 Nm to 0.02 Nm (offline simulation results).

Note The literature reference [1] does not describe a specific startup procedure for this sensorless control algorithm. However, the system start-up procedure is especially delicate when using sensorless control schemes. This demo model uses a simple open-loop start-up procedure to ramp up the controls and system operation but does not go into further details. This means depending on the exact timing of events in a real-time simulation the control may fail to start up properly. In such a case the controller needs to be restarted.

3.2 Conclusions

This model demonstrates a PMSM drive system which can run in both offline simulation and real-time operation for Hardware-in-the-loop testing and rapid control prototyping. Rotor-field oriented control is applied to the drive system where the stator current is regulated in the dq-frame. A higher-level controller is implemented for speed control, whereby an observer is used to track the electrical angular velocity and rotor position.

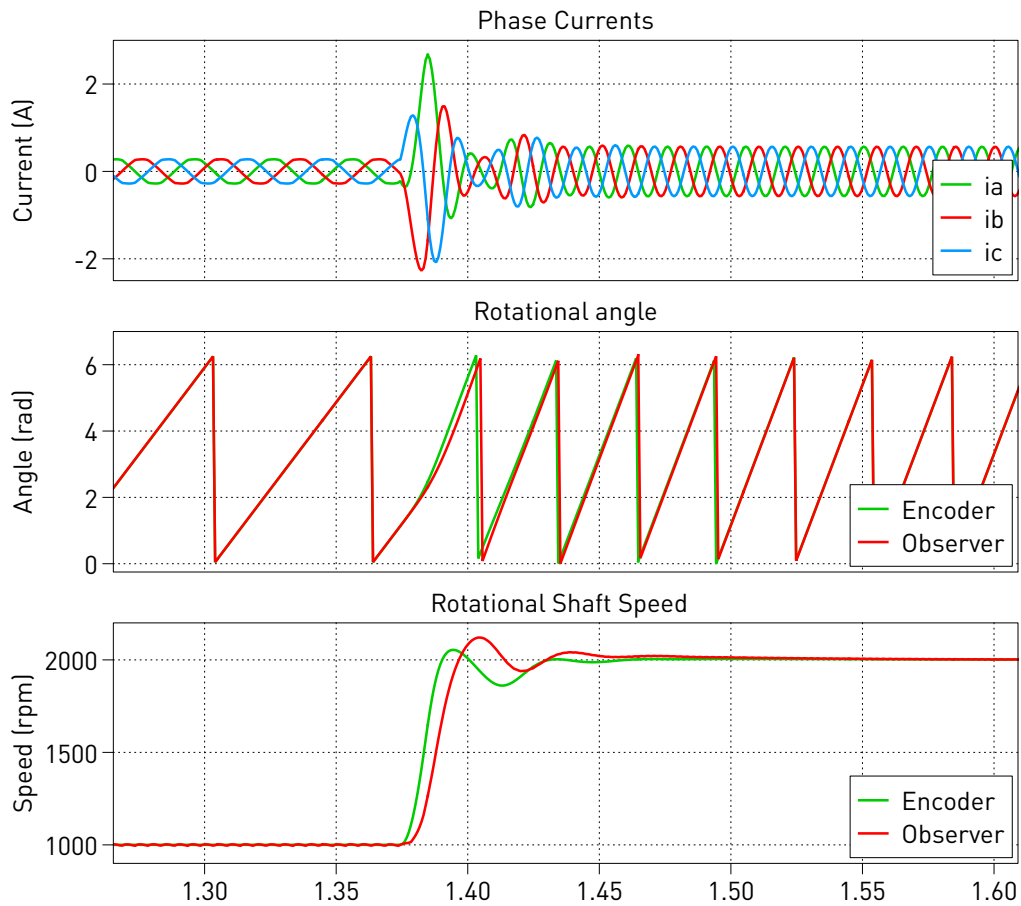


Figure 10: Transient response of speed controller for a change in speed reference from 1000 RPM to 2000 RPM (offline simulation results).

References

- [1] S. Morimoto, K. Kawamoto, M. Sanada and Y. Takeda, "Sensorless control strategy for salient-pole PMSM based on extended EMF in rotating reference frame", IEEE Trans. Ind. Appl., vol. 38, no. 4, pp. 1054-1061, Jul./Aug. 2002.
- [2] R. De Doncker, D. Pulle and A. Veltman, "Advanced electrical drives", Springer, 2011.

Revision History:

RT Box Target Support Package 2.1.5

First release

How to Contact Plexim:

☎	+41 44 533 51 00	Phone
	+41 44 533 51 01	Fax
✉	Plexim GmbH Technoparkstrasse 1 8005 Zurich Switzerland	Mail
@	info@plexim.com	Email
	http://www.plexim.com	Web

RT Box Demo Model

© 2002–2022 by Plexim GmbH

The software PLECS described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from Plexim GmbH.

PLECS is a registered trademark of Plexim GmbH. MATLAB, Simulink and Simulink Coder are registered trademarks of The MathWorks, Inc. Other product or brand names are trademarks or registered trademarks of their respective holders.