Zynq Implemented Luenberger Disturbance Observer Based Predictive Control Scheme for PMSM Drives

Long He, Fengxiang Wang, *Senior Member, IEEE*, Junxiao Wang, *Member, IEEE*, and José Rodríguez, *Fellow, IEEE*

***Abstract*—A Luenberger disturbance observer based control scheme for surface mounted permanent magnet synchronous motor (SPMSM) is proposed in this paper. First, an extended SPMSM model is given by considering the external disturbances and parameter mismatches. Second, a Luenberger observer is introduced to estimate the lumped disturbances in the speed and current loops, respectively. Third, based on the observed disturbances, a speed controller is designed, which is proved to be stable. Finally, an improved deadbeat-based predictive current control (DPCC) based on the estimated disturbances and the new SPMSM model is designed to control the current loop. The proposed method is implemented on a Xilinx Zynq SoC XC7Z020-CLG484-1 and field programmable gate array (FPGA) implementations of the improved DPCC and conventional DPCC are compared and analyzed in detail in terms of area utilization and time consumption. Although the execution time of both methods is very short, the improved DPCC takes less time than the conventional DPCC due to the parallel processing capabilities of FPGAs. The experimental results verify that the proposed method has good dynamic performance, load disturbance suppression performance, and parameter robustness.**

***Index Terms*—Deadbeat-based predictive current control (DPCC), Luenberger disturbance observer, permanent magnet synchronous machine (PMSM), Vivado High-Level Synthesis (HLS), Zynq SoC.**

I. Introduction

PERMANENT magnet synchronous motor (PMSM) has been extensively used in a lot of applications such as robotics, electric vehicles, and computerized numerical control machines, due to its superior features: high efficiency, compact structure, and high torque-to-inertia ratio [1], [2]. Field oriented control (FOC) has been widely applied in the PMSM driver to obtain better performance. There are external speed loop and internal current loop in a common FOC strategy.

Besides conventional proportional integral (PI) control, many speed control algorithms including sliding mode control (SMC) [3], two-degrees-of-freedom PI speed control [4], fuzzy control [5], [6], and predictive functional control (PFC) [2] have been proposed so far. For good disturbance rejection performance, the disturbance observer has been widely used in various motor control systems. By introducing disturbance observer into feed-forward compensation, better disturbance rejection ability is obtained [7], [8]. SMC is applied to speed control in [9]. Combining with extended sliding mode disturbance observer, better speed tracking performance is achieved. PFC+ESO (ESO: extended state observer) method is introduced into the control design of speed loop in [2], which is verified in the simulation and experiment.

Many advanced methods have been applied to the current loop as well, such as PI control [10], predictive current control [11], [12], hysteresis current control [13], etc. The hysteresis current control has a fast current response and small tracking error, but the switching frequency is not fixed and the accuracy of current tracking is closely related to the hysteresis width. As for the PI control, its performance deteriorates when the PMSM parameters change. Model-based predictive current control (MPCC) [14], [15] and deadbeat-based predictive current control (DPCC) [16], [17] are two kinds of predictive current control. MPCC has strong robustness, but requires large calculate efforts and additional hardware. Based on the discrete PMSM model, the DPCC calculates voltage references directly, which is transformed into switching signals through the pulsewidth modulation (PWM) technology [3], [18]. The switching frequency of the DPCC is fixed, and it has good current tracking performance and fast current dynamic response. However, the performance of the DPCC depends entirely on the parameters of given model, which are inaccurate and constantly changing in the actual system. A robust DPCC based on the weight modification algorithm is introduced in [19] to improve the system stability. A nonlinear disturbance observer is proposed in [7] to enhance the system robustness. A sliding mode disturbance observer based on a novel reaching law is proposed to estimate the parameter variations and compensate the voltage reference calculated by the DPCC in [18].

The PMSM control algorithms are traditionally implemented on digital signal processors (DSPs) or microcontroller units (MCUs) [20], [21] where plenty of peripherals specially designed for motor control are integrated. However, DSP or MCU

Manuscript received February 1, 2019; revised April 11, 2019; accepted May 27, 2019. Date of publication June 2, 2019; date of current version November 12, 2019. This work was supported in part by National Natural Science Funds of China under Grants 51877207 and 61803335, in part by the Science and Technology Program of Fujian Province (2017H0044), and in part by CONICYT through Project BASAL-FB0008 and Project FONDECYT 1170167. Recommended for publication by Associate Editor A. Trzynadlowski. *(Corresponding author: Fengxiang Wang.)*

L. He and F. Wang are with the Quanzhou Institute of Equipment Manufacturing, Haixi Institutes, Chinese Academy of Sciences, Jinjiang 362200, China (e-mail: long.he@fjirsm.ac.cn; fengxiang.wang@fjirsm.ac.cn).

J. Wang is with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China, and also with the Key Laboratory of Industrial Internet of Things and Networked Control, Ministry of Education, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: wangjunxiao19860128@126.com).

J. Rodríguez is with the Faculty of Engineering, Universidad Andres Bello, Santiago 8370146, Chile (e-mail: jose.rodriguez@unab.cl).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TPEL.2019.2920439

programs are executed sequentially, which results in longer computation time and greater control delays than field programmable gate array (FPGA) programs with parallel processing capabilities. Recently, the price of FPGAs has dropped dramatically, and high-density FPGAs have been applied to high-performance motor drives [6], [22]. Programming FPGAs has traditionally been difficult and requires expertise in specialized "hardware description languages" like very high speed integrated circuit hardware description language or Verilog language. It takes a long time to design and verify the FPGA implementations of complex algorithms. Zynq SoC XC7Z020-CLG484-1, where dual-core Cortex-A9 processors and a FPGA core are incorporated, is an all-programmable system-on-chip (SoC) produced by Xilinx. It has the advantages of both FPGAs and DSPs/MCUs. Moreover, the Xilinx Vivado High-Level Synthesis (HLS) tool, which is provided by Xilinx transforms a C specification written in C or C++, System C, into a register transfer level implementation that can be synthesized into a Xilinx FPGA. Vivado HLS improves the level of design abstraction and accelerates the development of complex algorithms implemented on FPGAs [23].

To improve the tracking performance and dynamic response of surface mounted permanent magnet synchronous motor (SPMSM) drives, a robust predictive control scheme is proposed in this paper. By introducing an extended SPMSM model, which contains a lumped disturbance part, a Luenberger disturbance observer is designed to estimate the lumped disturbances in the speed and current loops, respectively. Compared to the sliding mode observer, the Luenberger observer has no chattering and requires less computation. Based on the estimated disturbances, the deadbeat-based predictive control law is applied to the speed loop and the current loop. Finally, the proposed control scheme is implemented on a Zynq-based motor driver to evaluate the HLS-based FPGA design method.

This paper is structured as follows. In Section II, one extended SPMSM model is built up by considering the external disturbances and parameter mismatches, then the proposed speed loop and current loop algorithms are explained. Then, in Section III, Zynq SoC implementation of the whole driver is presented and the conventional DPCC and the Luenberger disturbance observer based DPCC (LDOB-DPCC) are implemented on the FPGA core of Zynq. In Section IV, experimental results are presented and analyzed. Finally, the conclusions are drawn in Section V.

II. PMSM Model and Control Algorithms

A. Modeling of the SPMSM

The model of an SPMSM in synchronous rotating frame can be written as [24]

$$\begin{cases} \frac{di_d}{dt} = \frac{1}{L}(u_d - Ri_d + Lp\omega_m i_q) \\ \frac{di_q}{dt} = \frac{1}{L}(u_q - Ri_q - Lp\omega_m i_d - \lambda p\omega_m) \\ \frac{d\omega_m}{dt} = \frac{3}{2}\frac{p\lambda i_q}{J} - \frac{B\omega_m}{J} - \frac{T_L}{J} \end{cases} \tag{1}$$

where u_d, u_q mean the d- and q- axis stator voltages, L means the stator inductance, R means the stator resistance, i_d, i_q mean the d- and q- axis stator currents, ω_m means the angular velocity of the rotor, λ means the permanent magnet flux linkage, p means the number of pole pairs, B is the viscous damping coefficient, and J is the moment of inertia.

In consideration of the parameter variations, an extended PMSM model, which contains nominal parameters and a disturbance part, is proposed in [9] and [18]. An improved model is developed in our proposal. The extended d-axis current equation is derived as follows.

1) Assuming $v_d = \frac{1}{L}(-Ri_d + Lp\omega_m i_q)$, the first equation of (1) can be written as $\frac{di_d}{dt} = \frac{1}{L}u_d + v_d$.
2) As additional hardware is required to obtain u_d, it can be replaced by the voltage reference to the d-axis u_d^*. The mismatch between u_d and u_d^* can be incorporated into the disturbance part $d_d = v_d - \frac{1}{L}u_d^* + \frac{1}{L}u_d$, and the extended d-axis current equation is $\frac{di_d}{dt} = \frac{1}{L}u_d^* + d_d$.

Similarly, define $d_q = \frac{u_q}{L} - \frac{u_q^*}{L} - \frac{Ri_q}{L} - p\omega_m i_d - \frac{\lambda p\omega_m}{L}$ and $d_\omega = -\frac{B\omega_m}{J} - \frac{T_L}{J} + \frac{3}{2}p\lambda(\frac{i_q}{J} - \frac{i_q^*}{J_n})$, where J_n is the rotor inertia, i_q^* is the reference of q-axis stator current, and u_q^* is the voltage reference to q-axis. Then, the extended SPMSM model can be written as

$$\begin{cases} \frac{di_d}{dt} = \frac{u_d^*}{L} + d_d \\ \frac{di_q}{dt} = \frac{u_q^*}{L} + d_q \\ \frac{d\omega_m}{dt} = \frac{i_q^*}{k} + d_\omega \end{cases} \tag{2}$$

where $k = \frac{J_n}{\frac{3}{2}p\lambda}$.

B. Luenberger Disturbance Observer for the Speed Loop

As the sampling period of the speed loop is very small, d_ω can be regarded as a constant in one sampling period, that is $\frac{dd_\omega}{dt} = 0$, and the state function can be designed as follows:

$$\begin{cases} \frac{d\boldsymbol{x}}{dt} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ y = \boldsymbol{C}\boldsymbol{x} \end{cases} \tag{3}$$

where

$$\boldsymbol{x} = [\omega_m \quad d_\omega]^T, y = \omega_m, u = i_q^* \tag{4a}$$

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} \frac{1}{k} & 0 \end{bmatrix}^T, \boldsymbol{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}. \tag{4b}$$

In the Luenberger observer, the estimated state variables are corrected by comparing the estimated variables with the measured variables. The Luenberger observer is designed as

$$\begin{cases} \frac{d\hat{\boldsymbol{x}}}{dt} = \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}u + \boldsymbol{H}(y - \hat{y}) \\ \hat{y} \quad = \boldsymbol{C}\hat{\boldsymbol{x}} \end{cases} \tag{5}$$

where $\boldsymbol{H} = [h_1 \quad h_2]^T$, which is the observer gain matrix, $\hat{\boldsymbol{x}}$ is the estimated value of $\boldsymbol{x}$, $\hat{y}$ is the estimated value of y. From (3) and (5), the estimation error $\tilde{\boldsymbol{x}} = \hat{\boldsymbol{x}} - \boldsymbol{x}$ can be described by

$$\frac{d\tilde{\boldsymbol{x}}}{dt} = (\boldsymbol{A} - \boldsymbol{H}\boldsymbol{C})\tilde{\boldsymbol{x}}. \tag{6}$$

The observer gain matrix $\boldsymbol{H}$ should be designed to ensure that the estimation error decreases with time, so that all eigenvalues

of $(\boldsymbol{A}-\boldsymbol{HC})$ have negative real parts

$$Re\{\lambda_i(\boldsymbol{A}-\boldsymbol{HC})\}<0 \quad i=1,2. \tag{7}$$

Suppose $\lambda_1=\lambda_2=-l_\omega \quad (l_\omega>0)$, and the following equation can be solved to obtain the $\boldsymbol{H}$ matrix:

$$|\lambda \boldsymbol{I}-(\boldsymbol{A}-\boldsymbol{HC})|=(\lambda+l_\omega)^2. \tag{8}$$

So that, $\boldsymbol{H}=\begin{bmatrix}2l_\omega & l_\omega^2\end{bmatrix}^T$, and (5) can be discretized as

$$\begin{cases}\hat{\omega}_m(t+1)= \hat{\omega}_m(t)+T_s(\hat{d}_\omega(t)+\frac{i_q^*}{k} \\ \qquad\qquad +2l_\omega(\omega_m(t)-\hat{\omega}_m(t))) \\ \hat{d}_\omega(t+1)= \hat{d}_\omega(t)+T_s l_\omega^2(\omega_m(t)-\hat{\omega}_m(t))\end{cases} \tag{9}$$

where T_s is the sampling period of the speed loop. The selection of l_ω should be balanced between response performance and robustness. The larger the l_ω is, the farther the observer poles are from the origin, and the faster the state values are observed. However, the control system will be unstable when l_ω is too large.

C. Design of Speed Control

According to Euler formula, the third equation of (2) can be discretized into

$$\omega_m(t+1)-\omega_m(t)=T_s\left(\frac{i_q^*}{k}+d_\omega(t)\right). \tag{10}$$

Letting $\omega_m(t+1)=\omega_m^*$, where ω_m^* is the given speed. So, a deadbeat-based predictive speed control law is designed as $i_q^*=\frac{k}{T_s}(\omega_m^*-\omega_m(t))-kd_\omega(t)$.

Assuming $k_p=\frac{1}{T_s}$, which is adjusted according to actual needs, the speed control law is

$$i_q^*=k_pk(\omega_m^*-\omega_m(t))-kd_\omega(t). \tag{11}$$

A Lyapunov function is constructed to prove that the $e=\omega_m^*-\omega_m(t)$ vanishes with time when the speed controller (11) is applied, given as follows:

$$V=\frac{1}{2}e^2. \tag{12}$$

Derivation of the V is

$$\frac{dV}{dt}=-e\left(\frac{i_q^*}{k}+d_\omega(t)\right)=-k_pe^2. \tag{13}$$

So that, in the condition of $k_p>0$, e will be decreased with time.

Substituting the estimated disturbance $\hat{d}_\omega(t)$ into the (11), the Luenberger disturbance observer based speed controller (LDOB-SPDC) is

$$i_q^*=k_pk(\omega_m^*-\omega_m(t))-k\hat{d}_\omega(t) \tag{14}$$

where a large k_p results in a more responsive or more sensitive controller. But, if the k_p is too high, the controller can become unstable. Fig. 1 shows the block diagram of the proposed speed controller.

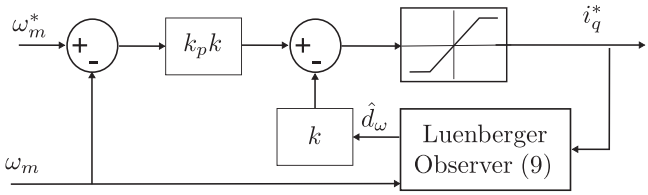

Fig. 1. Block diagram of the speed controller that incorporates the Luenberger observer (LDOB-SPDC).

D. Conventional DPCC

From (1), according to Euler formula, the discrete voltage model of SPMSM can be written as

$$\begin{cases}u_d(t)= L\frac{i_d(t+1)-i_d(t)}{T_c} \\ \qquad\quad +Ri_d(t)-Lp\omega_m(t)i_q(t) \\ u_q(t)= L\frac{i_q(t+1)-i_q(t)}{T_c} \\ \qquad\quad +Ri_q(t)+Lp\omega_m(t)i_d(t)+\lambda p\omega_m(t)\end{cases} \tag{15}$$

where T_c is the sampling period of the current loop. Setting $i_d(t+1)=i_d^*$, $i_q(t+1)=i_q^*$, where i_d^* and i_q^* are the current references to *d*- and *q*- axis, and the DPCC law can be described as

$$\begin{cases}u_d(t)= L\frac{i_d^*-i_d(t)}{T_c}+Ri_d(t)-Lp\omega_m(t)i_q(t) \\ u_q(t)= L\frac{i_q^*-i_q(t)}{T_c}+Ri_q(t)+Lp\omega_m(t)i_d(t)\cdot \\ \qquad\quad +\lambda p\omega_m(t)\end{cases} \tag{16}$$

From (16), the performance of the DPCC is largely determined by the accuracies of the motor parameters such as the stator inductance, the stator resistance, and the permanent magnet flux linkage, which vary with the motor operating conditions and environmental conditions.

To analyze the parameter sensitivity of the DPCC, assume $R_0=R+R^{'}$, $L_0=L+L^{'}$, $\lambda_0=\lambda+\lambda^{'}$, where $R^{'}$, $L^{'}$, $\lambda^{'}$ are the parameter mismatches. Applying R_0, L_0, and λ_0 to (16), it obtains

$$\begin{cases}u_d^{'}= L\frac{i_d^*-i_d(t)}{T_c}+Ri_d(t) \\ \qquad\quad -Lp\omega_m(t)i_q(t)+\Delta u_d(t) \\ u_q^{'}= L\frac{i_q^*-i_q(t)}{T_c}+Ri_q(t) \\ \qquad\quad +Lp\omega_m(t)i_d(t)+\lambda p\omega_m(t)+\Delta u_q(t)\end{cases} \tag{17}$$

where $\Delta u_d(t)=L^{'}\frac{i_d^*-i_d(t)}{T_c}+R^{'}i_d(t)-L^{'}p\omega_m(t)i_q(t)$, $\Delta u_q(t)=L^{'}\frac{i_q^*-i_q(t)}{T_c}+R^{'}i_q(t)+L^{'}p\omega_m(t)i_d(t)+\lambda^{'}p\omega_m(t)$, which are introduced by parameter mismatches. Combining (15) and (17), it obtains

$$\begin{cases}i_d(t+1)=i_d^*+\Delta u_d(t)\frac{T_c}{L} \\ i_q(t+1)=i_q^*+\Delta u_q(t)\frac{T_c}{L}.\end{cases} \tag{18}$$

There are errors between the given current and the response current due to parameter variations as shown in (18). The work presented in [18] has proposed a sliding mode disturbance observer to estimate the $\Delta u_d(t)$ and $\Delta u_q(t)$. The voltage reference calculated by deadbeat predictive current controller is compensated

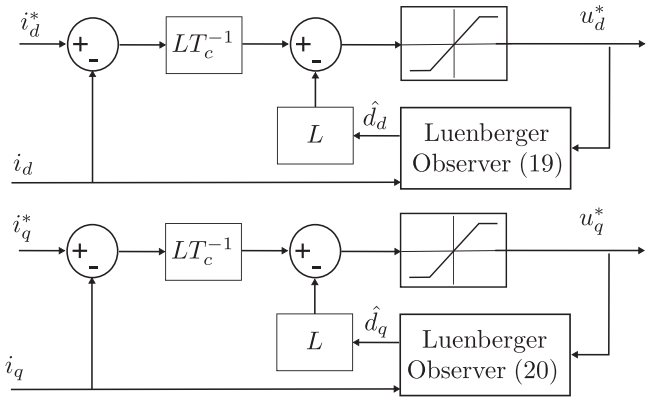

Fig. 2. Block diagram of the LDOB-DPCC.

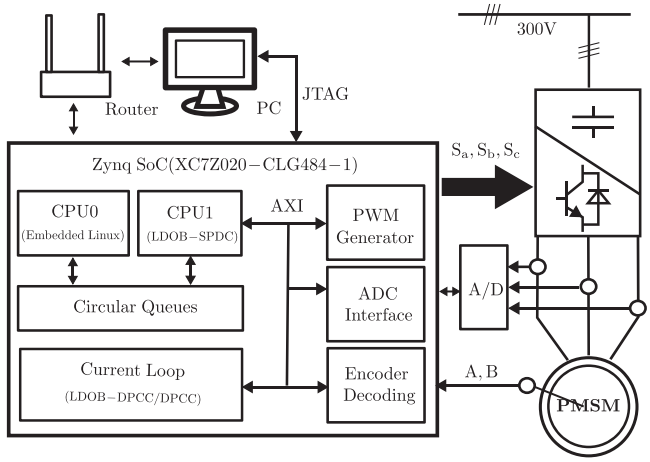

Fig. 3. Block diagram of the designed Zynq-based controller for an SPMSM drive.

by the estimated disturbance. However, the nominal values of R and λ are required by the method proposed by the work presented in [18]. In Section II-E, an improved DPCC method based on the Luenberger disturbance observer is proposed, which only requires the initialization of the nominal parameter of L. The proposed method, in turn reduces the computational complexity and improves the robustness.

E. Design of Improved DPCC

To estimate the d_d and d_q from (2), two Luenberger observers designed in a similar way as that of the speed loop are given by

$$\begin{cases} \hat{i}_d(t+1) = \hat{i}_d(t) + T_c(\hat{d}_d(t) + \frac{u_d^*}{L} \\ \qquad\qquad + 2l_d(i_d(t) - \hat{i}_d(t))) \\ \hat{d}_d(t+1) = \hat{d}_d(t) + T_c l_d^2(i_d(t) - \hat{i}_d(t)) \end{cases} \tag{19}$$

and

$$\begin{cases} \hat{i}_q(t+1) = \hat{i}_q(t) + T_c(\hat{d}_q(t) + \frac{u_q^*}{L} \\ \qquad\qquad + 2l_q(i_q(t) - \hat{i}_q(t))) \\ \hat{d}_q(t+1) = \hat{d}_q(t) + T_c l_q^2(i_q(t) - \hat{i}_q(t)). \end{cases} \tag{20}$$

According to the Euler formula, the current equations of (2) are discretized as

$$\begin{cases} u_d^* = L\frac{i_d(t+1) - i_d(t)}{T_c} - Ld_d(t) \\ u_q^* = L\frac{i_q(t+1) - i_q(t)}{T_c} - Ld_q(t). \end{cases} \tag{21}$$

Substituting the estimated disturbances $\hat{d}_d$ and $\hat{d}_q$ from (19) and (20) into (21) and let $i_d(t+1) = i_d^*$, $i_q(t+1) = i_q^*$, then the LDOB-DPCC law can be designed as

$$\begin{cases} u_d^* = L\frac{i_d^* - i_d(t)}{T_c} - L\hat{d}_d(t) \\ u_q^* = L\frac{i_q^* - i_q(t)}{T_c} - L\hat{d}_q(t). \end{cases} \tag{22}$$

The block diagram of the LDOB-DPCC is shown in Fig. 2.

III. Zynq SoC Implementation

A. Design of Zynq SoC Based Controller

Fig. 3 shows the block structure of the designed Zynq SoC based controller for an SPMSM drive. It is noted that XC7Z020-CLG484-1, which incorporates dual-core Cortex-A9 processors and an FPGA core, is functionally divided into four parts as follows.

1) FPGA. The current loop control module and hardware interface modules such as PWM generator, analog-to-digital converter (ADC) interface, and encoder decoding are implemented on the FPGA core. They are designed as peripherals of CPU1 and communicate via the advanced extensible interface (AXI) bus. The current loop control module is designed based on the Vivado HLS tool. The hardware interface modules, which require more stringent timing and higher efficiency, are designed by using Verilog language.
2) CPU1. The speed loop control module (the proposed LDOB-SPDC) that requires flexible configuration, good expandability, and strong portability in industry applications is implemented on the CPU1, where a real-time operating system works.
3) CPU0. The embedded Linux operating system responsible for communicating with the personal computer (PC) works on the CPU0.
4) On chip memory (OCM). Two circular queues for communication between CPU0 and CPU1 are implemented on the OCM.

The Vivado tool is used for programming the Zynq SoC, and they communicate with each other by Joint Test Access Group (JTAG). A Qt-based graphics monitoring program has been developed that runs on the PC and communicates with Zynq over Gigabit Ethernet where a router helps to "route" data.

B. HLS-Based FPGA Implementation

Since the Vivado HLS tool supports the C++ float-point data type, which represents numbers in a wider dynamic range than fixed point, floating point is used to construct the current loop

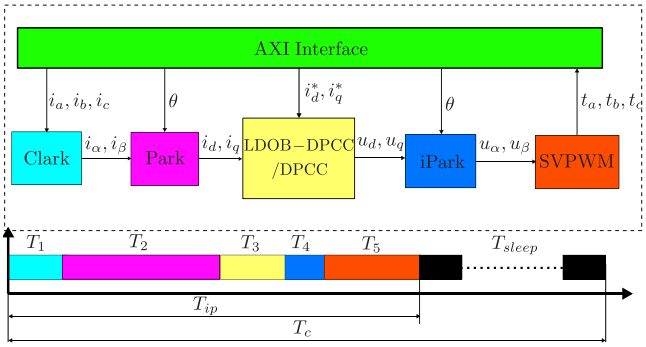

Fig. 4. FPGA implementation overview of the current loop module.

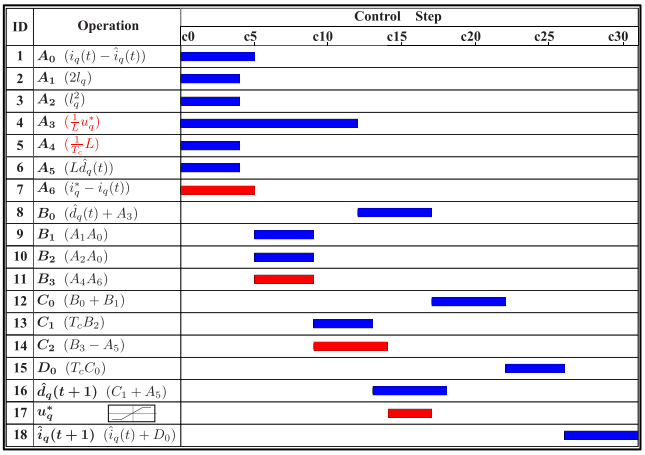

Fig. 5. Process of calculating u_q^* with the LDOB-DPCC.

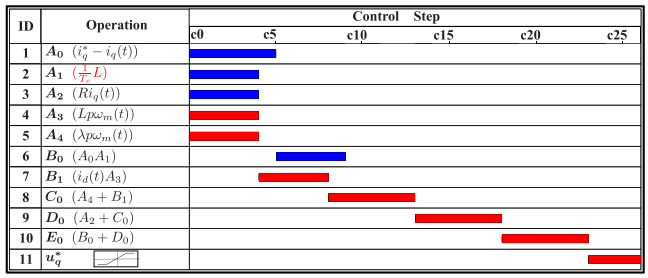

Fig. 6. Process of calculating u_q^* with the DPCC.

TABLE I
FPGA DESIGN DETAIL

Timing	LDOB-DPCC	DPCC	Resources	LDOB-DPCC	DPCC
T_1	14 ticks	14 ticks	FF	9281	8780
T_2	91 ticks	91 ticks	LUT	9861	9528
T_3	17 ticks	26 ticks	DSP48E	52	38
T_4	8 ticks	8 ticks	$BRAM_18K$	0	0
T_5	56 ticks	56 ticks			
T_{sleep}	$\geq 78\mu s$	$\geq 78\mu s$			
T_{ip}	$\leq 2\mu s$	$\leq 2\mu s$			
T_c	$80\mu s$	$80\mu s$			

module. Blocks such as Park, Clark, LDOB-DPCC, DPCC, iPark, space vector PWM (SVPWM) are realized by C++ functions. The INLINE directive is applied to small functions such as Park, Clark, etc., which permits operations within the functions to be shared and optimized more effectively. The PIPELINE directive is a good choice for high-performance operation, and it is applied to loops in the SVPWM block to reduce the initiation interval. The AXI interface is packaged in the current loop module, enabling the CPU1 to access the module like a peripheral device. The implementation architecture of the FPGA-based current loop module is depicted in Fig. 4, where different processing times are represented by different colors. The implementation of the module is introduced as follows in sequence from T_1 to T_{sleep}.

1) During T_1 (14 ticks, i.e., 0.14 μs with 100 MHz clock): A Clarke transformation from i_a, i_b, i_c, which are obtained from AXI interface, to i_α, i_β is performed by Clark block.
2) During T_2 (91 ticks): A park transformation from i_α, i_β to i_d, i_q is performed by Park block. The sine and cosine functions provided by the Vivado HLS math library are used, which have an accuracy of 1 unit of least precision. But, they are time consuming, approximately 81 ticks in all.
3) During T_3 (17 ticks for the LDOB-DPCC, 26 ticks for the DPCC): The calculation processes of u_q using the LDOB-DPCC and DPCC are, respectively, shown in Figs. 5 and 6, where the critical paths are indicated in red. 1) For the LDOB-DPCC, it takes 18 operations and 31 ticks to obtain the u_q^*, $\hat{d}_q(t+1)$, $\hat{i}_q(t+1)$. The computation processes of the u_q^*, $\hat{d}_q(t+1)$, $\hat{i}_q(t+1)$ are scheduled in parallel with each other. As the LDOB-DPCC block is inlined, the $\hat{d}_q(t+1)$ and $\hat{i}_q(t+1)$ calculations are optimized and calculated in parallel with iPark block, which starts operating as soon as the u_q^* is inputted. Therefore, the time consumption of the LDOB-DPCC is reduced to 17 ticks. 2) For the DPCC, it takes 11 operations and 26 ticks to obtain the u_q^*. The d-axis computation process is similar, and operates in parallel with that of the q-axis. Although the LDOB-DPCC requires more operations than the DPCC, it takes less time due to the parallel processing capabilities of FPGAs.
4) During T_4 (8 ticks): An inverse park transformation from u_d, u_q to u_α, u_β is performed by iPark block. Since the sine and cosine functions have been calculated during T_2, only 8 ticks are needed in this procedure.
5) During T_5 (56 ticks): The duty cycle for each switching device is calculated in this procedure based on the u_α, u_β.

The area utilization and time consumption of the two implementations are given in Table I. It is noted that the LDOB-DPCC takes less time than the DPCC, but requires more resources.

IV. EXPERIMENTAL RESULTS

Experimental tests were carried out on the test bench shown in Fig. 7. The test bench consists of two SPMSMs made by Wenling Yuhai Electromechanical Corporation with the parameters given in Table II. The load motor is driven by a 3.0 kW vector control inverter manufactured by Micno Corporation, while the driven motor is driven by a Zynq-based inverter that includes a two-level voltage source inverter and a Zynq board. The dc bus of the load and driven inverters are connected together. The proposed

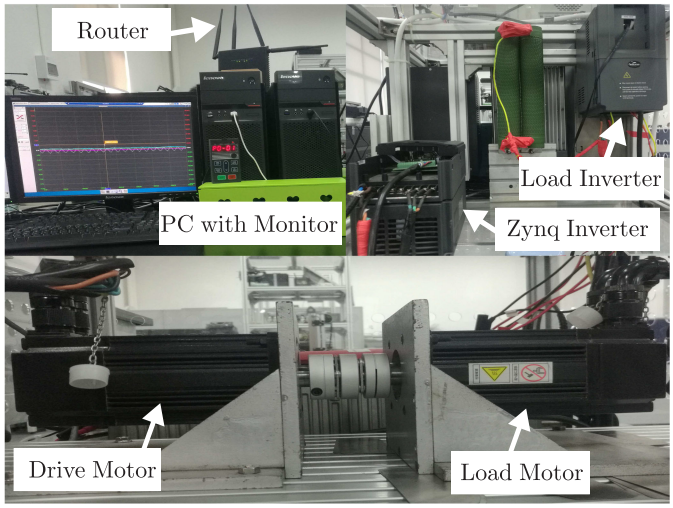

Fig. 7. Experimental setup description.

TABLE II
PARAMETERS OF SPMSM

Descriptions	Parameters	Nominal Values
Stator resistance	R	1.7912 Ω
Stator inductance	L	3.5 mH
Pole pairs	p	4
PM flux	λ	0.0799 Vs
Rated power	P_N	0.75 kW
Rotor inertia	J_n	0.00024 $Kg \cdot m^2$
Rated current	I_N (eff.)	3.5 A
Rated speed	ω_{MN}	3000 rpm
Rated voltage	U_N (eff.)	220 V
Rated torque	T_{MN}	2.4 Nm
DC link voltage	V_{dc}	300 V

control scheme operates at 12.5 kHz, which is equal to the PWM switching frequency.

A. Speed Reversal Performance

An experiment to verify the performance during a rated full-speed range is carried out in Fig. 8. As the target speed ω_m^* changes from 3000 to -3000 r/min, the q-axis current, which is linearly related to the electromagnetic torque, reaches its maximum value within 3 ms. The speed reverses from 3000 to -3000 r/min within 0.06 s. The experimental results show that the proposed method has fast dynamic response and good performance in the full speed range.

B. I_q Step Performance

The i_q step performances of the LDOB-DPCC and DPCC are evaluated in Figs. 9 and 10. The Zynq inverter operates at current control mode that i_q^* is directly set in the program. The speed of the motor is given by the Micno load inverter. The SPMSM operates at 600 r/min, and the i_q^* alters from 0 to 2 A. It can be seen that the LDOB-DPCC achieves better dynamic performance than the DPCC. For the DPCC, i_q rises to about 4 A first, and then drops back to 2 A within 3 ms. For the LDOB-DPCC, i_q reaches the target value within 1.0 ms, and there is no overshoot. Moreover, the i_q ripple of the LDOB-DPCC (0.2 A) is less than that of the DPCC (0.3 A).

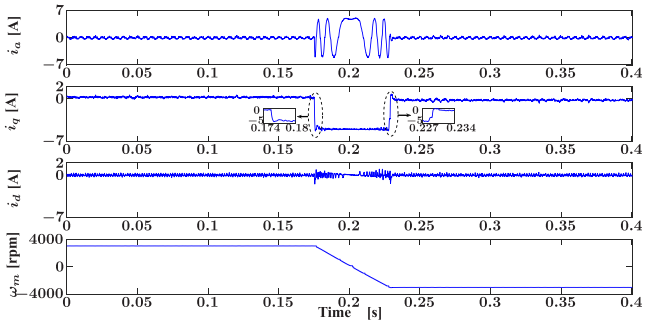

Fig. 8. Experimental results during the rated full-speed reversal process. (LDOB-SPDC cascade with LDOB-DPCC, from 3000 to -3000 r/min).

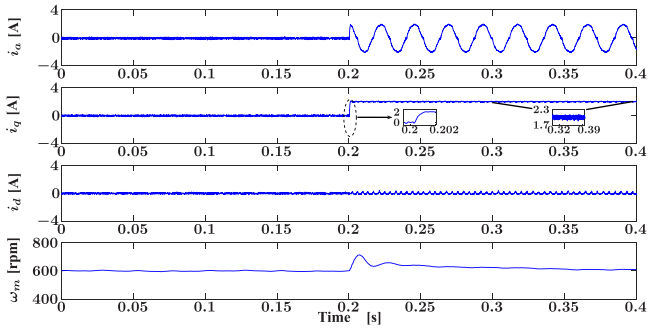

Fig. 9. Experimental results of i_q step performance using the LDOB-DPCC. (600 r/min, i_q alters from 0 to 2 A).

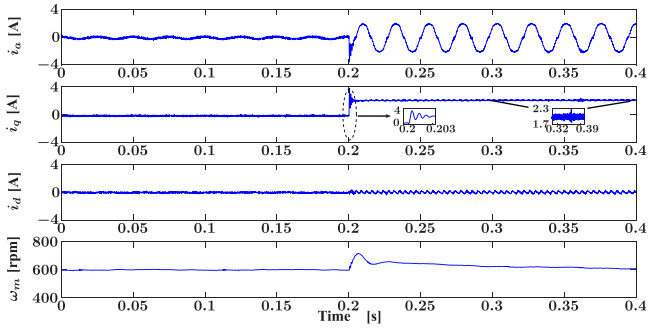

Fig. 10. Experimental results of i_q step performance using the DPCC. (600 r/min, i_q alters from 0 to 2 A).

C. Load Disturbance Performance

The load disturbance performances of the LDOB-SPDC and PI control are studied in Figs. 11 and 12. Both methods are implemented at a speed of 600 r/min with a 2.4 N·m load disturbance. When the load disturbance is applied, the rotor speed of the LDOB-SPDC has a slight reduction of 60 r/min at the

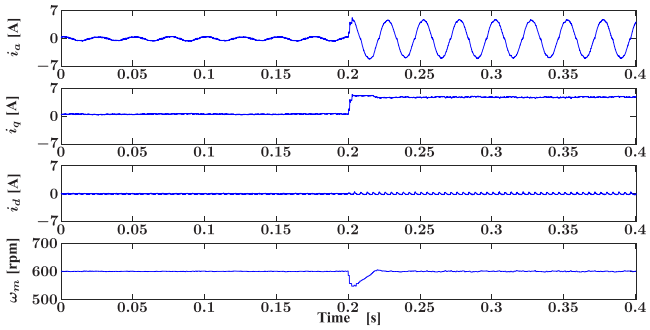

Fig. 11. Experimental results of load disturbance performance. (LDOB-SPDC cascade with LDOB-DPCC, 600 r/min, load torque alters from 0 to 2.4 N·m).

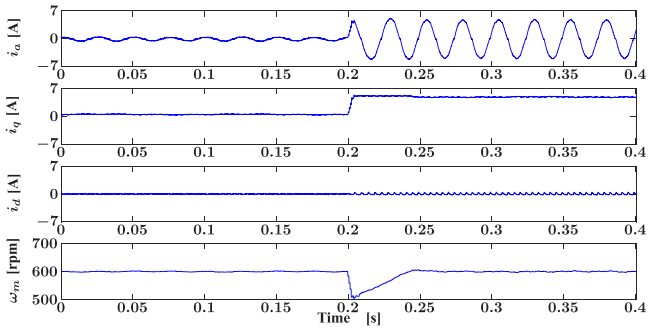

Fig. 12. Experimental results of load disturbance performance. (PI control cascade with LDOB-DPCC, 600 r/min, load torque alters from 0 to 2.4 N·m).

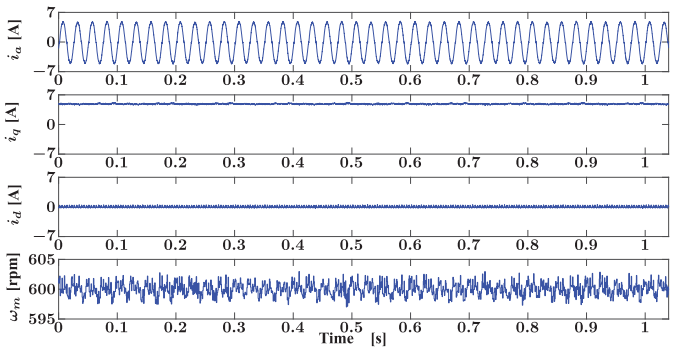

Fig. 13. Experimental results of steady-state performance. (LDOB-SPDC cascade with LDOB-DPCC, 600 r/min, 2.4 N·m).

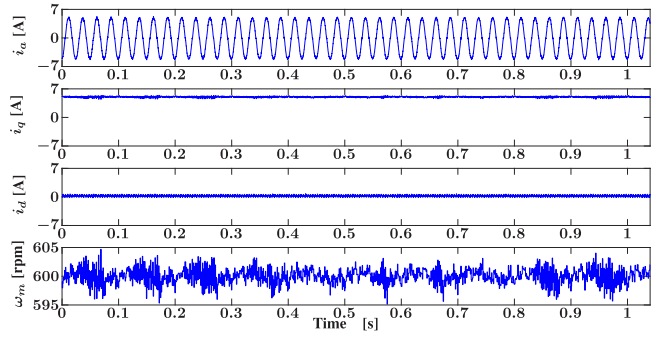

Fig. 14. Experimental results of steady-state performance. (PI control cascade with LDOB-DPCC, 600 r/min, 2.4 N·m).

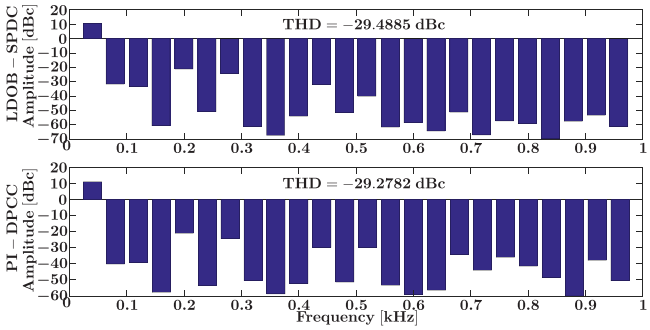

Fig. 15. THDs of a-phase stator currents using the PI control and LDOB-SPDC. (LDOB-DPCC current loop, 600 r/min, 2.4 N·m).

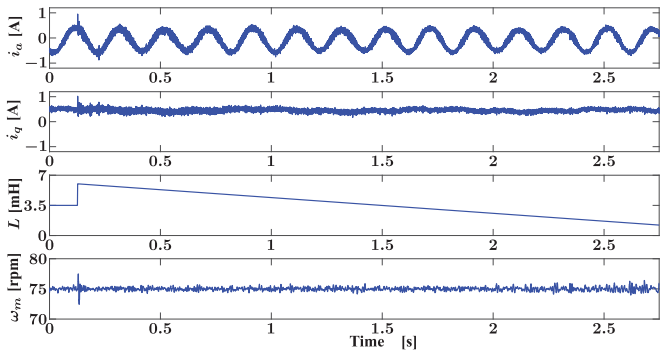

Fig. 16. Steady-state performance under stator inductance variation. (LDOB-SPDC cascade with LDOB-DPCC, 75 r/min).

initial time (smaller than that of the PI control, 100 r/min) and then returns to the given speed within 0.04 s (0.06 s for the PI control). It is illustrated that the LDOB-SPDC is superior to the PI control in terms of load disturbance performance.

D. Steady-State Performance

The rotor speeds, d- and q- axis stator currents and a-phase stator currents of the LDOB-SPDC and PI control at a given speed of 600 r/min with a load torque of 2.4 N·m are shown in Figs. 13 and 14. It is noted that the speed ripple of the LDOB-SPDC is less than 7 r/min, and that of the PI control is greater than 10 r/min as a comparison. Both methods achieve good stator currents. In order to compare the stator current quality of the two methods, the total harmonic distortions (THDs) of a-phase stator currents are further analyzed in Fig. 15, where a-phase stator current of the LDOB-SPDC has smaller THD (-29.4885 dBc) than that of the PI control (-29.2782 dBc).

E. Parameter Sensitivity Evaluation

From (19), (20), and (22), the LDOB-DPCC is only affected by stator inductance, while the DPCC is affected by stator inductance, stator resistance, and permanent magnet flux linkage of SPMSM. Figs. 16 and 17 show the steady-state comparison of the LDOB-DPCC and DPCC under stator inductance variation, respectively. The speed reference is given as 75 r/min, and the stator inductance is reduced from 6 to 0.6 mH within 2.6 s. It can be seen that the LDOB-DPCC performs better than the DPCC in

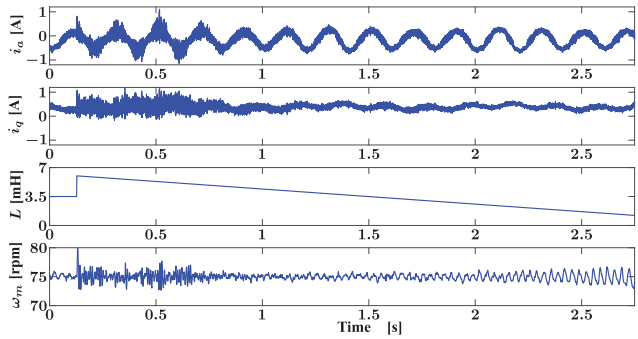

Fig. 17. Steady-state performance under stator inductance variation. (LDOB-SPDC cascade with DPCC, 75 r/min).

terms of i_q ripple and speed variation over the full range of stator inductance variation. The speed ripple of the LDOB-DPCC (less than 1 r/min) is larger than that of the DPCC (about 5 r/min) when the stator inductance varies from 6 to 5 mH and from 1.8 to 0.6 mH.

V. Conclusion

In this paper, the disturbances and mismatches in speed d- and q-axis current loops are estimated by using the Luenberger observer. Based on the estimated disturbances, the LDOB-SPDC and LDOB-DPCC are applied to the speed and current loops of SPMSM control scheme, respectively. The proposed method is implemented on the Zynq SoC, and the FPGA implementations of the LDOB-DPCC and DPCC are compared and analyzed in detail in terms of area utilization and time consumption. The current loop module, PWM generator module, encoder decoding module, and ADC interface module are fully implemented on the FPGA core of Zynq, and the current loop execution time is greatly shortened benefit from the parallel processing capabilities of the FPGAs. HLS-based design not only speeds up the development of the current loop, but also improves the accuracy of the algorithm. Experiments show that the LDOB-DPCC has better dynamic performance and parameter robustness than the DPCC, and the LDOB-SPDC has better steady-state performance, dynamic response performance, and load disturbance suppression ability than the PI control.

References

[1] A. Houari *et al.*, "An effective compensation technique for speed smoothness at low-speed operation of PMSM drives," *IEEE Trans. Ind. Appl.*, vol. 54, no. 1, pp. 647–655, Jan./Feb. 2018.

[2] H. Liu and S. Li, "Speed control for PMSM servo system using predictive functional control and extended state observer," *IEEE Trans. Ind. Electron.*, vol. 59, no. 2, pp. 1171–1183, Feb. 2012.

[3] Y. Jiang, W. Xu, C. Mu, and Y. Liu, "Improved deadbeat predictive current control combined sliding mode strategy for PMSM drive system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 251–263, Jan. 2018.

[4] C. Xia, B. Ji, T. Shi, and Y. Yan, "Two-degree-of-freedom proportional integral speed control of electrical drives with Kalman-filter-based speed estimation," *IET Elect. Power Appl.*, vol. 10, no. 1, pp. 18–24, 2016.

[5] S. Li and H. Gu, "Fuzzy adaptive internal model control schemes for PMSM speed-regulation system," *IEEE Trans. Ind. Inform.*, vol. 8, no. 4, pp. 767–779, Nov. 2012.

[6] Y. Kung and M. Tsai, "FPGA-based speed control IC for PMSM drive with adaptive fuzzy control," *IEEE Trans. Power Electron.*, vol. 22, no. 6, pp. 2476–2486, Nov. 2007.

[7] X. Liu, H. Yu, J. Yu, and L. Zhao, "Combined speed and current terminal sliding mode control with nonlinear disturbance observer for PMSM drive," *IEEE Access*, vol. 6, pp. 29594–29601, 2018.

[8] Y. Yan, J. Yang, Z. Sun, C. Zhang, S. Li, and H. Yu, "Robust speed regulation for PMSM servo system with multiple sources of disturbances via an augmented disturbance observer," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 2, pp. 769–780, Apr. 2018.

[9] X. Zhang, L. Sun, K. Zhao, and L. Sun, "Nonlinear speed control for PMSM system using sliding-mode control and disturbance compensation techniques," *IEEE Trans. Power Electron.*, vol. 28, no. 3, pp. 1358–1365, Mar. 2013.

[10] S.-H. Chang and P.-Y. Chen, "Self-tuning gains of PI controllers for current control in a PMSM," in *Proc. 5th IEEE Conf. Ind. Electron. Appl.*, Jun. 2010, pp. 1282–1286.

[11] M. Siami, D. A. Khaburi, M. Rivera, and J. Rodriguez, "An experimental evaluation of predictive current control and predictive torque control for a PMSM fed by a matrix converter," *IEEE Trans. Ind. Electron.*, vol. 64, no. 11, pp. 8459–8471, Nov. 2017.

[12] F. Wang, S. Li, X. Mei, W. Xie, J. Rodriguez, and R. M. Kennel, "Model-based predictive direct control strategies for electrical drives: An experimental evaluation of PTC and PCC methods," *IEEE Trans. Ind. Inform.*, vol. 11, no. 3, pp. 671–681, Jun. 2015.

[13] B.-J. Kang and C.-M. Liaw, "A robust hysteresis current-controlled PWM inverter for linear PMSM driven magnetic suspended positioning system," *IEEE Trans. Ind. Electron.*, vol. 48, no. 5, pp. 956–967, Oct. 2001.

[14] S. Wendel, B. Haucke-Korber, A. Dietz, and R. Kennel, "Cascaded continuous and finite model predictive speed control for electrical drives," in *Proc. 20th Eur. Conf. Power Electron. Appl.*, Sep. 2018, pp. 1–10.

[15] M. Abdelrahem, C. Hackl, R. Kennel, and J. Rodriguez, "Efficient direct-model predictive control with discrete-time integral action for PMSGs," *IEEE Trans. Energy Convers.*, vol. 34, no. 2, pp. 1063–1072, Jun. 2019.

[16] S. A. Davari, F. Wang, and R. M. Kennel, "Robust deadbeat control of an induction motor by stable MRAS speed and stator estimation," *IEEE Trans. Ind. Inform.*, vol. 14, no. 1, pp. 200–209, Jan. 2018.

[17] J. Stumper, S. Kuehl, and R. Kennel, "Robust deadbeat control for synchronous machines rejecting noise and uncertainties by predictive filtering," in *Proc. 8th Int. Conf. Power Electron. – ECCE Asia*, May 2011, pp. 1378–1385.

[18] X. Zhang, B. Hou, and Y. Mei, "Deadbeat predictive current control of permanent-magnet synchronous motors with stator current and disturbance observer," *IEEE Trans. Power Electron.*, vol. 32, no. 5, pp. 3818–3834, May 2017.

[19] L. Niu, M. Yang, and D. Xu, "Predictive current control for permanent magnet synchronous motor based on deadbeat control," in *Proc. 7th IEEE Conf. Ind. Electron. Appl.*, Jul. 2012, pp. 46–51.

[20] H. H. Choi, N. T. Vu, and J. Jung, "Digital implementation of an adaptive speed regulator for a PMSM," *IEEE Trans. Power Electron.*, vol. 26, no. 1, pp. 3–8, Jan. 2011.

[21] T. Tarczewski and L. M. Grzesiak, "Constrained state feedback speed control of PMSM based on model predictive approach," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3867–3875, Jun. 2016.

[22] Z. Ma, S. Saeidi, and R. Kennel, "FPGA implementation of model predictive control with constant switching frequency for PMSM drives," *IEEE Trans. Ind. Inform.*, vol. 10, no. 4, pp. 2055–2063, Nov. 2014.

[23] A. Qamar, F. B. Muslim, F. Gregoretti, L. Lavagno, and M. T. Lazarescu, "High-level synthesis for semi-global matching: Is the juice worth the squeeze?" *IEEE Access*, vol. 5, pp. 8419–8432, 2017.

[24] Y. Feng, X. Yu, and F. Han, "High-order terminal sliding-mode observer for parameter estimation of a permanent-magnet synchronous motor," *IEEE Trans. Ind. Electron.*, vol. 60, no. 10, pp. 4272–4280, Oct. 2013.

Long He was born in Fuzhou, China, in 1984. He received the B.S. degree in applied physics and the M.S. degree in physical electronics from Fuzhou University, Fuzhou, China, in 2007 and 2010, respectively.

He is currently with Haixi Institutes, Chinese Academy of Sciences, Quanzhou, China. His research interests include predictive control and sensorless control for electrical drives.

Fengxiang Wang (S'13–M'14–SM'18) was born in Jiujiang, China, in 1982. He received the B.S. degree in electronic engineering and the M.S. degree in automation from Nanchang Hangkong University, Nanchang, China, in 2005 and 2008, respectively, and the Ph.D. degree from the Institute for Electrical Drive Systems and Power Electronics, Technische Universitaet Muenchen, Munich, Germany, in 2014.

He is currently with Haixi Institutes, Chinese Academy of Sciences, Quanzhou, China. His research interests include predictive control and sensorless control for electrical drives.

Junxiao Wang (S'14–M'18) was born in Wuxue, Hubei Province, China, in 1986. He received the B.S. and M.S. degrees in control theory and control engineering from Information Engineering College, Henan University of Science and Technology, Luoyang, China, in 2008 and 2011, respectively, and the Ph.D. degree in control theory and control engineering from the School of Automation, Southeast University, Nanjing, China, in 2017.

From September 2015 to 2016, he was a Visiting Ph.D. Student with the Institute for Electrical Drive Systems and Power Electronics, Technical University of Munich, Munich, Germany. He is currently with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His research interests include advanced control theory and its application to power electronics and motion control systems.

Dr. Wang was the recipient of the 2017 IET Premium Award for best paper in IET Control Theory and Applications.

José Rodríguez (M'81–SM'94–F'10) received the Engineering degree in electrical engineering from the Universidad Tecnica Federico Santa Maria, Valparaiso, Chile, in 1977, and the Dr.-Ing. degree in electrical engineering from the University of Erlangen, Erlangen, Germany, in 1985.

Since 1977, he has been with the Department of Electronics Engineering, Universidad Tecnica Federico Santa Maria, where he was a Full Professor and the President. Since 2015, he has been the President of Universidad Andres Bello, Santiago, Chile. He has coauthored two books, several book chapters, and more than 400 journal and conference papers. His research interests include multilevel inverters, new converter topologies, control of power converters, and adjustable-speed drives.

Dr. Rodríguez is the recipient of a number of Best Paper awards from the journals of the IEEE. He is a member of the Chilean Academy of Engineering. In 2014, he was the recipient of the National Award of Applied Sciences and Technology from the Government of Chile. In 2015, he was the recipient of the Eugene Mittelmann Award from the Industrial Electronics Society of the IEEE.