Single-chip FPGA Implementation of a Sensorless Speed Control IC for Permanent Magnet Synchronous Motors

Yen-Chuan Chang and Ying-Yu Tzou, *Member, IEEE*

Power Electronics Systems and Chips Lab.
Advanced Power Electronics Center
Department of Electrical and Control Engineering
National Chiao Tung Univ., Hsinchu, Taiwan

***Abstract*—This paper presents a single-chip FPGA implementation of a speed sensorless control IC with flux-linkage estimation algorithm for permanent-magnet synchronous motors (PMSM) with sinusoidal flux distribution. By using hierarchical and modular realization strategy, each functional block for the digital sensorless IC is designed as a reusable IP with FPGA implementation. The digital sensorless IC is configurable with its modular blocks and is programmable with its control registers via a serial interface to fit various applications without complicated software programming as realized by using conventional DSP or microcontrollers. The sensorless algorithm implemented uses the incremental values of flux linkage, the normalized back-EMF voltage, and the estimated peak back-EMF value to estimate the incremental rotor position. An internal closed-loop correction mechanism within this algorithm can automatically correct the rotor position estimation drift, which could be resulted by the quantization error, circuit nonlinearities, and sampling noises. An initial rotor position detection scheme with ramping speed control has been developed for startup control. Experimental verification has been carried out on a sensorless spindle motor drive system for DVD applications.**

***Index Terms*—sensorless operation, start-up, FPGA, PMSM, digital control IC.**

I. Introduction

Permanent-magnet synchronous motors (PMSM), due to its high ratio of torque to weight, low noise, fast response, high efficiency and easy maintenance, are wildly used in many high-performance speed and position controlled applications. Usually, absolute encoders or Hall effect sensors are attached to the shaft of the motor to detect the rotor position. However, there also exists some disadvantages such as additional installation of the sensors and signal conditioning circuit, sensitive to magnetic noises, and unsuitable for high temperature application environment. To prevent the above problems, many sensorless techniques are presented to develop reliable and low cost control strategies for PMSMs [1].

With the rapidly development of integrated circuit, digital motor control systems have been widely implemented with software based on microcontrollers or digital signal processors [2]. These approaches provide flexibility and are suitable for many applications. However, this control scheme suffers from a long period of development and requires a high sampling rate to achieve a wide bandwidth performance. In recent years, FPGA-based hardware implementation technology has been used to motor control systems due to the advantages of their programmable hard-wired feature, fast time-to-market and reusable IP (Intellectual Property) cores. Besides, the FPGA-based system can get a very high speed level, since it can carry out parallel processing by means of hardware mode. Therefore, many research in ac motor drives have been developed with FPGA [3]-[5].

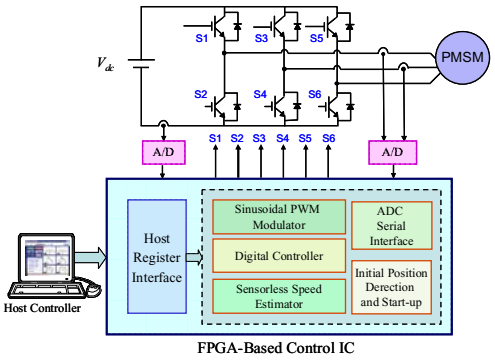

Fig. 1. Block diagram of a sensorless speed control system with the proposed FPGA implementation architecture.

The research goal of this paper is to design a programmable speed sensorless control IC, which can be used for not only high-speed spindle motor drives, but also other high-performance applications with PMSMs. Fig. 1 shows the system block diagram of the speed control system using the proposed sensorless IC. It contains all the blocks needed for realization a PMSM drive system, such as sinusoidal PWM modulation, sensorless position estimator, digital controller, initial position estimator and analog-to-digital read-out interface. By using the hierarchical and modular realization strategy, every block in FPGA is designed toward a specific function of control law, and has its own registers which can be configured by users to build up a control system according to different applications. Unlike traditional DSP or microcontrollers, users do not need to design software programs to develop control algorithms. Therefore, it takes

This work was supported by the National Science Council, Taipei, Taiwan, R.O.C. Project no. NSC 95-2221-E-009-338-MY3.

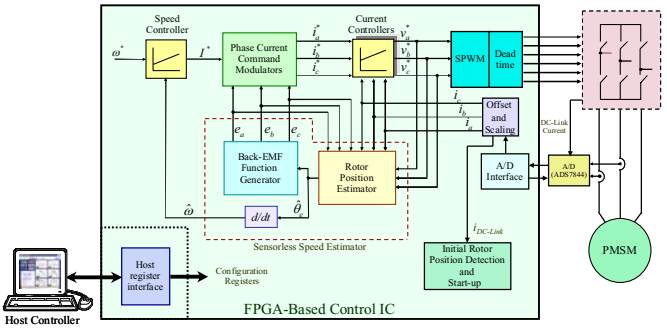

Fig. 2. The internal architecture of the FPGA-based sensorless speed control IC.

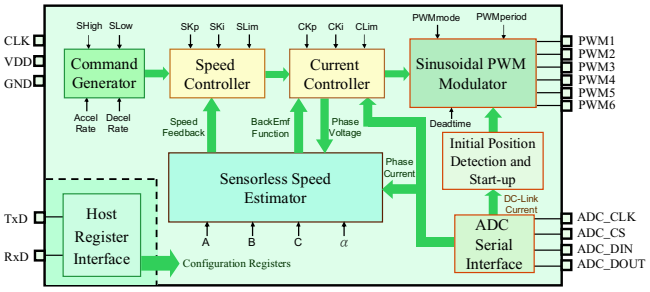

Fig. 3. Functional block diagram of the sensorless speed control IC.

little effort building up a control system, and provides a flexible and fast time-to-market solution for high-performance PMSM motor drives without Hall effect sensors.

II. Architectural Realization and Design Considerations

A. System Configuration

Fig. 2 shows the internal architecture of the FPGA-based sensorless control IC. The developed sensorless control IC consists of four major control blocks: a speed controller, a field-oriented torque controller, a sensorless position estimator, and an initial position estimator. The speed controller is used to regulate the desired speed and generate the torque command which is multiplied by the estimated back-EMF to produce phase current command. In this way, the flux-oriented torque control is achieved by generating current references in the three-phase stationary frame without the stationary-to-rotating reference frame transformation and its inverse.

The rotor position information is obtained by the sensorless algorithm based on the incremental flux-linkage and the speed is calculated by differentiate the position. The initial rotor position estimator and start-up block is used to make a smooth sensorless starting process. The rotor position at standstill is estimated first, and then an open-loop control is used to speed up the motor. Moreover, in order to configure the registers in the control IC, a host register interface by serial communications is made. System constructers can setup registers of every block in the control IC by a Graphical User Interfaces (GUI) program in Matlab on PC. Fig. 3 shows the functional block diagram of the sensorless speed control IC.

B. Principle of Position Sensorless Estimation

The sensorless algorithm realized in the control IC uses the incremental values of flux linkage, the normalized back-EMF functions and the estimated back-EMF peak value to calculate incremental rotor position [6]. An internal closed-loop correction mechanism within this algorithm can correct rotor position estimation drift, which could due to quantization error of digital processing or measurement noise. Form [6], the incremental rotor position value can be obtained as

$$\Delta\hat{\theta}_e = \frac{P}{K_E}\frac{\Delta\psi_a e_b(\hat{\theta}_e)+\Delta\psi_b e_c(\hat{\theta}_e)+\Delta\psi_c e_a(\hat{\theta}_e)}{e_a(\hat{\theta}_e)e_b(\hat{\theta}_e)+e_b(\hat{\theta}_e)e_c(\hat{\theta}_e)+e_c(\hat{\theta}_e)e_a(\hat{\theta}_e)} \quad (1)$$

where P is the number of pole pairs, K_E is the back-EMF constant, $\Delta\psi_a$, $\Delta\psi_b$ and $\Delta\psi_c$ are defined as the flux linkage increments within the sampling interval of each phase, $\hat{\theta}_e$ is the estimated electrical rotor position, e_a, e_b and e_c are the normalized estimated back-EMF functions that vary with rotor position. For a sinusoidal back-EMF three-phase motor,

$$e_a(\hat{\theta}_e)e_b(\hat{\theta}_e)+e_b(\hat{\theta}_e)e_c(\hat{\theta}_e)+e_c(\hat{\theta}_e)e_a(\hat{\theta}_e)=-0.75 \quad (2)$$

is always valid, Hence, (1) becomes

$$\Delta\hat{\theta}_e = \frac{P}{-0.75K_E}[\Delta\psi_a e_b(\hat{\theta}_e)+\Delta\psi_b e_c(\hat{\theta}_e)+\Delta\psi_c e_a(\hat{\theta}_e)]. \quad (3)$$

The incremental electrical rotor position can also be calculated by

$$\Delta\hat{\theta}_e = \frac{T_{est}P\hat{E}}{K_E} \quad (4)$$

where T_{est} is the sampling time and $\hat{E}$ is the estimated peak back-EMF voltage. By combining (4) with (3), the estimated rotor position can be expressed as

$$\Delta\hat{\theta}_e = \frac{P}{-0.75K_E}\Big[\lambda\big(\Delta\psi_a e_b(\hat{\theta}_e)+\Delta\psi_b e_c(\hat{\theta}_e)+\Delta\psi_c e_a(\hat{\theta}_e)\big) -0.75(1-\lambda)T_{est}\hat{E}\Big] \quad (5)$$

where λ is a weighting factor. Assuming that the previous value of rotor position is known, an updated value of the rotor position estimation can be written as

$$\hat{\theta}_e(k)=\hat{\theta}_e(k-1)+\Delta\hat{\theta}_e(k). \quad (6)$$

After the position is estimated, the normalized back-EMF functions e_a, e_b and e_c are acquired from a sine look-up table and feed back to (5) to calculate the incremental position during the next sampling time. This forms a closed-loop correction mechanism which can correct the estimated rotor position at each sampling period.

C. Initial Position Detection and Start-up

The digital realization of a simple initial position detecting method has been developed to fulfill low-cost sensorless PMSM drive applications [7]. The principle of the method is based on the variation of the DC-link current response caused by the magnetic saturation of the stator core. When applying a DC voltage to the windings and producing a magnetic field aligned with the rotor field. The inductance of the wildings will decrease. Therefore, the rising time of the current will increase.

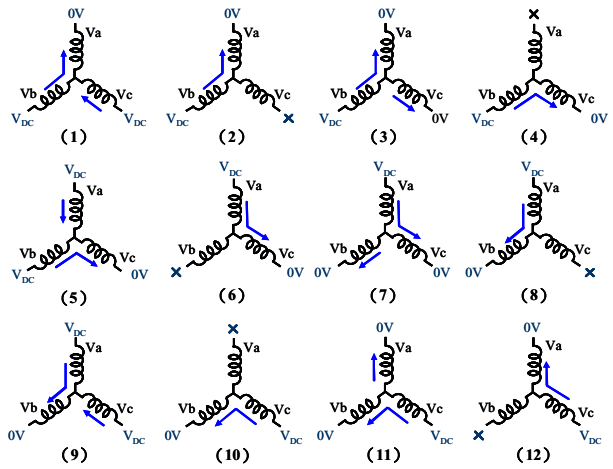

Fig. 4. Twelve voltage vectors used in the estimation process.

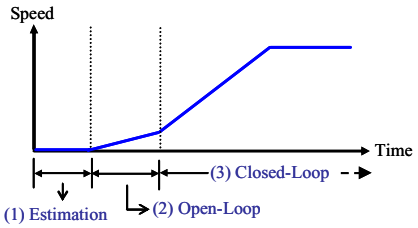

Fig. 5. The starting procedure.

The initial position of the magnetic pole can be determined by this rule. Fig. 4 shows the twelve voltage vectors applied in the initial estimation process. The voltage vector which maximum value of the DC-link current occurs represents where the permanent magnetic pole is located.

This method does not depend on the motor model, and is robust to motor parameter variations. Thus, it is suitable to the control IC for different applications. After the initial rotor position is determined by the proposed method, an open-loop starting method is used to speed up the motor until a given speed at which the rotor position estimation is sufficiently accurate. The starting procedure is shown in Fig. 5.

III. Digital Realization of the Functional Blocks

A. PWM Generator

A 12-bit digital three-phase complementary PWM generator with configurable PWM frequency and deadtime is implemented in the control IC. Three compare registers D_a, D_b and D_c are used to hold the modulating values, and they are constantly compared with the value of the timer counter TCNT as shown in Fig. 6. In this way, the output pulse is generated whose on (or off) duration is propositional to the value in the compare registers. The PWM generator can be set as asymmetric or symmetric mode, and once the PWM mode is selected the period is determined by the maximum value that the timer counter counts. Fig. 7 shows the hardware circuit of the PWM generator. In order to prevent short circuit in the DC-link voltage, the deadtime should be inserted when a transistor is turning on. A 7-bit deadtime generator is implemented, and for a 40 MHz system clock, it can generate a time delay from 0

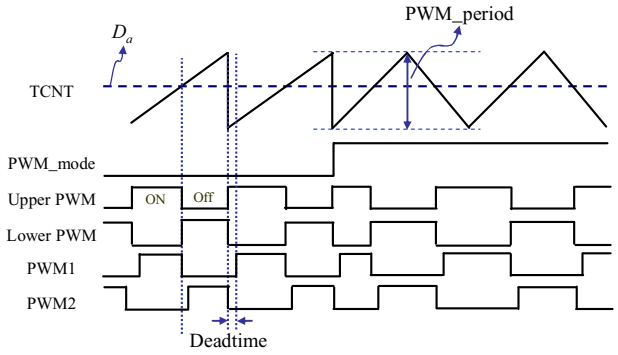

Fig. 6. Timing diagram of the PWM generator.

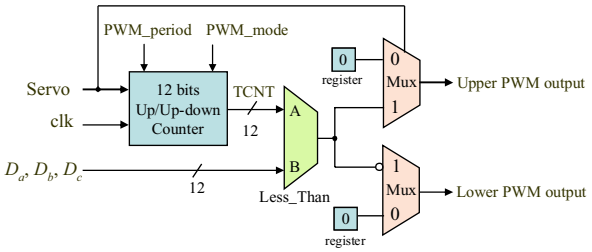

Fig. 7. Hardware circuit of the PWM generator.

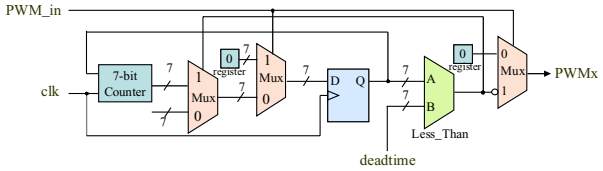

Fig. 8. Hardware circuit of the deadtime generator.

to 3.2 μs. Fig. 8 shows the hardware circuit of the deadtime generator.

B. Speed Controller

The speed loop is regulated by the digital PI control law, which is formulated as

$$u[k] = P[k] + I[k] \tag{7}$$

where the proportional term is

$$P[k] = K_{ps} \cdot [\omega^*(k) - \hat{\omega}(k)] \tag{8}$$

and the integral term is

$$I[k] = K_{is} \cdot [\omega^*(k) - \hat{\omega}(k)] + I[k-1]. \tag{9}$$

In (8) and (9), ω^* is the speed command, $\hat{\omega}$ is the feedback estimated speed and K_{ps}, K_{is} are the proportional and integral gain, respectively.

Fig. 9 shows the hardware structure of the PI controller. In order to reduce the whole chip size, a scheduling strategy for realizing the PI controller by using finite-state-machine is used. Only one multiplier and one adder are required to execute the PI control. The bit length of the PI controller is 16-bit, and K_{ps}, K_{is} are in Q-10 format. There are two saturation limiters in Fig. 9. The anti-windup limit at the output of the integrator is to

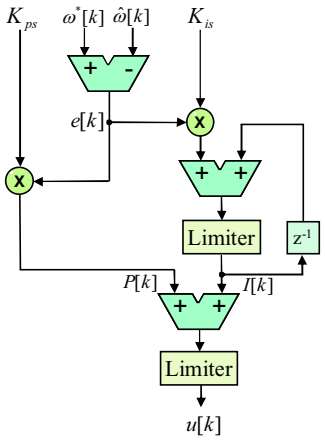

Fig. 9. Hardware structure of the PI controller.

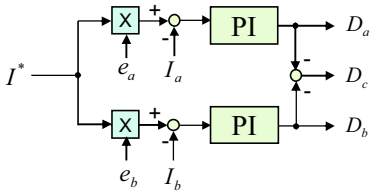

Fig. 10. Block diagram of current control loop.

prevent unwanted overshoot caused by the inertial property of the integrator. The output limit of PI the regulator is directly sent to the current regulator, thus it limits the maximum torque command.

C. Field-oriented Torque Controller

Fig. 10 shows the block diagram of field-oriented torque control loop with PI controller. The phase current should be aligned in phase with the back-EMF waveforms so as to achieve high torque/current ratio. The current command I^* from speed controller is first multiplied by e_a, e_b and e_c, where e_a, e_b and e_c are the estimated back-EMF functions that vary with rotor position, then PI controllers are used to generate the duty D_a, D_b and D_c, which will be sent to the PWM generator and sensorless position estimator.

In order to reduce the whole chip size, a scheduling strategy for realizing the field-oriented torque control is used by using finite-state-machine. Only one multiplier and one adder are required to execute the PI control law as shown in Fig. 11.

D. Sensorless Position Estimator

In the algorithm of the sensorless position estimation, the incremental flux linkage $\Delta\psi_a$ in (5) can be represented as

$$\Delta\psi_a = (v_{an} - Ri_a)\cdot T_{est} - L\Delta i_a . \tag{10}$$

The line-to-neutral voltage v_{an}, v_{bn} and v_{cn} is indispensable in computing the increased flux linkage. However, it is inconvenient and increases cost if we obtain them by feeding back line voltage and neutral voltage. Thus, in (10), v_{an} is calculated by the Duty of PWM generator alternatively. Then (10) can be written as

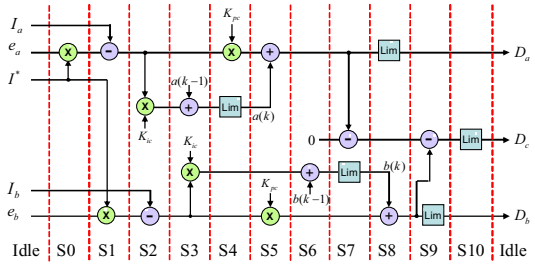

Fig. 11. The computation steps of the current control loop.

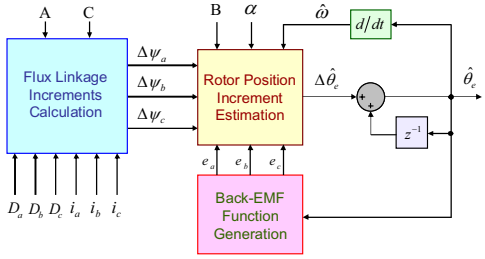

Fig. 12. Block diagram of sensorless position estimator.

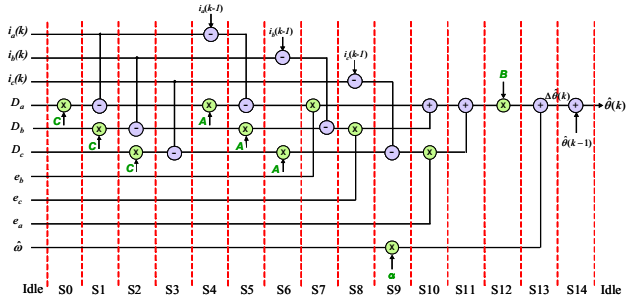

Fig. 13. The computation steps of the sensorless position estimator.

$$\Delta\psi_a = \frac{L}{K_c}\left[\left(\frac{K_c}{R}K_v D_a - i_{a(AD)}\right)\cdot\frac{R}{L}T_{est} - \Delta i_{a(AD)}\right] \tag{11}$$

where $v_{an} = D_a \times K_v$, D_a is the duty of phase a, K_v is the inverter gain, K_c is the current sensing gain and $i_{a(AD)}$ is the result read from A/D converter. The flux linkage $\Delta\psi_b$ and $\Delta\psi_c$ can be represented in the same manner as (11), then (5) becomes

$$\begin{aligned}\Delta\hat{\theta}_e = B\cdot\{&\left[(CD_a - i_{a(AD)})\cdot A - \Delta i_{a(AD)}\right]\cdot e_b \\ &+\left[(CD_b - i_{b(AD)})\cdot A - \Delta i_{b(AD)}\right]\cdot e_c \\ &+\left[(CD_c - i_{c(AD)})\cdot A - \Delta i_{c(AD)}\right]\cdot e_a\}+\alpha\hat{\omega}\end{aligned} \tag{12}$$

where $A = \frac{R}{L}T_{est}$ $\quad B = \frac{PL\lambda}{-0.75K_E K_c}$

$C = \frac{K_c}{R}\times K_v$ $\quad \alpha = PT_{est}(1-\lambda)$.

Fig. 12 shows the block diagram of the sensorless position estimator. The four parameters A, B, C and α should be calculated in advance to simplify the estimation process. The unit of $\Delta\hat{\theta}_e$ in (12) is in radians, and one revolution (0~2π) is

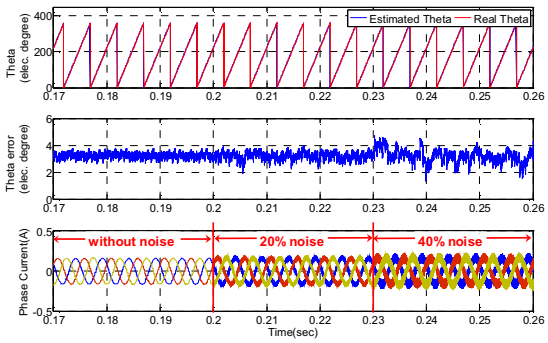

Fig. 14. Simulation results of the sensorless estimator with or without noise on the phase current.

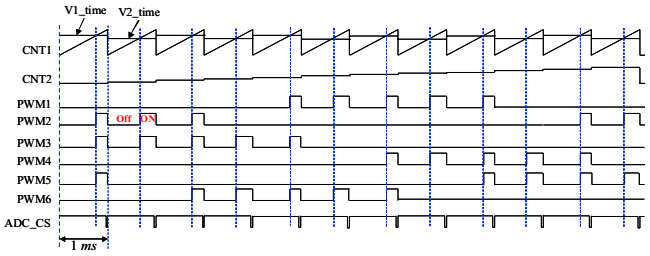

Fig. 15. The timing diagram of the twelve testing voltage vectors.

scaled to 0~8000 in the sensorless estimator. Thus, the parameter B and α should multiply by $8000/2\pi$. The normalized estimated back-EMF function e_a, e_b and e_c are generated from a 500 points sine-table with Q-9 format. Fig. 13 shows the computation steps of the sensorless position estimator. A binary format is used with a variable word-size approach [8], which permits to reduce truncation errors through calculation process. Besides, by using finite-state-machine method, only one 32-bit multiplier and one 32-bit adder are needed in the sensorless position estimator. To verify the sensorless estimator design in VHDL, a co-simulation with Matlab, PSIM and Modelsim is applied. A sensorless motor drive system is constructed in Matlab by linking PSIM and Modelsim. The parameters of tested motor and sensorless estimator in simulation are in Table I and Table II respectively. Fig. 14 shows the simulation result under 2000 rpm operation with or without noise on three phase current. The estimated error is about 3 electric degrees and the noise on phase current has little influence on the estimated results.

E. Initial Position Detection

The initial rotor position is detected by comparing the DC-link current after injecting twelve directions of voltage vectors successively. Fig. 15 shows the timing diagram of the testing voltage vectors. The time interval between the voltage vectors is 1 ms and the whole estimation time is about 12 ms. A counter CNT1 is used to count the time interval and two registers V1_time and V2_time are compared with it to determine the time duration of the voltage vectors. At the end of the voltage vector, the ADC is enabled by the signal ADC_CS to sample the peak current at DC-link side. No motor parameters have to be known during the detection process

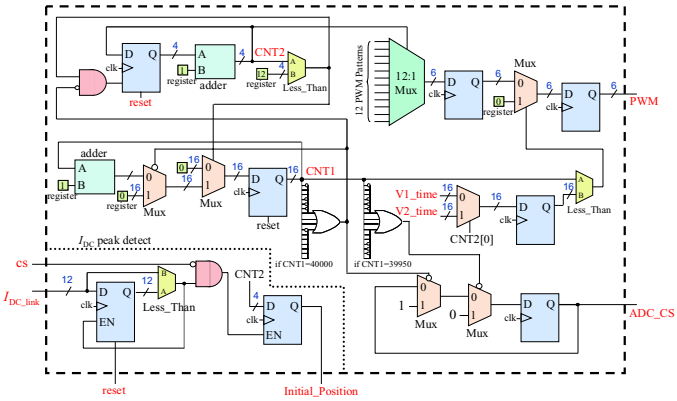

Fig. 16. Hardware circuit of the initial position detection.

Fig. 17. Graphical user interface in Matlab.

TABLE I
PARAMETERS OF THE TESTED MOTOR

3-phase permanent magnet synchronous motor	
Type	Y-connection, 12 poles
Rated voltage	12 V
Stator resistance	0.6 Ω
Stator inductance	0.102 mH
Back-EMF constant	0.423 mV/rpm
Rotor inertia	1.056×10^{-6} kg·m^2
Mech. time constant	0.27 sec

TABLE II
PARAMETERS SETTING FOR SIMULATION

	Real value	Implement value	Error
A	0.049	100 (Q11)	0.39%
B	0.1889	193 (Q10)	0.23%
C	6.347	53 (Q3)	0.38%
α	0.0064	105 (Q14)	0.14%

except that the output time of the voltage vectors should be determined by user. Fig. 16 shows the hardware structure of the initial position detection circuit.

IV. EXPERIMENTAL RESULTS

The proposed control IC has been realized by an Altera EP1C12F256C8, and the total logic cells are about 5550 of total 12060 (46%). The designed IC can operate at 40MHz system clock, and all control parameters are configured via the serial communication port by a host PC. A program designed by GUI in Matlab is used to setup the registers as shown in Fig. 17.

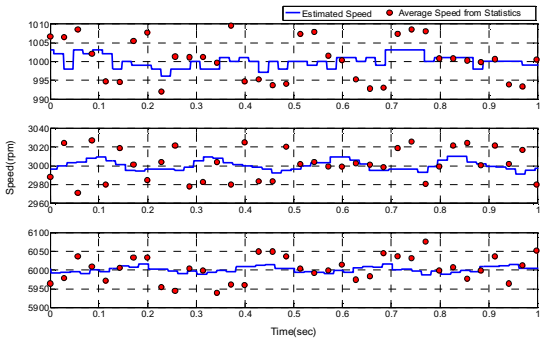

Fig. 18. Steady-state speed response.

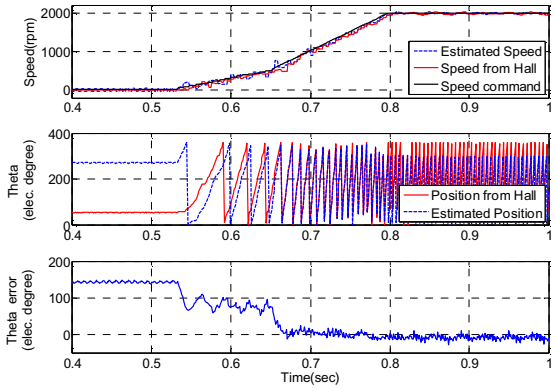

Fig. 19. Start-up response from standstill to 2000 rpm.

To test the performance of the servo controller, a DVD spindle PMSM is driven under 20 kHz PWM frequency. The sampling rate of the current loop and speed loop are 20 kHz and 2 kHz respectively. The estimated speed is compared with the speed calculated from linear Hall sensor signals. Fig. 18 shows the steady state speed response under speed command 1000, 3000 and 6000 rpm. The estimated speed is compared with the statistics of the average speed calculated from linear Hall sensor in a random period of time. Fig. 19 shows the start-up response form standstill to 2000 rpm, where the open-loop control scheme is used below 500 rpm. It can be seen that the error between the real and estimated position is large at low speed, and it decreases during the accelerating procedure. By detecting the motor position at standstill, the start-up process is smooth without temporary reversing rotation. Fig. 20 shows the response of a ramp command from 500 to 7000 rpm with the acceleration rate of 40 rpm per millisecond. Fig. 21 shows the response of a step command from 5500 to 6000 rpm where the transient time is about 30 ms without overshoot.

V. Conclusion

This paper presents a digital sensorless speed control IC, which can provide a simple and feasible solution for PMSM motor drives from standstill to full speed operation with fast and robust responses. The proposed control IC has been realized by an Altera EP1C12F256C8 FPGA device, and the total logic cells are about 5550 of total 12060 (46%). This provides the possibility of manufacturing low-cost digital

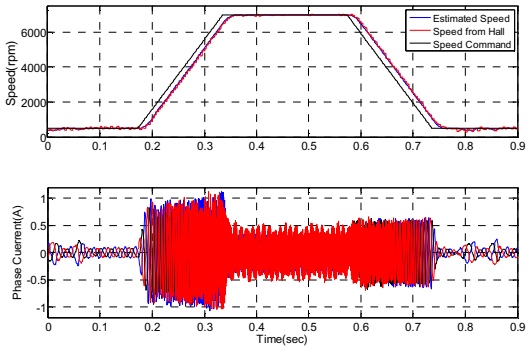

Fig. 20. Ramp response from 500 to 7000 rpm.

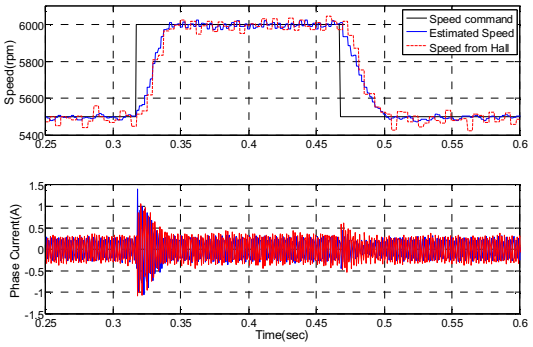

Fig. 21. Step response from 5500 to 6000 rpm.

sensorless PMSM/BLDC motor control ICs using ASIC technology. Experiment results show that the starting process is smooth without reversing rotation, and the speed ripple is about 1% under steady-state operation. By using the GUI interface, it is easy to setup the parameters in the control IC, and is suitable to different motors. A fully digital speed sensorless control system for PMSMs from standstill to high speed has been achieved by the proposed control IC.

References

[1] P. P. Acarnley and J. F. Watson, "Review of position-sensorless operation of brushless permanent-magnet machines," *IEEE Trans. Ind. Electron.*, vol. 53, no. 2, pp. 352-362, Apr. 2006.

[2] T. M. Jahns, "Motion control with permanent-magnet AC machines," *IEEE Proc.*, vol. 82, no. 8, pp. 1241-1252, Aug. 1994.

[3] J. Moutinho, R. E. Araujo, and V. Leite, "Low cost control and monitoring motion control ICs," *IEEE Electrotechnical Conf.*, pp. 1138-1141, May 2006.

[4] L. Charaabi, E. Monmasson, M. Nassani, and I. Slama-Belkhodja, "FPGA-based implementation of DTSFC and DTRFC algorithms," *Industrial Electronics Society*, *IECON* 2005. 32nd, pp. 245-250, Nov. 2005.

[5] Zhaoyong Zhou, Tiecai Li, T. Takahashi, and E. Ho, "FPGA realization of a high-performance servo controller for PMSM," *IEEE APEC,* vol. 3, pp. 1604-1609, 2004.

[6] L. Ying and N. Ertugrul, "A novel, robust DSP-based indirect rotor position estimation for permanent magnet AC motors without rotor saliency," *IEEE Trans. Power Electron.*, vol. 18, no. 2, pp. 539-546, Mar. 2003.

[7] Yen-Chuan Chang and Ying-Yu Tzou, "A New Sensorless Starting Method for Brushless DC Motors without Reversing Rotation," *IEEE PESC Conf.*, in press, 2007.

[8] S. Ferreira, F. Haffner, L. F. Pereira, and F. Moraes, "Design and prototyping of direct torque control of induction motors in FPGAs," *Integrated Circuits and Systems Design*, *SBCCI Proc*, pp. 105-110, 2003.