Implementation of Real Time Embedded Control System for Interior Permanent Magnet Motors

Ahmed M. Omara, Mohamed K. El-Nemr, Essam M. Rashad*
Dept. of Electrical Power and Machines Engineering
Faculty of Engineering, Tanta University
Tanta, Egypt
*emrashad@ieee.org

***Abstract*— Aiming at achieving a low cost and precise closed loop speed control of interior permanent magnet synchronous motor (IPMSM), this paper introduces an implementation of cost effective real time embedded control system. Scalar (constant *V/f*) closed loop speed control strategy was implemented using a conventional PI regulator in conjunction with space vector pulse width modulation (SVPWM) technique. The motor speed has been obtained using incremental shaft encoder with reasonable resolution and price. Due to the necessity of accurate position and current measurements, fast PI controller actions, and digital implementation of SVPWM technique, a suitable digital signal controller (DSC) has been used as the computing and control center. Optimized embedded software was developed using C programming language combined with assembly instructions. Several experimental results show that the proposed drive system has a worthy dynamic response and precise tracking of the speed trajectory in a wide speed range, including zero speed without losing synchronization.**

Keywords*— Line Start IPMSM, Space Vector Modulation, V/f Control, Digital Signal Controller, Embedded Systems

I. Introduction

Due to distinctive advantages such as high power and torque density, high efficiency, and lower maintenance, interior permanent magnet synchronous motors (IPMSM) are favorably chosen for a lot of applications, e.g., traction drives for electric vehicle, fans and pumps [1–2]. The popularity of such motors is increasing because they overcome the limitations of both the induction motor and conventional synchronous motor ac drives [2–5]. Modern ac drives require a variable frequency drive (VFD) system to control speed, position, and torque as shown in Fig. 1. Continuous and extensive research of IPMSM has been progressing by leaps and bounds for the past two decades. However, most of the practical implementations of IPMSM drive systems use a high price DSP-based controller kit installed on a personal computer (PC) to obtain the control actions, in conjunction with a very high level programming language supported by model-based design software packages [6–10]. The objective of this paper is to provide a low cost and precise closed loop speed control system for a line start IPMSM. A low price digital signal controller (DSC) is used. Low cost hardware provides high performance only using sophisticated implementation techniques, where know-how presents the added value compared to hardware expenses.

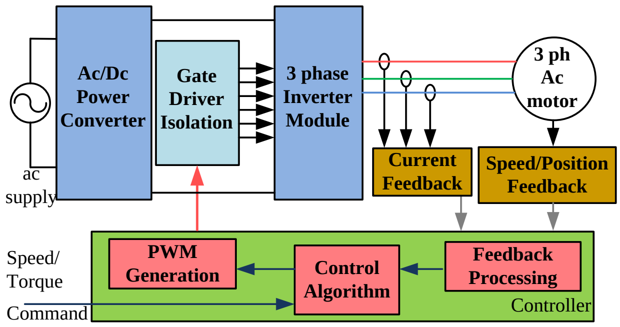

Fig. 1. Variable frequency drive system

The control system is designed to maintain both stable and high efficient operation of the drive system. Embedded C program is developed and optimized using shareware software utilities. The drive system is experimentally implemented, verified, and validated over a wide speed range including the zero speed.

II. Model and Control of IPMSM

A. *Model of line start IPMSM*

Line start IPMSM is basically a salient pole synchronous machine. Unlike conventional synchronous machine, the saliency is created due to the presence of low permeability permanent magnets in the rotor while air gap is smooth. Such a motor has a squirrel cage rotor to provide starting torque. However, the drive system has to be designed to overcome the magnet brake torque at line starting [11–12]. In absence of cage winding IPMSM can be started using intelligent variable frequency power inverters. Fig. 2 is an example of line start IPMSM adopted in this research. In this type, the quadrature axis inductance is higher than the direct axis inductance i.e.:

$$L_q > L_d \tag{1}$$

This is quite contrary to that of wound rotor salient pole synchronous machine. Such inequality in inductances has a direct consequence on the machine operation. In addition to synchronous torque, a reluctance torque component exists.

At steady-state the cage is ineffective and operation is typically of salient pole synchronous machine. For a balanced

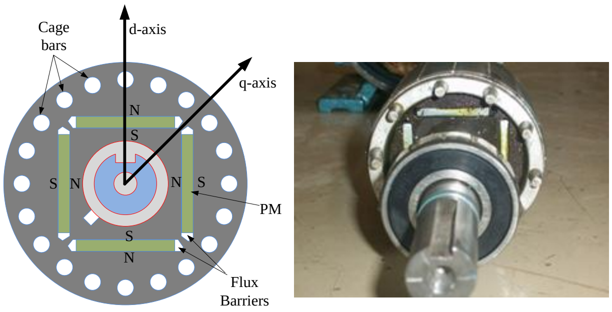

Fig. 2. Line start IPMSM rotor structure

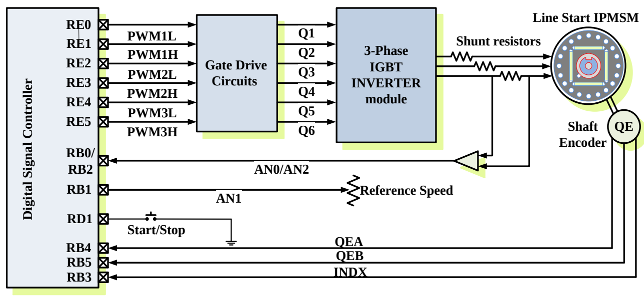

Fig. 3. Closed loop speed control system

three phase condition, the developed power P_d may be expressed as follows [11]:

$$P_d = 3\left(\frac{2}{P}\right)\left\{\frac{V_{ph}E_m}{X_d}\sin\delta + \frac{V_{ph}^2(X_d - X_q)}{2X_d X_q}\sin 2\delta\right\} \qquad (2)$$

where, V_{ph} is the stator supply voltage per phase, E_m is the excitation voltage per phase, X_d and X_q are the direct and quadrature (d-q) axis reactances, respectively and δ is the phase shift between V_{ph} and E_m voltage phasors, which is known as load or torque angle.

In *dq0* reference frame, developed torque can be expressed as follows [11]:

$$T_{em} = \frac{3}{2}P\left\{\underbrace{\Psi_{md} I_{qs}}_{Excitation} - \underbrace{(L_d - L_q) I_{qs} I_{ds}}_{Reluctance}\right\} \qquad (3)$$

where, P is number of poles, Ψ_{md} is the permanent-magnet flux linkage, L_d & L_q are the d-q axis inductances, respectively, I_{ds} & I_{qs} are the d-q axis components of the IPMSM stator current, respectively. The first term of (3) is due to PM's field and the second is due to reluctance variation.

B. Motor Control System

Similar to most three phase ac machines, IPMSM has two distinctive control methods; scalar and vector control. In the scalar method of speed control, the motor speed is controlled via variable frequency voltage source inverter (VSI). Scalar control has an acceptable performance for many applications [13]. However, a large sudden change in the reference speed may result in unstable operation or loss of speed tracking.

In order to achieve higher performance, vector control technique are earning wide acceptance. However it was found that there is a major limitation for real-time implementation of these algorithms, especially for nonlinear cases. This is owed to their high computational burden which limits sampling frequency for digital implementation. Accordingly, most of available research work is of either simulation nature or of very high cost experimental implementations [6–10], [13].

A simplified system block diagram of the adopted line start IPMSM is shown in Fig. 3. This embedded hardware topology consists of DSC, gate drive circuits, IGBT power module, incremental shaft encoder, current measurement circuit, potentiometer for reference speed input and bush button for starting and stopping the controller.

The torque-speed characteristics of the drive system depend on the type of control. It is necessary to vary the voltage and frequency to meet the torque-speed requirements. An example of variable frequency control is the scalar constant *V/f* speed control. There are two distinct regimes of operation, constant torque below base speed and constant power above base speed. The base speed is defined as the maximum speed at which rated torque can be developed with rated current flowing without exceeding the maximum terminal voltage available from the inverter.

III. EMBEDDED CONTROL SYSTEM IMPLEMENTATION

Embedded system is synonym for embedded computer based on microcontroller or microprocessor. It operates as human-interactive, network-interactive and/or, autonomous. It is deployed effectively in diverse environments. However, it is sold into a competitive and cost-conscious market. To implement a reliable real time embedded control system for IPMSM, both hardware and software have to be designed precisely and tuned for energy consumption, code size, execution time, weight, dimensions, and cost.

A. Hardware Synthesis

1) Power Circuit

The drive system of the adopted line start IPMSM is shown in Fig. 4.

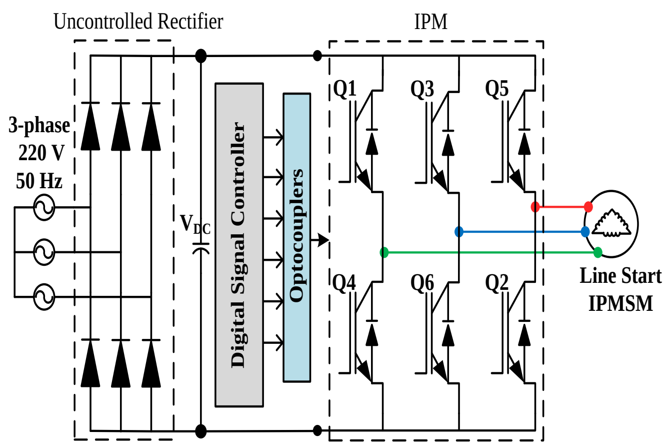

Fig. 4. Power Circuit

It consists of an IGBT power module, three phase bridge rectifier, and a capacitor connected in the dc-link. The uncontrolled rectifier module converts the fixed input 222V ac line-to-line voltage into a fixed 300V dc-link voltage. The IGBT inverter module is controlled to convert the dc bus voltage to an output ac variable voltage with variable frequency.

Intelligent power module (IPM), a production of Mitsubishi is chosen as the key component of the inverter part. PM25RSB120 is a 1200V, 25 Ampere Intelligent Power Module. It is an isolated base module designed for power switching applications operating at frequencies up to 20 kHz. Its built-in control circuits provide optimum gate drive and protection for the IGBT and free-wheel diode power devices. To ensure the reliability of the system, the PWM signals between DSC and IPM should be isolated.

2) *Control Topology*

The control system design is shown at Fig. 5. It consists of the DSC and other interfaces connected to different DSC peripherals. DSC is a hybrid of MCU and DSP. Inclusively DSC combines the processing power of a DSP and the functionality of a MCU with a flexible set of peripherals to create a cost-effective solution. In this way, DSCs are excellent at executing the complex, high-speed mathematical functions required by many real time control systems. The dsPIC30F2010 family member is a 28-pin 16-bit DSC specifically designed for low-cost/high efficiency motor control applications.

The reference speed input signal is obtained from a potentiometer connected to a 10-bit analog to digital converter (ADC). There is a reset button for microcontroller reset (MCLR) signal. Start/Stop button connected on a general purpose input/output (GPIO) digital port. Motor position and direction is obtained from the incremental encoder signals connected to quadrature encoder interface (QEI) module. For protection purposes the motor currents are measured and interfaced with DSC via three ADC pins. Output SVPWM pulses from motor control pulse width modulation (MCPWM) peripheral is interfaced with the power circuit and isolated using photoelectric optocoupler circuits.

B. *Software Design*

1) *Main Finite State Machine*

Finite state diagram is shown in Fig. 6. It illustrates the behavior of the motor's control software. At Power-on Reset, the software initializes all the software variables and enables all the peripherals to be used by the embedded control system. Afterwards, the software enters the Motor Stopped state and remains there until a start command is executed from the external push button (Start/Stop pressed). When Start/Stop button is pressed, the variables used for controlling the motor are initialized, the timer counters are also initialized to zero, the interrupt flags are cleared and the interrupts are enabled.

Once the variables have been initialized, the software enters the Motor Running state, and all other activities within the state machine are performed by six interrupt service routines.

In embedded systems programming, an interrupt handler, also known as an interrupt service routine (ISR), is a callback subroutine in microcontroller firmware, operating system or device driver whose execution is triggered by the reception of an interrupt request [14–15]. Interrupt handlers have a multitude of functions, which vary based on the reason the interrupt was generated and the speed at which the interrupt handler completes its task. The motor control system interrupt service routines are summarized as shown in TABLE I. It indicates when they are called and provides a brief description of the operations executed in each particular ISR. If Start/Stop button is pressed while the motor is running, all the interrupts are disabled and the motor is stopped.

TABLE I
Summary of interrupt service routines

ISR	Calling frequency	Operations performed
A/D	20 kHz	Reads new reference speed value
PWM	5 kHz	Generates sine wave using SVM
T1	1 kHz	Calculate the actual speed
QEA	Every QEA transition	Calculates rotor position
QEB	Every QEB transition	Determines rotor mechanical direction
INDX	Every INDX transition	Resets rotor position counter

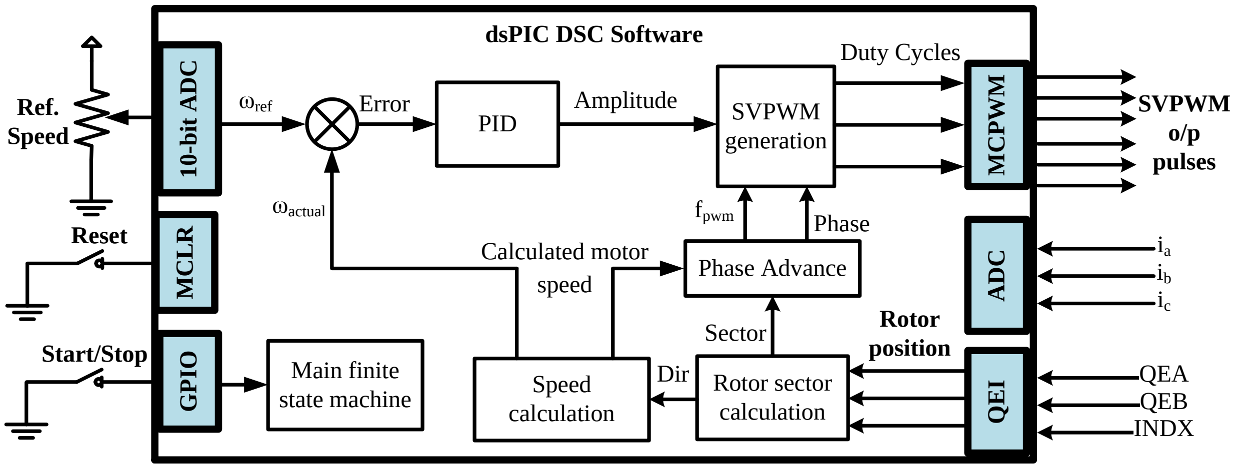

Fig. 5. Control System Design

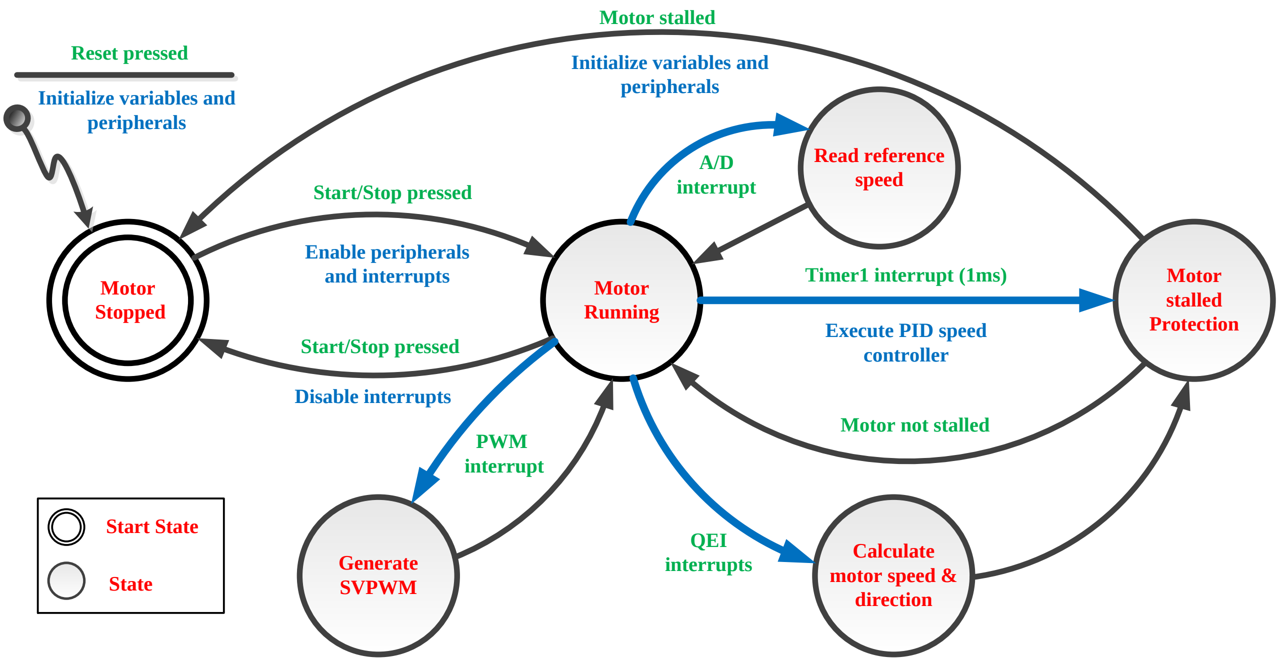

Fig. 6. Finite state machine

2) Position Measurement

Quadrature shaft encoders (also known as incremental encoders or optical encoders) enable closed loop control. They are used for position and speed detection of the motor. The adopted DSC has a quadrature encoder interface (QEI) module, which provides an interface to incremental optical encoders. The QEI module accepts channel A, channel B and index signal from the incremental encoder and stores the accumulated count pulses in a dedicated 16-bit time base. This QEI module allows obtaining signed velocity and relative rotor position information from the motor. For precise position measurement, each signal edge of the QSE has been captured using the designed software.

3) PID Digital Implementation

PID controller is used for speed regulation. It is executed in the Timer1 interrupt (T1 ISR) every 1 millisecond. Motor measured speed is subtracted from the reference speed to determine the speed error, the sign of which decides if the motor must accelerate or decelerate. To ensure smooth operation of the motor, the error value is parsed into proportional, integral and derivative (PID) components to produce a composite output that is used to compensate for the speed error. The PID controller implementation takes advantage of the multiplier and accumulator (MAC) instruction of the dsPIC DSC for fast execution. The trapezoidal method of integration is used. Limits are set to limit the output of the PID controller. The formula used to generate the controller output is as follows:

$$\begin{aligned} U(n) = {} & U(n-2) + K_1 e(n) \\ & + K_2 e(n-1) + K_3 e(n-2) \end{aligned} \tag{4}$$

where, $U(n)$ is the current output of the PID controller at n^{th} sample, $U(n\text{-}2)$ is the output at $(n\text{-}2)^{th}$ sample, $e(n)$ is the error at n^{th} sample, $e(n\text{-}1)$ is the error at $(n\text{-}1)^{th}$ sample and, $e(n\text{-}2)$ is the error at $(n\text{-}2)^{th}$ sample. K_1, K_2, and K_3 are weighting parameters which depend on PID controller parameters.

When first tuning a controller, the K_i and K_d gains should be set to zero. The K_p gain can then be increased until the system responds well to set-point changes without excessive overshoot or oscillations. After a reasonable K_p gain is selected, the K_i gain can be slowly increased to force the system error to zero. If oscillation occurs, reducing the K_i gain and increasing the K_p gain usually solves the problem. The K_d gain can speed up system response. However, it must be used carefully because it can produce large output swings, which potentially cause mechanical damage on the motor. The PI gains of the controller is obtained empirically, and with the values of K_p=0.45 V/rad/s, K_i=0.01 V/rad, K_d=0 V/rad/s^2.

4) Generation of SVPWM

There are several techniques for pulse width modulation (PWM). However, the most commonly used PWM techniques are sinusoidal PWM, and space vector pulse width modulation (SVPWM). Easy implementation of SPWM technique and minimum required online computations lead to its wide spread. However this modulation algorithm has some drawbacks e.g., low dc bus utilization, high switching losses, and high total harmonic distortion (THD) [16].

The SVPWM technique is a sophisticated averaging algorithm. It gives 15% more voltage output compared to SPWM algorithm. Thereby, it increases the dc bus supply voltage utilization, and also minimizes the THD as well as switching losses [16–17]. All the possible switching states of the VSI are shown in Fig. 7.

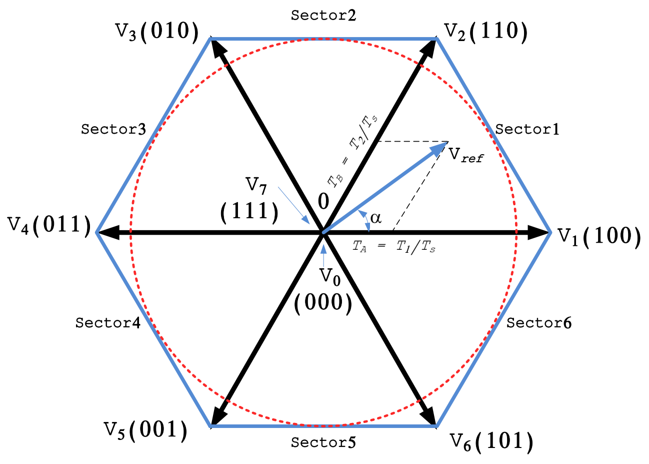

Fig. 7. Space vector hexagon

Vectors V_1 through V_6 are called active vectors. V_0 and V_7 are called the inactive or zero vectors. Each switching state can be represented as a voltage vector in space, as shown in Fig. 7. The entire space is divided into six equal sized sectors of 60°. Each sector is bounded by two active vectors. Zero vectors are located at the hexagon origin.

To digitally implement SVM a reference voltage V_{ref} is sampled with a very high frequency f_{PWM}. This frequency is high enough so as not to generate audible noise due to switching. The sampling time T_s for V_{ref} is determined from f_{PWM}, where $T_s = 1/f_{PWM}$. The reference vector is then synthesized using a combination of the two adjacent active switching vectors and one or both of the zero vectors. Various strategies of selecting the order of the vectors and which zero vector(s) to use exist. Strategy selection will affect the harmonic content and the switching losses. SVPWM allows any resultant vector be represented by the sum of the components of the two adjacent vectors. In Fig. 7, V_{ref} is the desired resultant. It lies in the sector1 between V_1 and V_2. During a given PWM period T_s, V_1 is the output for T_1 /T_s, V_2 is the output for T_2 /T_s, and the average for this period is V_{ref}.

IV. MOTOR SPECIFICATIONS

The specifications of the motor under control are presented in TABLE II. The stator consists of 24 slots, three phase single layer windings wound by mush winding type. The stator windings are delta connected. The rotor has four interior permanent magnets which forms a 4-pole machine. Squirrel cage used to provide self-starting at constant supply frequency.

TABLE II
MOTOR SPECIFICATIONS

Description	Value	unit
Rated Output Power	0.5	Hp
Rated Line-to-Line Voltage	220	V(rms)
Rated Line Current	2.68	A(rms)
Rated Speed	1500	r/min
Number of Poles	4	-
Rotor Inertia	0.00145	kg.m/sec^2

V. EXPERIMENTAL RESULTS AND DISCUSSION

In order to assess the suggested system, experimental measurements have been obtained at different operating conditions. Fig. 8 shows the experimental setup. A sample of measured output line-line voltage waveform is shown in Fig. 9 that has an rms voltage of 78.3V, and a frequency of 15.6Hz.

A. *Starting Behavior*

Fig. 10 shows motor starting behavior from standstill to the rated speed. It is depicted that the motor starts smoothly with acceptable speed overshoots and without losing synchronism. Suitable current is also drawn from supply.

B. *Response to Step Change in Reference Speed*

System response to a step change in speed from 150 to 500r/min is shown in Fig. 11. For a large sudden step change in speed from 500 to 1500 r/min, speed oscillations shown in Fig. 12 are limited due to the presence of cage winding. Speed changes following preset speed profile are shown in Fig. 13. It is noted that the actual speed has a fast transient response, while the motor draws acceptable currents over a wide speed range including zero speed.

Fig. 8. Drive system rig

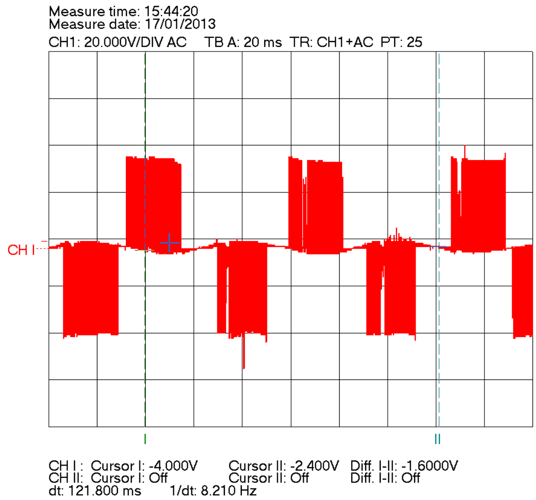

Fig. 9. Sample of measured line-line voltage

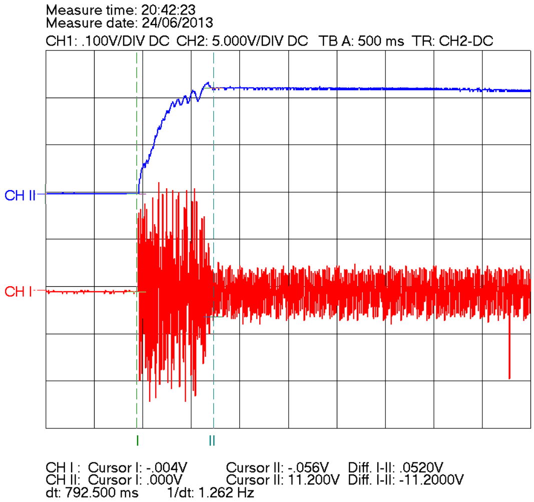

Fig. 10. Starting from standstill to rated speed (2A/div)

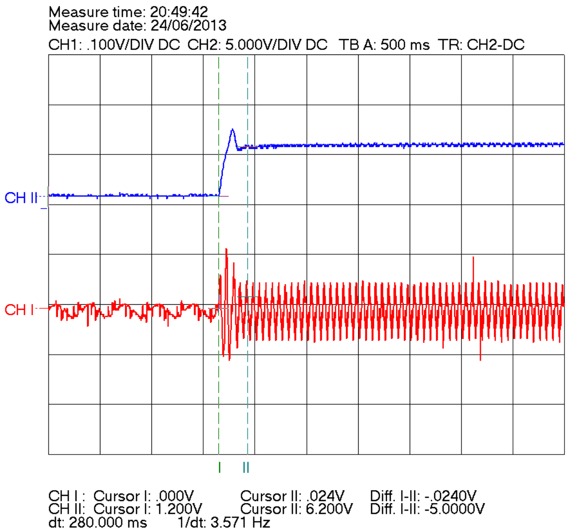

Fig. 11. Speed change from 150 r/min to 500 r/min

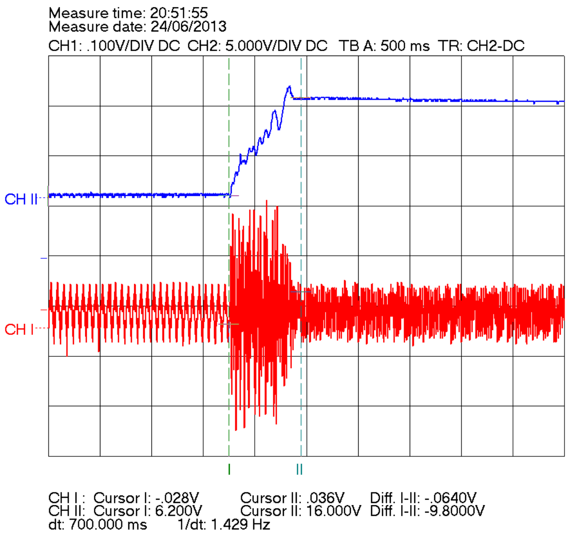

Fig. 12. Speed change from 500 r/min to 1500 r/min

Fig. 13. Speed profile over a wide speed range in both directions of rotations

C. Software Performance Measures

Software performance measures (e.g., memory usage and CPU utilization) show how the software is optimally designed. Program and data memory usage is 1771 and 56 bytes respectively, which means that there is enough memory available for code expansion. CPU utilization is also calculated found to be 65%, this means that the CPU is idle for 35% of time so that more tasks can be added.

CONCLUSIONS

This paper has presented a detailed cost effective implementation of a real time embedded closed loop control system for a line start interior permanent magnet synchronous motor (IPMSM). The system uses a digital signal controller (DSC) as a processor. DSC has both the advantages of a microcontroller unit (MCU) and also the computational power of a digital signal processor (DSP). Both hardware and software have been designed precisely and optimized for energy consumption, code size, execution time, weight, dimensions, and cost. Motor position has been measured using a low price quadrature shaft encoder (QSE) with appropriate resolution. Scalar *V/f* control technique has been implemented, where variable voltage and frequency were obtained using space vector pulse width modulation (SVPWM) technique. Motor line currents have been measured and a proper control action has been considered for protecting the motor from drawing high currents. To evaluate the performance of the proposed real-time embedded control system, experimental works are carried out at wide range of speed operation including zero speed. The experimental results show that motor speed tracks the specified reference speed trajectories. The speed has been controlled in a wide speed range including zero speed without losing synchronization.

REFERENCES

[1] T. Ding and N. Takorabet, “Design and analysis of different line-start PM synchronous motors for oil-pump applications,” Magnetics, IEEE Transactions on, vol. 45, no. 3, pp. 1816–1819, 2009.

[2] M. Ehsani, G. Yimin, and S. Gay, “Characterization of electric motor drives for traction applications,” Industrial Electronics Society, 2003. IECON '03. The 29th Annual Conference of the IEEE, vol. 1, Nov. 2003, pp. 891 – 896.

[3] M. G. Kahrisangi, A. H. Isfahani, S. Vaez-Zadeh, and M. R. Sebdani, “Line-start permanent magnet synchronous motors versus induction motors: A comparative study,” Frontiers of Electrical and Electronic Engineering, Nov. 2012.

[4] M. Zeraoulia, M. E. H. Benbouzid, and D. Diallo, “Electric Motor Drive Selection Issues for HEV Propulsion Systems: A Comparative Study,” IEEE Transactions on Vehicular Technology, vol. 55, Issue 6, Apr. 2006, pp. 1576 – 1764.

[5] Deshpande, U.S., 2003. “Recent Advances in Materials for Use in Permanent Magnet Machines: a Review,” Proceedings of IEEE Electric Machines and Drives Conference, IEMDC'03, vol. 1, Jun. 2003, pp. 509-515.

[6] S. Jung, S. Member, and J. Hong, “Current Minimizing Torque Control of the IPMSM Using Ferrari ’ s Method,” Power Electronics, IEEE Transactions on, vol. 28, no. 12, pp. 5603–5617, 2013.

[7] a. a. Hassan and E. G. Shehata, “High performance direct torque control schemes for an IPMSM drive,” Electric Power Systems Research, vol. 89, pp. 171–182, Aug. 2012.

[8] W. Peters, O. Wallscheid, and J. Bocker, “A precise open-loop torque control for an interior permanent magnet synchronous motor (IPMSM) considering iron losses,” IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society, pp. 2877–2882, 2012.

[9] G. Andreescu, S. Member, C. I. Pitic, and F. Blaabjerg, “Enhancement for Wide Speed Range Sensorless Direct Torque Control of IPMSM Drives,” Energy Conversion, IEEE Transactions on, vol. 23, no. 2, pp. 393–402, 2008.

[10] M. N. Uddin and M. A. Rahman, “High-Speed Control of IPMSM Drives Using Improved Fuzzy Logic Algorithms,” IEEE Transactions on Industrial Electronics, vol. 54, no. 1, pp. 190–199, Feb. 2007.

[11] S. Rabbi and M. Rahman, “Analysis of starting and synchronization process for line start IPM motors,” Electrical & Computer Engineering (ICECE), pp. 311–314, 2012.

[12] S. Morimoto, M. Sanada, and Y. Takeda, “Performance of PM-assisted synchronous reluctance motor for high-efficiency and wide constant-power operation,” IEEE Transactions on Industry Applications, vol. 37, no. 5, Sept./Oct. 2001, pp. 1234 – 1240.

[13] J.-I. Itoh, N. Nomura, and H. Ohsawa, “A comparison between V/f control and position-sensorless vector control for the permanent magnet synchronous motor,” Proceedings of the Power Conversion Conference-Osaka 2002 (Cat. No.02TH8579), vol. 3, pp. 1310–1315, 2002.

[14] L. Leyva-del-Foyo, “Integrated Task and Interrupt Management for Real-Time Systems,” ACM Transactions on Embedded Computing Systems (TECS), vol. 11, no. 2, p. Article No. 32, 2012.

[15] H. Kopetz, “Real-time systems: design principles for distributed embedded applications,” p.228–229, 2011.

[16] J. Pradeep and R. Devanathan, “Comparative analysis and simulation of PWM and SVPWM inverter fed permanent magnet synchronous motor,” Emerging Trends in Electrical Engineering and Energy Management (ICETEEEM), p. 299,305, 2012.

[17] A. Gupta and A. Khambadkone, “A general space vector PWM algorithm for multilevel inverters, including operation in overmodulation range,” Power Electronics, IEEE …, vol. 22, no. 2, pp. 517–526, 2007.