A DSP-Based Fully Digital PMSM Servo Drive Using On-Line Self-Tuning PI Controller

Bin Zhang, Yaohua Li, Yansheng Zuo
Institute of Electrical Engineering, Chinese Academy of Sciences, Beijing 100080, P. R. China
Phone: 86-10-62558089 Fax: 86-10-62541870
Email: zhangbin@piano.iee.ac.cn

Abstract-*A DSP-based fully digital drive for AC servo application is presented in this paper, in which a high performance DSP is used and all the control loops (speed and current) are implemented with software. Taking advantage of the fast calculation ability of DSP, this paper also designs an on-line self-tuning PI controller, which can detect the mechanical inertia time of this drive and realize the automatic adjustment of speed loop PI controller parameter. The simulation and experiment results have demonstrated that the drive has a good dynamic and robust performance.*

Key words: Servo drive, PMSM, self-tuning control, DSP

1. Introduction

In many operations, some parameters of servo drives vary as the load and the operation condition change. Variation of the parameters such as mechanical inertia time and load torque will degrade the drive performance. In some cases when the mechanical inertia time of the motor changes to some extent, servo drives cannot maintain good dynamic performance and may even lose their stability. In the past, it was very difficult to solve the problem because the overall control, especially the current control, was realized with analogy circuits. Such an analogy control cannot easily realize complicated control strategy such as parameter estimation and self-tuning control.

Recently, digital implementation of the controllers has become a trend in the area of the servo drive because high-speed processors such as DSPs have been developed which allow high accurate computation within a short sampling time. With DSPs, most of the control function can be realized by software instead of hardware. Software control can not only make the servo drive more reliable but also realize parameter estimation and self-tuning control much more easily than by hardware.

By employing microprocessors, lots of methods in the area of self-tuning control have been proposed in technical publications over the recent years. Roughly speaking, there are two categories to auto-tune the PI or PID gains so that the controllers can adapt to the varying conditions: model-based analytical method [1-6] and model-free method [7].

Model-based analytical method is based on an assumed mathematical model and some assumed conditions. Because of the system non-linearity, torque disturbance, noises etc., the assumed mathematical model is often not correspondent to the real applications very well. In these cases, the self-tuning controller cannot achieve good performance. What is more, model-based analytical method is a complex process and it needs a lot of calculation which cannot be implemented easily with microprocessors.

On the model-free method, only a few practical self-tuning methods have been reported. In [7], the mechanical inertia is obtained by analyzing the overshoot of the speed step response. But in this method, the load disturbance could affect the accuracy of the inertia calculation. The ideal speed reference is also difficult to obtain in many applications.

This paper describes a fully digital servo drive with DSP. Taking advantage of current saturation limit, this paper designs a new self-tuning PI controller.

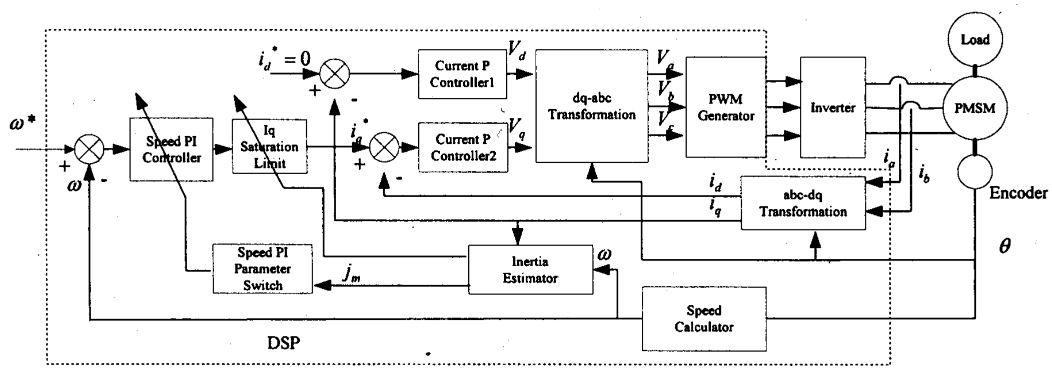

Fig.1 Block diagram of the PMSM drive

The controller detects the mechanical inertia variation periodically. If the inertia change exceeds a certain value, the controller will calculate the mechanical inertia and adjust the speed PI controller parameter according to the estimated inertia. The effectiveness is well demonstrated through some comparative simulations and experiments.

2. Fully Digital Control of Servo Drive

Fig. 1 shows the complete block diagram of the electric drive.

The speed control is achieved by a software self-tuning PI controller, which is described as:

$$\Delta\omega(n)=\omega^*(n)-\omega(n) \tag{1}$$

$$I^*_q(n+1)=K_p\Delta\omega(n)+K_i \bullet \sum_{i=0}^{n}\Delta\omega(n) \tag{2}$$

Where the input of speed controller $\omega^*(n)$ and $\omega(n)$ denote the speed command and speed feedback of PMSM respectively. K_p and K_i denote the proportional and integral gain of the speed PI controller respectively. The speed controller output I_q^* is the q-axis current command.

By employing the field-oriented control, the d-axis current command

$$I_d^*=0 \tag{3}$$

The current control is achieved by two PI controllers whose structures are the same as (2).

The dq-abc Transformation is describes as:

$$\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix}=\sqrt{\frac{2}{3}}\begin{bmatrix} \cos\theta & -\sin\theta \\ \cos(\theta-\frac{2}{3}\pi) & -\sin(\theta-\frac{2}{3}\pi) \\ \cos(\theta+\frac{2}{3}\pi) & -\sin(\theta+\frac{2}{3}\pi) \end{bmatrix}\begin{bmatrix} v_d \\ v_q \end{bmatrix} \tag{4}$$

Where θ is the rotor electrical angular position detected by the encoder. The sin and cos table is stored in the data memory.

The abc-dq Transformation is described as:

$$\begin{bmatrix} i_d \\ i_q \end{bmatrix}=\sqrt{\frac{2}{3}}\begin{bmatrix} \cos\theta & \cos(\theta-\frac{2}{3}\pi) & \cos(\theta+\frac{2}{3}\pi) \\ -\sin\theta & -\sin(\theta-\frac{2}{3}\pi) & -\sin(\theta+\frac{2}{3}\pi) \end{bmatrix}\begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \tag{5}$$

Where i_a and i_b are detected by Hall current sensors, and $i_c=-(i_a+i_b)$.

The on-line self-tuning control includes inertia estimator, speed PI parameter switch, and Iq saturation limit.

3. On-Line Self-Tuning Control

The on-line self-tuning control is carried out in three steps: the inertia variation detection, inertia calculation, and parameter switching.

3.1 Inertia Variation Detection

The mechanical equation of PMSM is described as:

$$J\frac{d\omega}{dt} = T_e - M_L \quad (6)$$

$$J = J_m + J_l \quad (7)$$

where J_m and J_l denote the inertia constants of the rotor and the load respectively; T_e and M_L denote the electromagnetic torque and the external load respectively; ω is the rotor mechanical speed.

The electromagnetic torque can be described as:

$$T_e = p[\psi_f i_q + (L_d - L_q) i_d i_q] \quad (8)$$

where L_d and L_q denote d-axis and q-axis inductances respectively. ψ_f denotes linkage flux of permanent magnet. P is pole pair number.

From (2), (3), and (8), one can get

$$T_e \approx p\psi_f i_q \quad (9)$$

So T_e is mainly determined by i_q.

When a speed command $\omega^*(n)$ is given, the speed controller output I_q^* will rise or decrease quickly. So in most drives, a saturation limit of current is set to protect drives and motors. If $\Delta\omega(n)$ is big enough, the absolute value i_q will reach the saturation limit quickly and remain this value in a definite period. During this period, if M_L does not change, $d\omega/dt$ will keep constant. The on-line self-tuning control is fulfilled in this period.

In this paper, the saturation limit value $I_{q\lim it} = 3I_n$, where I_n denotes rated current.

Then during current saturation, the torque of the PMSM is:

$$T_e = 3T_n \quad (10)$$

where T_n denotes rated torque.

In most cases, $|M_L| < |T_n|$ (11)

In a very short period t1 to t2 during current saturation, i_q and M_L are assumed to keep constant. According to (6), (9), and (10), one gets:

$$\omega(t_2) - \omega(t_1) = \frac{t_2 - t_1}{J_c}(3T_n - M_L) \quad (12)$$

in which J_c is the current mechanical inertia.

Let $a^* = \frac{t_2 - t_1}{J^*} \cdot (3T_n)$ (13)

where J^* is the mechanical inertia measured at the last time.

The self-tuning controller calculates $\omega(t_2) - \omega(t_1)$ periodically on-line.

Assume that the current inertia keeps the same as that of the last time, that is to say:

$$J_c = J^* \quad (14)$$

From (12), (13), and (14) one can get

$$\left| |\omega(t_2) - \omega(t_1)| - a^* \right| = \left| M_L \cdot \frac{t_2 - t_1}{J^*} \right| \quad (15)$$

Let $e = \left| T_n \cdot \frac{t_2 - t_1}{J^*} \right|$. From (11), (15) one can get

$$\left| |\omega(t_2) - \omega(t_1)| - a^* \right| < e \quad (16)$$

So if (15) is not satisfied according to on-line measurement, the assumption (14) is not satisfied. In this case, it can be said that the mechanical inertia has changed: $J_c \neq J^*$.

Then the self-tuning controller processes the next step: inertia calculation.

If the current remains $3T_n$ too long, the drive should pull the current down to T_n to protect the drive and PMSM.

3.2 Inertia Calculation

In this step, at first, the current limit is set as $I_{q\lim it} = 2I_n$, then the electromagnetical torque of the

PMSM will be $T_e = 2T_n$. In the next time period t3 to t4, one can get:

$$\omega(t_4) - \omega(t_3) = \frac{t_4 - t_3}{J_n}(2T - M_L) \tag{17}$$

Let $t_4 - t_3 = t_2 - t_1$. Assume that M_L keeps constant during these two periods. Subtracting (12) from (17) yields

$$J_c = T_n \cdot \frac{t_2 - t_1}{[\omega(t_4) - \omega(t_3)] - [\omega(t_2) - \omega(t_1)]} \tag{18}$$

3.3 Parameter Switching

Let $b = [\omega(t_4) - \omega(t_3)] - [\omega(t_2) - \omega(t_1)]$.

To simplify the control algorithm, a PI controller parameter selection table is built based on several experiments and calculation. In this table, DSP looks up the current J and the appropriate K_p and K_i according to the current b calculated by self-tuning controller. Normally, if b decreases, the calculated J will become bigger and K_p, K_i should be set with larger values. Then the DSP switches PI controller parameter to appropriate values.

4. Simulations and Experiments

4.1 Parameters of Experimental System

To show the feasibility of the proposed control scheme, the computer simulations and experiments are carried out for the proposed control scheme. The parameters are listed in Table 1.

Table.1 PMSM Parameters and Control System Constants

Machine Parameters: Nominal speed=2000rpm Nominal torque=10N.m Nominal current=20A $J_m = 0.0021 Kg \cdot m^2$ $P = 4$, $\Psi_f = 0.078\,Wb$
Control System Constants: $M_l = 5NM$ Speed command $\omega_1^* = 0\,rpm$ Speed command $\omega_2^* = 500\,rpm$

4.2 Simulations

In Fig. 2 and Fig. 3, the mechanical inertia J is increased to 10 times at t=0.75s.

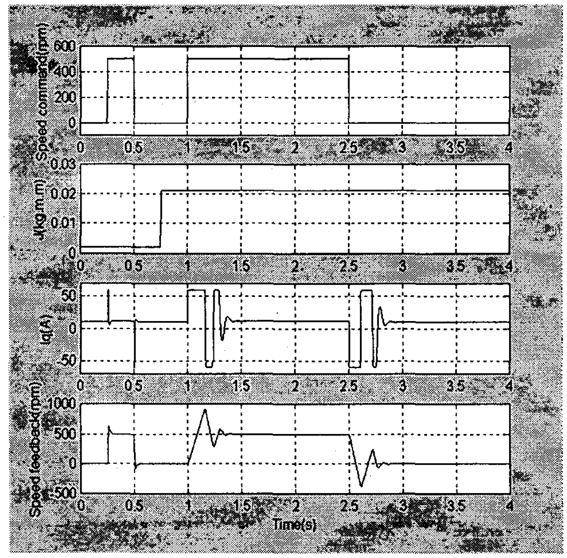

Fig.2 Simulation results of the drive without self-tuning PI controller

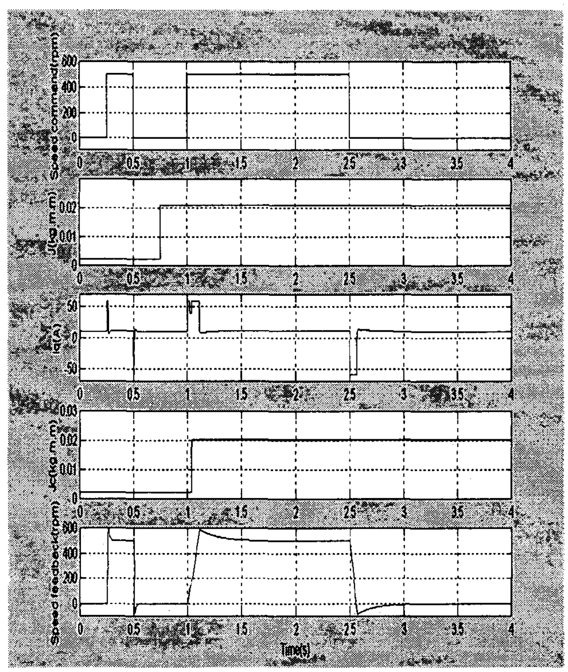

Fig.3 Simulation results of the drive with self-tuning PI controller

Fig.2 shows that the current I_q and speed ω responses of the servo drive without self-tuning PI controller. Obviously, the dynamic performance of the drive becomes worse after t=0.75s.

Fig. 3 shows the speed and current responses and inertia calculation of the drive with self-tuning PI controller. At t=1.04s, the controller calculates out the current mechanical inertia J_c and then switches K_p and K_i to corresponding appropriate values. As a result the dynamic performances are improved greatly.

4.3 Experiments

The control method is implemented with TMS320F240 which provides a single chip solution for PMSM servo drive by integrating on chip not only a high computational power but also all the peripherals necessary for electric motor control. The PWM Generator is realized by the event manager of the TMS320F240.

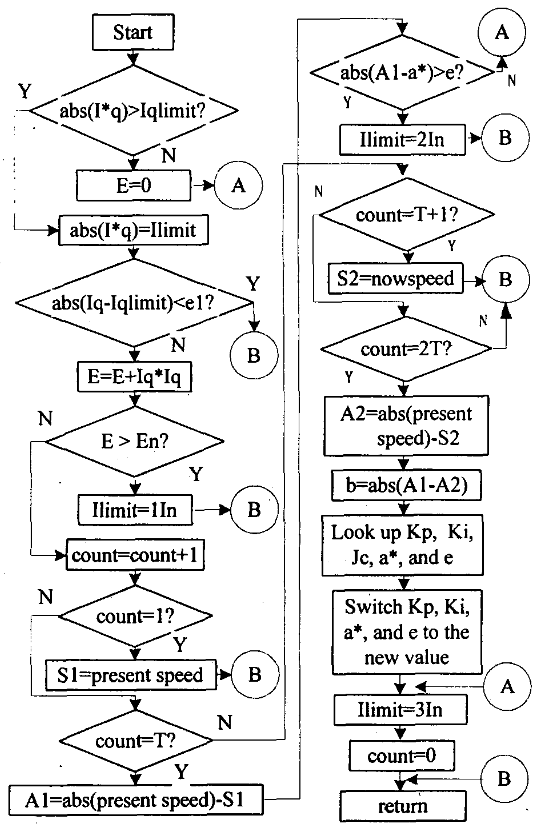

Fig. 4 Flowchart for Self-Tuning PI Controller

The flowchart of the interrupt program of the self-tuning PI controller is shown in Fig. 4.

The interrupt program is executed every 0.1ms. In the program, the detection period is 30ms (T=300).

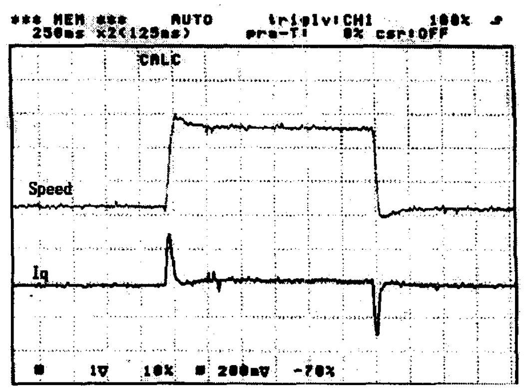

Fig. 5 The speed and Iq responses of the drive for unloaded PMSM. (Speed: 200rpm/div; Iq: 30A/div; time: 0.125s/div.)

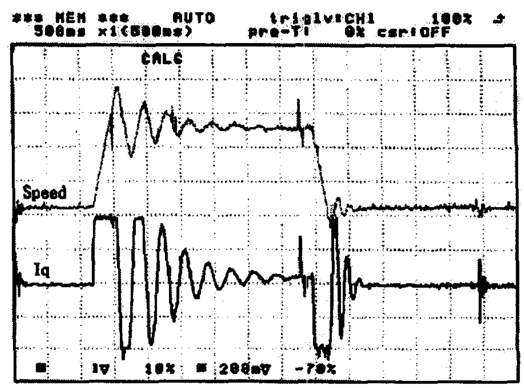

Fig. 6 The speed and Iq responses of the drive without self-tuning PI controller after the inertia J is increased 10 times. (Speed: 200rpm/div; Iq: 30A/div; time: 0.5s/div.)

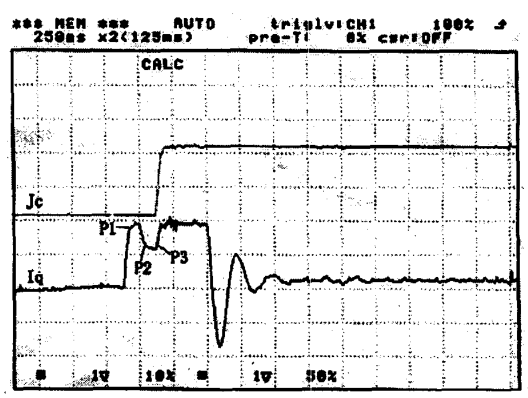

Fig. 7 The inertia calculation result Jc. (Iq: 30A/div; time: 0.125s/div.)

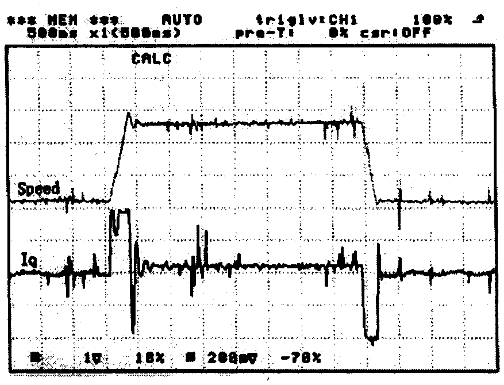

Fig. 8 The speed and Iq responses of the drive with self-tuning PI controller after the inertia J is increased 10 times. (Speed: 200rpm/div; Iq: 30A/div; time: 0.5s/div.)

Fig. 5-Fig. 8 show the results of the experiments.

In Fig. 5, the gains of the speed PI controller are initially set appropriate for the unloaded motor.

Comparing the overshoots of Fig. 5 and Fig. 6, it can be seen that the dynamic performance of the drive without self-tuning controller becomes worse when the inertia J is increased 10 times.

Fig. 7 shows the inertia calculation result of the self-tuning controller. At point P1, the controller starts the first step: inertia change detection. At Point P2, the controller finds that the inertia J has been changed and let $I_{qlimit} = 2I_n$. At point P3, the controller calculates out the mechanical Inertia Jc and switches the Kp and Ki to appropriate values.

Comparing Fig. 6 and Fig. 8, it can be seen that the overshoot and vibration of Fig. 8 are much smaller than those of Fig. 6.

5. Conclusions

The paper designs a DSP-based fully digital AC drive for servo application, in which speed control and current control are implemented with software. Taking advantage of the fast calculation ability of DSP, this paper also proposes an on-line self-tuning PI controller, which can detect the mechanical inertia variation periodically.

If the load inertia variation exceeds a certain value, the controller will calculate the mechanical inertia and adjust the gains of the speed loop PI controller based on the estimated inertia. Some simulation and experimental results have demonstrated that the motor drive with the proposed method can achieve a good dynamic and robust performance.

References

[1] R. Balasubramanian, K. H. Wong, "A Microcomputer-Based Self-Tuning IP Controller for DC Machines", IEEE Transaction On Industry Application, VOL. IA-22, NO6, 1986, pp989-999.

[2] Guo Qingding, Wang Limei, Luo Ruifu, "Completely Digital PMSM Servo System Based on New Self-Tuning PID Algorithm and DSP", IEEE International Conference on Industrial Technology, 1996, pp 71-75.

[3] A. Chandra, L. A. Dessaint, M. Saad, K. Al-Haddad, "Implementation of Self-Tuning Algorithms for Reference Tracking of a DC Drive Using a DSP Chip", IEEE Transaction On Industry Application, VOL. 41, NO1, 1994, pp104-108.

[4] Kouetsu Fujita and Katsumasa Sado, "Instantaneous Speed Detection With Parameter Identification for AC Servo Systems", Conference Record of the 1990 IEEE Industry Application Society Annual Meeting, Vol.1, pp 632-638.

[5] Haruo Naitoh, Susumu Tadakuma. "Microprocessor-Based Adjustable Speed DC Motor Drive Using Model Reference Adaptive Control", IEEE Trans. Ind. App Vol.IA-23. NO.2. March/April 1987.

[6] S. K. Chung, J. H. Lee, J. S. Ko, M. J. Youn, "Robust speed control of brushless direct-drive motor using integral variable structure control", IEE Proc-Electr. Power Appl. Vol. 142. No. 6. Nov. 1995, pp361-370.

[7] Francesco Parasiliti, Marco Tursini, Daqing Zhang, "On-line Self-tune of PI Controller for High Performance PMSM Drives", IEEE, IAS, Vol 3, 1996, pp1619-1625.

[8] Qingquan Li, "Theory, design and application of adaptive control systems", Science Press, Beijing, 1990