Abstract - The new generation of Field Programmable Gate Arrays (FPGAs) technology enables to be embedded a processor to construct an SoPC (System-on-a-Programmable-Chip) developing environment. Therefore, this study presents a servo control IC for X-Y table using this SoPC technology. In this proposed servo control IC, there are two modules. One module performs the functions of the motion trajectory and two-axis position controller. The other module performs the functions of current vector control of motor drives for two PMSMs (Permanent Magnet Synchronous Motors). The former is implemented by software using Nios II embedded processor due to the need of the complicated control algorithm and low sampling frequency control (motion trajectory & position control: less than 1kHz). The latter is implemented by hardware using PLD (programmable logic device) in FPGA due to the need of high sampling frequency control (current loop: 16kHz, PWM circuit: 4–8MHz) but simple computation. Furthermore, to improve the control accuracy of the table positioning, a full closed-loop control, which the signal of linear encoder is feed-backed for position control, is considered. As this result, the use of SoPC technology can make the servo controller of X-Y table more compact, high performance and cost down.

I. INTRODUCTION

Development of a compact and high performance servo controller system for precision X-Y table, CNC machine etc. is a popular research work in literature [1-2]. In position control of X-Y table, there are two methods to be considered, that one is semi closed-loop control and the other is full closed-loop control. The full closed-loop control with feed-backed by a linear encoder as the table position signal had been proven a better position control method then the semi closed-loop control, that a rotary encoder attached to ac motor is feed-backed as the position signal.

For the progress of VLSI technology, the FPGA have been widely investigated due to their programmable hard-wired feature, fast time-to-market, shorter design cycle, embedding processor, low power consumption and higher density for implementing digital system [3-4]. FPGA provides a compromise between the special-purpose ASIC (application specified integrated circuit) hardware and general-purpose processors [5]. Therefore, using an FPGA to develop a compact, low-cost and high performance servo system for precision machine become an important issue. But in many researches, the FPGA is only used to realize the hardware part of the overall servo control system [6].

Nowadays, SoPC and IP (Intellectual Property) designs can be developed and downloaded into FPGA with an embedded processor to construct an SoPC environment [7-9]. Using SoPC technology, the software and hardware co-design are reprogrammable and can be parallel processing in FPGA to increase the system performance and flexibility. Due to the advantage, a full close-loop control IC for X-Y table based on the SoPC technology is developed and shown in Fig.1. In Fig.1, the scheme of current vector control of two PMSMs can be realized by hardware in programmable logic devices (PLD), and the motion trajectory and position control algorithm for X-Y table can be realized by software using Nios II processor. Therefore, all of the function needed to construct a full closed-loop control for X-Y table can be integrated and realized in an FPGA chip. In this paper, FPGA chip adopts Altera Stratix II EP2S60 [10-11], which has 24,176 ALMs (equivalent 60,440LEs), maximum 718 user I/O pins, 36 DSP blocks, total 2,544,192 RAM bits, and a Nios II embedded processor which has a 32-bit configurable CPU core, 16 M byte Flash memory, 1 M byte SRAM and 16 M byte SDRAM, is used. Therefore, this FPGA chip is very suitable to develop a servo controller system for X-Y table.

II. SYSTEM DESCRIPTION AND DESIGN

The architecture of the proposed FPGA-based servo control IC for X-Y table is shown in Fig. 1, in which the motion trajectory, the position and speed control are implemented by software using Nios II embedded processor and the current vector of motor drives for two PMSMs are implemented by hardware in FPGA chip. The modeling of PMSM, fuzzy control algorithm and design of proposed servo control IC are introduced as follows:

A. Mathematical modeling of PMSM

The typical mathematical model of a PMSM is described, in two-axis d-q synchronous rotating reference frame, as follows

\[
\frac{di_d}{dt} = -\frac{R_d}{L_d}i_d + \omega_e \frac{L_q}{L_d}i_q + \frac{1}{L_d}v_d \\
\frac{di_q}{dt} = -\omega_e \frac{L_d}{L_q}i_d - \frac{R_q}{L_q}i_q - \omega_e \frac{L_d}{L_q}i_q + \frac{1}{L_q}v_q
\]  

(1) (2)

where \(v_d, v_q\) are the d and q axis voltages; \(i_d, i_q\) are the d and q axis currents, \(R_e\) is the phase winding resistance; \(L_d, L_q\) are the d and q axis inductance; \(\omega_e\) is the rotating speed of magnet flux; \(\lambda_f\) is the permanent magnet flux linkage.

Mathematical modeling of PMSM
The current loop control of each PMSM drive in Fig. 1 is based on a vector control approach and its detail description is shown in Fig. 2. If the \( i_d \) is controlled to 0 in Fig. 2, the PMSM will be decoupled, and control a PMSM like a DC motor. After simplification, the modeling of a PMSM can be written as the following equation,

\[
T_e = \frac{3P}{4} \lambda_s i_s K_f i_r
\]

with

\[
K_i = \frac{3P}{4} \lambda_s
\]

Considering the mechanical load with linear table, the overall dynamic equation of linear table system is obtained by

\[
T_e - T_L = J_m \frac{d^2 s_p}{dt^2} + B_m \frac{2\pi}{r} ds_p
\]

where \( T_e \) is the motor torque, \( K_i \) is force constant, \( J_m \) is the inertial value, \( B_m \) is damping ratio, \( T_L \) is the external torque, \( s_p \) represents the displacement of x-axis or y-axis table and \( r \) is the lead of the screw.

### B. Coordinate transformation

The coordination transformation of the PMSM in Fig. 2 can be described in synchronous rotating reference frame. The coordination system used in rotating motor includes stationary a-b-c frame, stationary \( \alpha-\beta \) frame and synchronously rotating d-q frame. These transform equations are described as follows, and the \( \bar{\gamma} \) represents a space vector refer to current, voltage or flux.

- **Clarke**: stationary a-b-c frame to stationary \( \alpha-\beta \) frame.

\[
\begin{bmatrix}
\alpha \\
\beta 
\end{bmatrix} = \begin{bmatrix}
2 & -1 \\
3 & 3 \\
0 & \sqrt{3}
\end{bmatrix} \begin{bmatrix}
\gamma_a \\
\gamma_b \\
\gamma_c
\end{bmatrix}
\]

- Modified **Clarke**\(^{-1}\): stationary \( \alpha-\beta \) frame to stationary a-b-c frame.

\[
\begin{bmatrix}
\gamma_a \\
\gamma_b \\
\gamma_c
\end{bmatrix} = \begin{bmatrix}
\frac{1}{2} & -\frac{\sqrt{3}}{2} \\
\frac{\sqrt{3}}{2} & \frac{1}{2} \\
0 & 0
\end{bmatrix} \begin{bmatrix}
\alpha \\
\beta
\end{bmatrix}
\]

- **Park**: stationary \( \alpha-\beta \) frame to rotating d-q frame.

\[
\begin{bmatrix}
f_\alpha \\
f_\beta
\end{bmatrix} = \begin{bmatrix}
\cos \theta & \sin \theta \\
-\sin \theta & \cos \theta
\end{bmatrix} \begin{bmatrix}
f_\alpha \\
f_\beta
\end{bmatrix}
\]

- Modified **Park**\(^{-1}\): rotating d-q frame to stationary \( \alpha-\beta \) frame.

\[
\begin{bmatrix}
f_\alpha \\
f_\beta
\end{bmatrix} = \begin{bmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{bmatrix} \begin{bmatrix}
f_\alpha \\
f_\beta
\end{bmatrix}
\]

### C. Fuzzy controller for position control loop

The internal block diagram of motion controller for X-Y table in Fig. 1 is shown in Fig. 3. The motion controller, which is implemented by software in Nios II embedded processor, performs the function of the motion trajectory, two-axis’ position and speed controller, etc.

Current vector control for each axis of X-Y table

Which is implemented by hardware (PLD) in FPGA

![Fig. 2. Block diagram of current vector control at each axis of X-Y table](Image 105x634 to 119x657)
FPGA-based servo control IC for X-Y Table

Fig.3. Block diagram of position control and trajectory computation in Nios II Processor

Each position controllers in Fig. 3 adopts fuzzy controller, which includes fuzzification, fuzzy rules, inference mechanism and defuzzification. At first, the position error and its error change, $\epsilon$, $\Delta \epsilon$ are defined by

$$\epsilon(n) = x_p^*(n) - x_p(n) \quad (10)$$

$$\Delta \epsilon(n) = \epsilon(n) - \epsilon(n-1) \quad (11)$$

and $K_0$, $K_e$, $K_{\Delta \epsilon}$ and $K_u$ are the input and output variable of fuzzy controller, respectively. Then, the design procedure of the fuzzy controller is as follows:

- Define the linguist value as $\{A_1, A_2, E\}$, they are symmetrical triangular membership function:

$$\xi_{\epsilon, \Delta \epsilon}(x_1, \pi^*_m, w^*_m) = \begin{cases} 0 & x_1 \leq \pi_m^* - w_m^*/2 \\ \frac{x_1 - \pi_m^* + w_m^*/2}{w_m^*/2} & \pi_m^* - \frac{w_m^*/2}{2} < x_1 < \pi_m^* \\ \frac{w_m^*/2}{2} & \pi_m^* < x_1 \leq \pi_m^* + w_m^*/2 \\ 0 & x_1 \geq \pi_m^* + w_m^*/2 \end{cases} \quad (12)$$

where $x_1$ is input value, $\xi_{\epsilon, \Delta \epsilon}(\bullet)$ is output value, $\pi_m^*$ and $w_m^*$ are mean value and width of the triangular function, respectively.

- Derive M fuzzy control rules from the dynamic response characteristics [12], as follows

IF $\epsilon$ is $A_m^\epsilon$ and $\Delta \epsilon$ is $A_m^{\Delta \epsilon}$ THEN $u_m$ is $E_m$, $m=1..M \quad (13)$

- Construct the fuzzy system with $u_f(\Sigma \theta)$ from those M rules using the singleton fuzzifier, product-inference rule, and central average defuzzifier method. Therefore, the fuzzy control rule in (13) can be replaced by the following expression:

$$u_f(\Sigma \theta) = \frac{\sum_{m=1}^{M} c_m \prod_{i=1}^{N} \xi_m(x_i, \pi_m^*, w_m^*)}{\sum_{m=1}^{M} \prod_{i=1}^{N} \xi_m(x_i, \pi_m^*, w_m^*)} - \frac{\sum_{m=1}^{M} c_m \mu_m}{\sum_{m=1}^{M} \mu_m} \quad (14)$$

where $c_1, c_2, .., c_m$ denote the center of the output membership function.

D. Point-to-point motion control

In Fig.1, the point-to-point motion control algorithm is used as the motion trajectory control for X-Y table. In this scheme, a trapezoidal velocity profile by constant acceleration and deceleration is considered, and the overall displacement $\Delta s = [\Delta x_p, \Delta y_p]$ (mm), the maximum velocity $[W_1, W_2]$ (mm/s), the acceleration and deceleration period $T_{acc}$, and the sampling interval $t_d$ are the input parameters of the scheme. Therefore, based on the velocity profile, the command of instantaneous position, $s^* = [x_p^*, y_p^*]$ at each sampling interval in Fig.3 can be determined by the following procedure.

Step 1: Computation of the overall running time. First, compute the running time without considering acceleration/deceleration:

$$T_1 = \max (\Delta x_p / W_1, \Delta y_p / W_2) \quad (15)$$

This $T_1$ must exceed acceleration time $T_{acc}$. Next, the acceleration/deceleration design is considered, and the overall running time is

$$T_p = \max (T_1, T_{acc}) + T_{acc} \quad (16)$$

Step 2: Adjustment of the overall running time to meet the condition of multiple of the sampling interval.

$$N' = [T_p / t_d] \quad (17)$$

where $N$ is the interpolation number and $[ ]$ represents the Gauss function. Therefore,

$$T_{acc} = N' * t_d \quad (18)$$

Step 3: Modification of the maximum velocity

$$W' = [W_1', W_2'] = [\Delta x_p, \Delta y_p] / (T_p - T_{acc}) \quad (19)$$

Step 4: Calculation of the acceleration/deceleration

$$A = W' / T_{acc} \quad (20)$$

Step 5: Calculation of the position command, namely $s^*_p = [x^*_p, y^*_p]$, at the mid-position

(a) Acceleration region:

$$s^*_p = s_{r0} + \frac{1}{2} * A * t^2 \quad (21)$$

where $t = n * t_d$, $0 < n \leq N'$, and $s_{r0}$ is the initial position.

(b) Constant speed region:

$$s^*_p = s_{r1} + W' * t \quad (22)$$

where $t = n * t_d$, $0 < n \leq N1$ and, $s_{r1}$ is the final position in the acceleration region and $N1 = N-2 * N'$.

(c) Deceleration region:

$$s^*_p = s_{r2} + (W' * t - \frac{1}{2} * A * t^2) \quad (23)$$

where $t = n * t_d$, $0 < n \leq N'$ and $s_{r2}$ is the final position in the constant speed region.
E. Design of FPGA-based servo control IC for X-Y Table

The internal architecture of the proposed FPGA-based servo control IC for X-Y table is shown in Fig. 4. The FPGA uses Stratix II EP2S60, which has 24,176 ALMs (equivalent 60,440LEs), maximum 718 user I/O pins, 36 DSP blocks, total 2,544,192 RAM bits, and a Nios II embedded processor which has a 32-bit configurable CPU core, 16 M byte Flash memory, 1 M byte SRAM and 16 M byte SDRAM. A custom software development kit (SDK) consists of a compiled library of software routines for the SoPC design, a Make-file for rebuilding the library, and C header files containing structures for each peripheral. In this control IC, it has two modules. One module performs the functions of the motion trajectory and two-axis position servo control. The other module performs the functions of vector control of two PMSMs. Module 1 in Fig.4 is Nios II embedded processor used to implement the complicated control algorithm by software. The flow charts of interrupt service routine (ISR) for motion control are plotted and shown in Fig.5. The module 2 in Fig.4 includes two set of circuits of service routine (ISR) for motion control are plotted and shown for each peripheral. In this control IC, it has two modules. One module performs the functions of the motion trajectory and two-axis position servo control. The other module performs the functions of vector control of two PMSMs. Module 1 in Fig.4 is Nios II embedded processor used to implement the complicated control algorithm by software. The flow charts of interrupt service routine (ISR) for motion control are plotted and shown in Fig.5. The module 2 in Fig.4 includes two set of circuits of current vector control which is implemented by hardware in FPGA due to the need of high-speed processing but simple computation. Each circuits of vector control includes two PI controllers, coordinate transformation of Clarke, Park, inverse Park, modified inverse Clarke and circuits of SVPWM, QEP and ADC conversion control. The implementation of PI controller which includes 3 adders, 2 multipliers, 2 D-type flip-flops and 3 max value limiters. Figures 7 to 8 are the circuits of modified Clarke\(^1\) and Park\(^1\) transformation, respectively. The block diagram of SVPWM circuit is shown in Fig. 9. SVPWM circuit is designed with 12 kHz frequency and 1\(\mu\)s dead-band. The overall circuits included a Nios II embedded processor IP (10%) and two current vector control circuits (15%) in Fig. 4, use 25% utility of Stratix II EP2S60.

---

**Fig. 4.** The block diagram of internal circuit of FPGA-based servo control IC

**Fig. 5.** Flow chart of main and ISR program in Nios II processor

**Fig. 6.** Digital circuit of PI controller

**Fig. 7.** The designed circuit of modified Clarke\(^1\) formulation

**Fig. 8.** The designed circuit of Park\(^1\) formulation

---

2916
The overall experimental system is depicted in Fig. 1. This system includes an FPGA experimental board, two sets of voltage source IGBT inverter and an X-Y table which is driven by two PMSMs and two ball-screws. The power, rating, voltage, current and rating speed of each PMSM are 200W, 92V, 1.6A and 3000rpm, respectively. A 2500 ppr rotary encoder attached to PMSM is used to the sensor of motor’s electrical angle. Two linear encoders with 5㎛ resolution are mounted on the x- and y-table as position sensor, respectively. Each ball-screw has 5mm lead. The inverter has 6 sets of IGBT type power transistors. The collector-emitter voltage of the IGBT is rating 600V, the gate-emitter voltage is rating ±12V, and the collector current in DC is rating 25A and in short time (1ms) is 50A. The photo-IC, Toshiba TLP250, is used for gate driving circuit of IGBT. Input signals of the inverter are PWM signals from FPGA chip. The FPGA-Altera Stratix II EP2S60 in Fig.1 is used to develop a full digital motion controller for X-Y table, which the motion trajectory and two-axis position servo control are implemented by software using Nios II processor, and the two axis current vector controller are implemental by hardware using programmable logic devices in FPGA. Therefore, the servo control IC can integrate and realize a fully digital motion control, are designed by hardware in FPGA. Therefore, the servo control IC can integrate and realize a fully digital motion controller of X-Y table in an FPGA chip. Experimental results successfully validated the step response, point-to-point path tracking and circular trajectory tracking, revealing that the proposed software/hardware co-design method with parallel processing performance well in the servo system of X-Y table.

ACKNOWLEDGEMENT

This work was supported by National Science Council of the R.O.C. under grant no. NSC 95-2221-E-218-071.
Fig. 12. Position and its velocity profile tracking response of X-axis table

Fig. 13. Position and its velocity profile tracking response of Y-axis table

Fig. 14. Circular motion tracking of X-Y table

REFERENCES