The Design of an Industrial Robot Controller using FPGA

CHEN Xuechang *

Institute of Applied Electronics, Chongqing College of Electronic Engineering; Chongqing 401331, China
pcboy2004@126.com

***Abstract* — A motion controller is the core of industrial robots, and its strengths and weaknesses directly affect the accuracy and performance of their motion. Current industrial robot controllers mostly adopt the universal motion controller. Their real-time performance is not high, the motion control algorithm is very simple and does not offer opportunities for development. The interface is simple, unable to meet the requirements of industrial robots with complex motion control systems. In order to meet the real-time requirement of robot motion, a controller using FPGA is studied, and the system structure features, hardware circuit design and the main module of the controller are analyzed. The results show the controller can achieve reliable motion control of the robot.**

Keywords - industrial robot; motion controller; FPGA; hardware circuit

I. INTRODUCTION

In recent years, the research on the openness and real-time of industrial robot control system is a hot research topic in the field of industrial robot. In order to adapt to the development of science and technology, the control system of the robot must be open in both hardware and software structure, but also able to maintain a high real-time response. To this end, the research institutions have done a lot of research in this direction. Some scholars have designed a motion control card based on the ARM micro-controller + special motion control chip MCX314 [1], whose smallest interpolation output unit is 0.001mm and the processing precision can reach 0.01mm. This motion control card uses a special motion control chip, so there is no need to consider the specific interpolation, but it cannot introduce complex control or interpolation algorithm, poor in scalability.

At the same time, some scholars have designed a motion control card based on ARM+FPGA, whose fine interpolation and signal detection of motion control and human machine interface are realized by FPGA, while the rough interpolation, servo algorithm, data processing and other auxiliary control of motion control is implemented by ARM [2]. The motion control card submits the interpolation work to ARM and FPGA to achieve, whose real-time is guaranteed, but inconvenient to introduce a complex interpolation algorithm. Some scholars have designed a motion control system based on PC + motion control card. The PC machine of this set of motion control system completes the data rough interpolation; the motion control card is composed of DPS+FPGA, in which DSP completes the data fine interpolation, and FPGA completes the main control logic [3]. The positioning accuracy of the motion control system is up to 1μm and the stability is good. After continuous processing for 48h on the machine, the machine runs stably, and all the processing data are correct, but the real-time is not ideal.

Obviously, current motion control system mainly adopts two forms: one is the PC+ motion control card. The motion control system of PC+ motion control card is not very ideal in real time, speed, processing capacity, volume, man-machine interface and complex network interface. It is difficult to achieve miniaturization and complex control and analysis; the other one is an embedded motion control system. The micro-controller based on kernel is very perfect in supervisory control, man-machine interface and so on [4]. Compared with the former one, the embedded motion control system is facing specific applications, with good mobility, low power consumption, high reliability, high cost-effective, strong real-time, and support for multiple-task application [5]. It can be said that the embedded structure is the future development direction of the robot control system and other intelligent machine control systems.

Based on this, this paper puts forward the industrial robot motion controller system based on FPGA. The hardware and software can be cut, and it can meet the strict requirements of the application system functions of reliability, cost, volume and power consumption and so on indexes.

II. STRUCTURE OF ROBOT MOTION CONTROLLER

Industrial robot, in the actual work, often needs to run a large number of real-time and non-real-time tasks, so when designing the robot controller, real-time and multiple-task processing ability are the two key consideration factors. On the one hand, in order to meet the two requirements, the robot control system often needs to be designed with very high hardware and software configuration, and the volume of components will increase [6]; on the other hand, in order to meet the requirements of miniaturization of robot controller, components must be intelligent enough, and the volume should be as small as possible [7]. The ARM+DSP+FPGA architecture scheme can solve the above problems satisfactorily, and it is a better scheme to solve the robot control system with embedded processor as the core. The ARM processor has the advantages of small size, light weight, low cost, high reliability, rich peripheral interfaces, and low power consumption, widely applied in the field of control; the DSP processor has special hardware multiplier, widely used in pipeline operation, and suitable for large amount of computing system with robot interpolation as representation the flexibility of FPGA makes the hardware circuit reconstructing, so the controller design is more flexible and convenient [8]. ARM+DSP+FPGA architecture can effectively combine the strengths of the three processors, mutually complementing, which is the mainstream of robot controller design. For the ARM+DSP+FPGA robot motion controller, the function framework is shown in Figure 1.

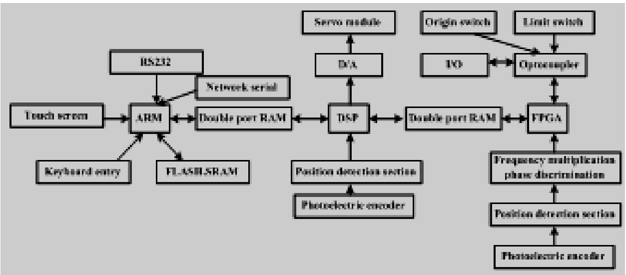

Figure 1. The algorithm flow chart

As shown in Figure 1, the ARM processor explains the motion instructions in the program, then the data is sent to the double-port RAM module. DSP reads these data for interpolation and obtains the theoretical position. The actual position of the motion axis calculated by FPGA is read from the feedback part, to calculate the error between the theory position and the actual position. Using PID controls the output position of the next step, and according to the error correction of D/ A output size, make real-time control of motion axes position and speed in the closed-loop robot control system.

III. Hardware Construction of Robot Motion Controller

1.ARM, DSP, and FPGA Interface Circuit

The normal communication among the ARM processor, the DSP processor and the FPGA is the basis of the robot motion control system. In the robot motion control system, with the increase of the amount of data acquisition and data processing tasks, the data transfer requirements are increasingly higher. Realizing data serial transmission by relying on the serial port of the ARM processor and DSP processor has been unable to meet the requirements, so it must be carried out with high-speed data transmission [9]. The double-port RAM can easily form high speed data transmission medium under a variety of working modes, which can solve the bottleneck problem caused by the low speed of data transmission. Whether it is the data sharing in parallel processing network, or high-speed data transmission of pipeline mode, double-port RAM can play an important role, so as to ensure the smooth flow of data access.

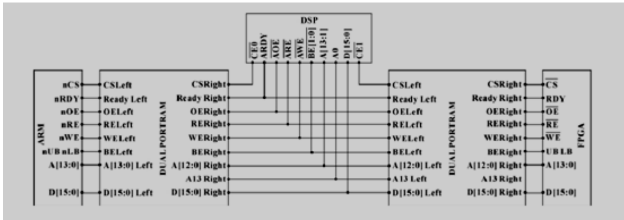

Figure 2. ARM, DSP, and FPGA interface circuit

As shown in Figure 2, use double-end RAM to make data exchange between the system ARM processor and DSP processor, DSP processor and FPGA. DSP achieves the communication between ARM and FPGA through two double-end. The main signal lines are as follows:

Address line. Double-end RAM address lines A0-A12 are connected with DSP A1-A13, and the double-end RAM A13 is connected with DSP A0 [10]. The double-end RAM chip has an address latch, and the address is latched in the write cycle.

(2) Data line. Double-end RAM data lines D0-D15 are respectively connected with DSP D0-D15. Double-end RAM data line, during the write cycle, an on-chip latch can be used to lock data [11]; during the reading out cycle, it is the data output, and it is in a high resistance state when the CE or OE is not valid.

(3) The chip select line. Double-end RAM chip select line is respectively connected with the CE0 and CE1 of DSP. Double-end RAM chip select line active low, data in the data buffer is output by the control chip during the read cycle, and when it is high active, the chip is in low power standby state.

(4) Allow OE. Double-end RAM allows AOE connection between OE and DSP. OE low level effective, the data in the data buffer is output during the read cycle.

2.Interface Circuit of FPGA and Absolute Encoder AC Servo Amplifier

FPGA is used to achieve the main external logic circuit. The control object of robot motion controller is Yaskawa Electric Company's SGDM-∑ II series servo unit and motor. In the FPGA chip, it mainly achieves various input and output signal acquisition of the absolute encoder AC servo amplifier. The interface circuit of each signal in FPGA are introduced:

(1) As shown in Figure 3, the servo function of ON signal, P motion signal P-CON, forbidden inverse forward signal N-OT and P-OT, alarm reset signal ALM-RST, inverse forward current limit signal N-CL and P-CL are issued by FPGA, and sent to the servo amplifier through the optocoupler [12];

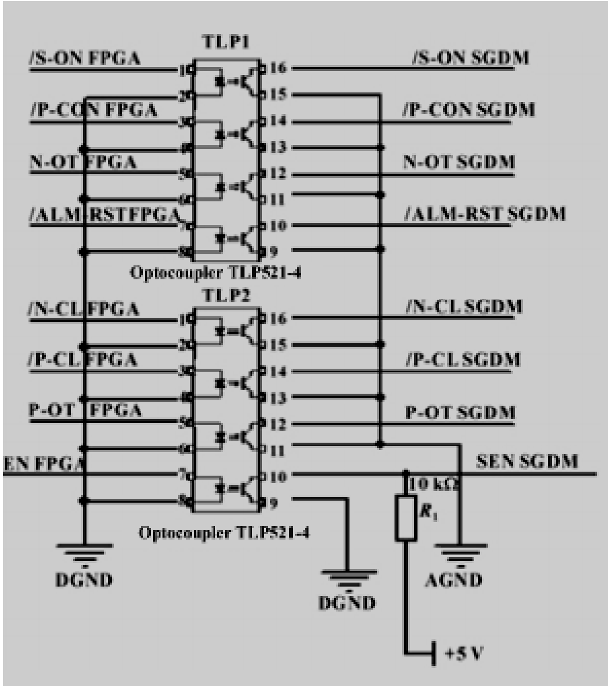

Figure 3. FPGA output signal

(2) As shown in Figure 4, the encoder output signals PAO, PBO, PCO, and PSO (the first three are output when the encoder is used as incremental encoder; PSO is the output when the encoder is used as the absolute encoder) are sent into the FPGA through the optocoupler isolation;

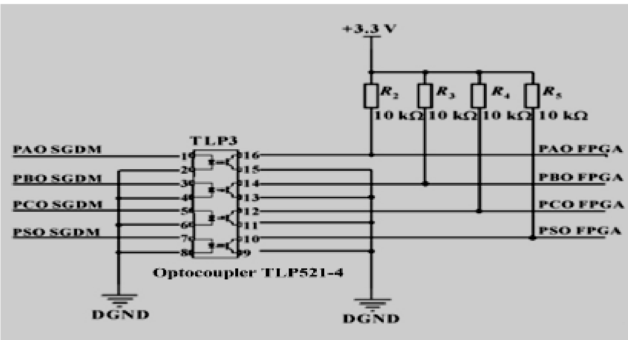

Figure 4. FPGA input signal

(3) The same speed detection signal V-CMP, the super-speed signal TGON, the servo ready signal S-RDY, and the servo alarm signal ALM are directly input to FPGA;

(4) The alarm code output signal ALO1, ALO2, ALO3 are directly sent into FPGA.

IV. Software Implementation of Robot Motion Controller

1、FPGA Control Software for AC Servo Amplifier

(1) The serial communication between FPGA and AC servo amplifier

Serial communication of robot motion controller is absolute encoding of SGDM-∑ II series servo unit and motor-provided 17 bits produced based on the Yaskawa Electric Company. The servo unit has a special signal interface which is given by the PSO signal with the absolute position of the motor in serial data. PS0 serial data outputs the 5-digit rotation and 7-digit absolute position in 1 circle [13]. The data output cycle is about 40ms, the baud rate is 9600b/s, and even parity check is used. Figure 5 is a schematic diagram of the PSO serial data specification.

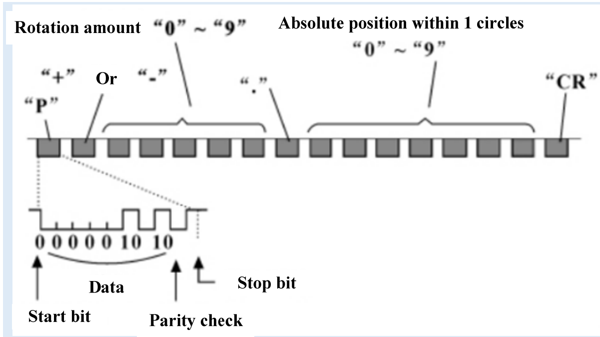

Figure 5. PSO serial data specifications sketch map

As shown in Figure 6, after the PSO signal is input, first of all, remove burrs (including high pulse and low pulse burrs); then detect PSO drop along, and when the drop along arrives, then begin to receive the first character; there will be the second drop along at the arrival of about 40ms [14], then begin to receive the second character; continue the operation until receiving the last character.

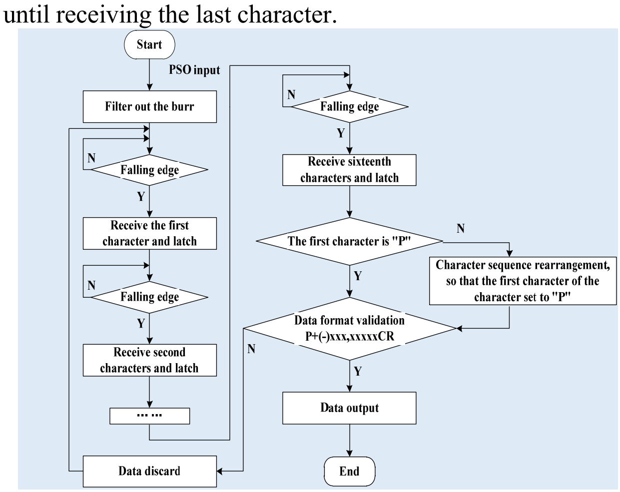

Figure 6. The flow chart of serial communication program

(2) The speed control of FPGA to the servo motor

Using the position control mode, FPGA inputs PULSE and SIGN signals to the servo amplifier. By changing the frequency of the PULSE signal, it can change the speed of the motor; by changing the level of the SIGN signal, it can change the motor's turn.

2、DSP D/A Output

D/A chip selects DAC7724, using direct binary code. The relationship between the ideal input digital and the output analog amount is shown in the following [15]:

VOUT=VREFL+(VREFH-VREFL)×N÷4096

Among them: N represents the digital input, VOUT refers to the analog output, VREFL suggests the low reference voltage, and VREFH is the high reference voltage.

3、DSP and Double-end RAM Data Read

When DSP reads the double-end RAM data, it is necessary to correctly configure the EMIF. The EMIF is configured to asynchronous memory mode, respectively distributing CE0 and CE1 used for data read of the double-end RAM on the left and right in Figure 2.

V. CONCLUSIONS

The development of robot motion control technology puts forward new requirements to the robot motion controller, which makes the new electronic components have more space to play the role. The seamless integration of ARM, DSP and FPGA makes the circuit design of the robot motion controller not only reduces the discrete components, but also enhances the stability, and has achieved good results.

In comparison, the real-time of PC robot control system usually needs to be ensured by the underlying hardware platform. While the robot motion controller embedded real-time operating system (RTOS) based on ARM+DSP+FPGA, has the advantages of real-time, low cost, miniaturization, customization and high reliability, fully guaranteeing the real-time requirements of the control system.

REFERENCES

[1] Ragaglia, M., Bascetta, L., Rocco, P., Zanchettin, A. M., Rocco, P., & Zanchettin, A. M. (2014). Towards Safe Human-Robot Intermotion: evaluating in real-time the severity of possible collisions in industrial scenarios. International Workshop on Human-Friendly Robotics.

[2] Guillo, M., & Dubourg, L. (2016). Impact & improvement of tool deviation in friction stir welding: weld quality & real-time compensation on an industrial robot ☆. Robotics and Computer-Integrated Manufacturing,39, 22-31.

[3] Han, D., Jang, Y. C., Oh, S. N., Chand, R., Lim, K. T., & Kim, K. I., et al. (2014). Mcu based real-time temperature control system for universal microfluidic pcr chip. Microsystem Technologies, 20(3), 471-476.

[4] Jiang, W., Wu, G., Wang, W., Zhang, J., Fan, F., & Ye, X. (2016). Robust motion control of live-line maintenance robot mechanical arm for high-voltage transmission line based on h theory.

[5] Zhao, J. D., & Yin, M. F. (2014). Development of automatic cnc grooving and bending machine based on mpc08sp motion control card. Applied Mechanics & Materials, 599-601, 1196-1200.

[6] Sujeeth, A. K., Brown, K. J., Lee, H., Rompf, T., Chafi, H., & Odersky, M., et al. (2014). Delite: a compiler architecture for performance-oriented embedded domain-specific languages. Acm Transmotions on Embedded Computing Systems, 13(4s), 392-398.

[7] Linsky, J. B., Michaelis, O., & Jaime, M. E. (2015). Method and apparatus for communicating control information by a wearable device to control mobile and consumer electronic devices. US, US9000887.

[8] Chowdhary, M., Kumar, A., Sharma, M., Paul, K., Jain, M., & Narula, G. (2015). Methods and applications for motion mode detection for personal navigation systems. US, US9163946.

[9] Rexrode, T., & Hilchey, M. (2014). Real-Time System Processing Capability Determination and Workflow Adaptation for Target Regions in a Customer Information Control System. US20140129284.

[10] Abdelmoula, C., Masmoudi, M., & Chaari, F. (2014). Real time algorithm implemented in altera's fpga for a newly designed mobile robot. Multidiscipline Modeling in Materials and Structures, 10(1), 75-93.