Applied Mechanics and Materials Vols. 271-272 (2013) pp 1536-1540 *Online: 2012-12-27*
© (2013) Trans Tech Publications, Switzerland
doi:10.4028/www.scientific.net/AMM.271-272.1536

A Chess-Playing Robot Based on SERCOS

Jilin He[1,a], Zhixian Chen[2,b]

[1]Intelligent Machinery Institute, Central South University, Changsha Hunan PRC, China,410083

[2]Intelligent Machinery Institute, Central South University, Changsha Hunan PRC, China,410083

[a]hejilin751021@yahoo.com.cn, [b]chenzhixian620@163.com

Keywords: Chess-playing robot SERCOS Real-time control

Abstract: This paper presents a chess-playing robot and its control system based on SERCOS. Thanks to Windows NT operating system and RTX real-time extension, the control system is excellent in real-time property. The whole system, supported by a standard IPC hardware platform and a modularized architecture of system software, is open-ended and expansible. In addition, the chess-playing software makes a contribution to the excellent interaction between players and the chess-playing robot.

1 Introduction

With the development of robot control technology, it is the trend to design “modular systems – open controllers simplifying the manufacturing process in the robot industry” [1]. The controller designed with open architecture could be used and transplanted in general. Most of them are developed based on PC platform, Programmable Multi-Axis Controller (PMAC), or other motion control device provided by the third party [2]. The important problem of controlling robot in industry application is in contact with environment [3].

In order to make the chess-playing robot a really excellent integration of robot and Chinese traditional chess-playing art, all kinds of high-tech were applied, and an authentic chessboard and chessman were employed. Our hardware design endowed the chess-playing robot with human characteristics and the ability to interact with people in an audio-visual way. The development of software enabled the robot to play chess with people intelligently. The information extraction, target identification and real-time motion control become the key points to realize high-level interaction between the robot and the player and the self-learning ability of the robot.

2 SERCOS Protocol

SERCOS (serial real time communication system) is a field bus interface and data communication protocol used in digital servo system and drive system [4]. In 1990s, Germany presented and began to establish SERCOS [5]. It is the only international standard for such communication. SERCOS simplified the control of motors and offered velocity, position and torque modules for motor control. SERCOS interface is made up of one master station and several slave stations. They are connected by fiber optic and constitute a loop net. Comparing to the other industry field bus, the SERCOS has an efficient channel for transferring synchronous, real-time data and valuable data, and high-quality sampling and timing regulator.

The application of DSP isolates the hardware real-time requirement of SERCOS to CPU and makes it possible to adopt different operating system. CPU can directly call SERCOS interface provided by DSP without SERCOS layer and send commands. With the capability of SERCOS and DSP, the robot controller with the performance of multi-axis motion simultaneously and full of open-endedness and modularization was designed.

3 Information and Command Transferring

Fig. 2 is the flow chart of the chess-playing robot.

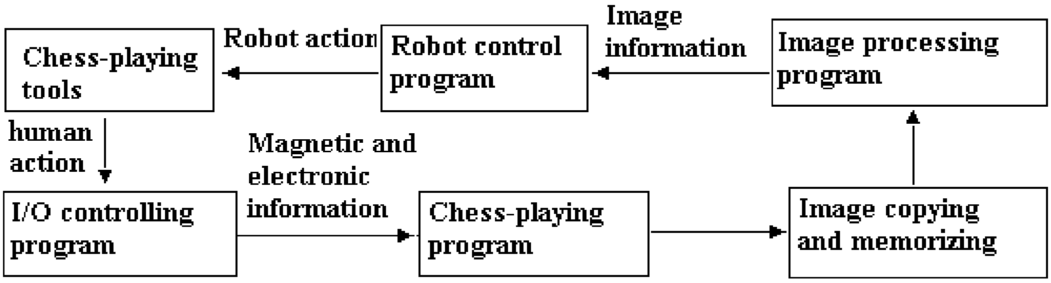

Fig. 2 Flow chart of the chess-playing robot

The chess-playing robot is designed to play chess with people. Firstly, when people take some action, the changing information of the target (chess-playing tools) must be acquired and send to the robot. The information is the integration of the magnetic, electronic and image signals by means of relevant sensors in the chess-playing tools. After certain analysis, the robot runs the chess-playing program and analyses information just received, responses and sends motion commands. It is no doubt that the reliable sensors, efficient processing program and real-time transferring are the key points [6].

We made some alternation to the chess-playing tools embedded with the hall sensors. There is a M lines and N columns array of hall sensors. The output ports of the SERCOS bus control the power supply of the sensors in each column, and the input ports of the SERCOS bus get the signal of the sensors in each line.

4 Controller Architecture

4.1 Hardware Architecture

The hardware architecture of the whole system is divided into NC control layer, SERCOS interface layer and motion execution layer (Fig. 3) [7].

Network devices, IPC and teaching box NC comprise the NC control layer. With the network devices, we can control the chess-playing robot remotely. The CPU and some other characteristics in IPC make the whole robot system reliable. The teaching box is also very necessary for some reason, such as self-learning of the robot, specified point teaching, track teaching, etc.

SERCOS interface layer consists of very important components such as chess-playing tools with our alternation, DSP, A/D, D/A and channels with specified characteristics and functions.

The motion execution layer mainly consists of motors and their actuators. And the working and measuring components monitor the operating status of the robot by means of various sensors, A/D, D/A and give feedback to guarantee the robot’s security and reliability.

The ever-increasing change and improvement of DSP become the most efficient promotion to the rapid development of the robot controller. It has proved that such hardware architecture takes advantage of every component's strongpoint and integrates all components excellently.

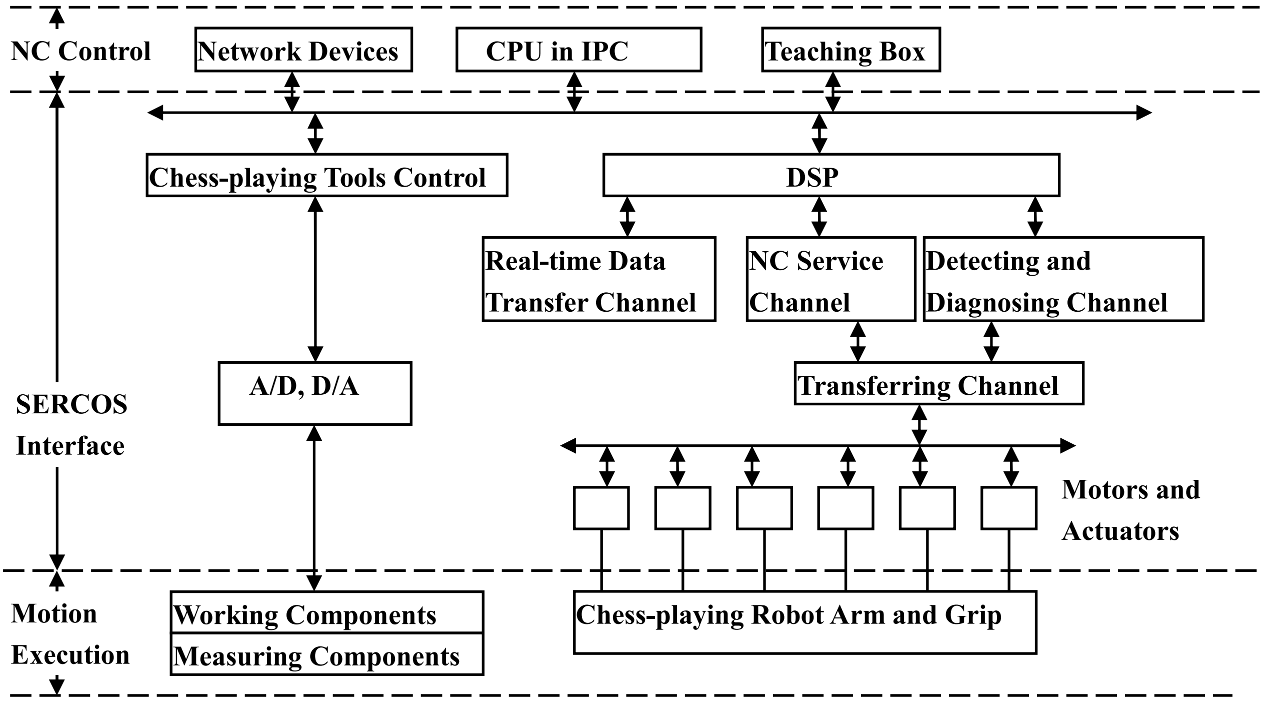

Fig. 3 Hardware architecture of system

4.2 Characteristics of Windows NT Plus RTX

The high-quality Windows NT plus RTX is chosen to safeguard the real-time property. Windows NT is better in graphic user interface and supported by a wide-range of hardware and software. RTX further extends the real-time function of the NT system [8]. The RTX real-time environment can fully achieve the process priorities of 0-127 and control the priority of the process to be executed [9]. The RTX system controlling is very efficient with a time delay of 45μs in communication between RTSS process and WIN 32 process and of only 8μs for inter-RTSS process communication. RTX is better than NT in real-time properties and guarantees higher quality pulses.

The execution programs can't operate the hardware directly in the NT system. However, RTX gives us access to the hardware and provides us with ample real-time API (Application Program Interface), which ensures the real-time property and accuracy of the entire control system.

4.3 Software Architecture

The software architecture is divided into the NC managing layer, the SERCOS interface layer, the task execution layer and the motion control layer (Fig. 4).

NC managing layer. It consists of several parts as following: The operating and controlling console is mainly for the interaction between players and the robot controller. The users can choose the difficulty level and the precedence relationship of playing. A real-time and on-line display of the whole playing process is provided. The instructing and teaching module's main function is off-line path planning, teaching and instructing. The chess-playing software and the intelligent library running background in the IPC provide necessary conditions for the image-processing module to analyze and extract the information of the chess-playing tool, so that the robot can make exact decisions and reach its goal. It is convenient to add other network devices through the remote network communication module. The rich net resources make it possible to control the robot remotely. Also many users can play the game at different places.

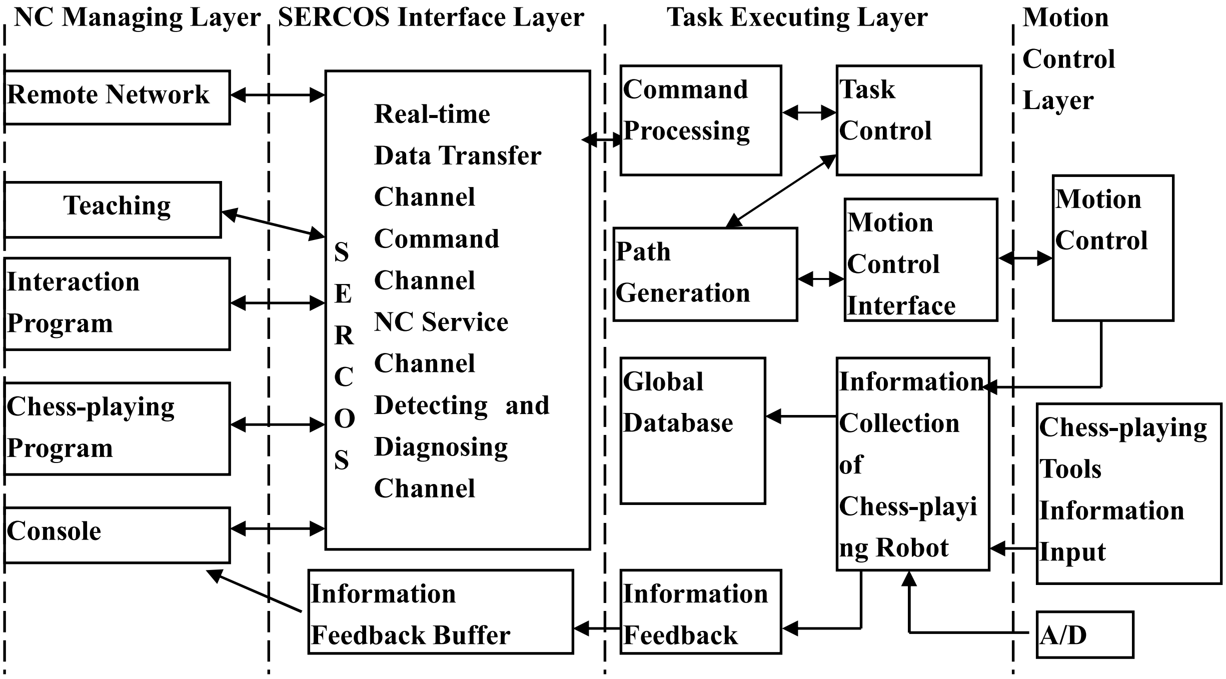

Fig. 4 Software architecture of system

SERCOS interface layer. There are the MST, AT and MDT in the SERCOS protocol. MST is from the NC managing layer to the task execution layer. The NC managing layer controls the synchronization of SERCOS interface by MST. AT, which is from the task execution layer feedbacks all kinds of task information to the NC managing layer. MDT sends the control commands to the task execution layer. The SERCOS bus interface card is connected to the IPC by EISA bus and wired to the robot servo system by fiber optic. The SERCOS bus provides IPC and robot servo system totally 6 communication channels for the aim of transferring data stream and command stream. Real-time data channel and real-time value channel consists of real-time data transferring channel. The former transfers the real-time data from the path-planning module to the task execution layer. The latter feedbacks the real value to the path-planning module after relevant commands. Command channel is aimed at transferring command information to servo driver and activating the inner command or function. The command control module transferring data and command to the servo driver uses NC service channel. So, the communication between this channel and the following task execution layer is guaranteed by SERCOS interface. Supervising and diagnosing channel can supervise the servo system, the execution process of the robot arm and transfer the data when in trouble.

Task execution layer. The task execution layer gets instructions from the manager layer and forms the control instructions. There are command processing, task controlling, path generating and motion control interfaces in this layer. The command-processing module receives instructions through SERCOS interface, processes the instructions and executes accordingly. Then the relevant data are put into the task buffer. After necessary processing, the task-controlling module activates the path-generating task according to task data. The information collection module of the robot processes the status information of the robot's arm, body, chess-playing tools, and the external analog information from A/D converter module every 8ms and transfers collected data to the user control console through shared memory. The manager layer and task-executing layer are run in WIN32 and RTSS respectively. RTX's inter-process transfers data between the two platforms. Same interface for all modules to communicate was designed. The communications details are packaged by the communication interface that allocates each module its exclusive ID number for sending and receiving operation.

Motion control layer. The motion control layer integrates servo driving, status checking and mechanical execution. The servo system controls the motor and the mechanical parts according to the data and commands. Thanks to the abidance of RTX technology to international standard and the abundant API resources, the system is simple in structure and easy to program, debug and maintain. Tasks with different real-time demands are assigned with relevant functions, which are favorable to making the system configurable and modular, and the motion control effectively.

5 Conclusion

The chess-playing robot is based on general operating system—Windows NT and RTX from VenturCom Corporation. High-capability SERCOS bus and its interface card can realize real-time position and motion control with high accuracy. The whole system is friendly in graphic user interface, excellent in management and performance controlling algorithms with high real-time property. Modularization in the software design of the control system makes it convenient to modify and append controlling algorithms. Its better expansibility and agility guarantees the real-time property and high-intelligence of the interaction between robot and player. The whole system which runs in practice proved to be reliable, high quality and intelligent.

Reference

[1] Th. Borangiu, F.D. Anton and S. Anton: Open Architecture for Robot Controllers. In: *2010 IEEE 19th International Workshop on Robotics in Alpe*, (2010), p. 181-186.

[2] Kailiang Zhanga, Zize Lianga, En Li, Degang Yang: Research on an arc welding robot controller with open architecture and the swing-welding interpolation algorithm. In: *Advanced Technology of Design and Manufacture (ATDM 2010)*, p.391-396.

[3] Lei Wang, Yongping Hao, Fei Wang, Hongyi Liu: Experimental Study of Force Control Based on Intelligent Prediction Algorithm in Open Architecture Robot System. In: *2007 IEEE International Conference on Robotics and Biomimetics*, p. 1675-1681.

[4] SERCANS SERCOS Interface Assembly, Germany Indramat Corporation (2001).

[5] SERCON816 SERCOS Interface Controller Reference Manual Version 7.4, Germany Indramat Corporation (2003).

[6] Xun Wang, Chunsheng Li, Yinqing Zhou. Journal of Beijing University of Aeronautics and Astronaustics, Vol. 20, No. 4, p. 402-406.

[7] INTERNATIONAL STANDARD—Electric Equipment of Industrial Machines—Serial Data Link for Real Time Communication between Controls and Drivers, IEC/TR 61491 Ed.1.0. 2010.

[8] VenturCom Inc. RTX 5.0 Users'Guide . Cambridage , MA, (2000).

[9] Wu Zhenwei, Li Bin, etc: the Design and Realization of Robot Controller Based on PC, In: *Robot* (2000), Vol. 22(7), p. 738-740.

Frontiers of Manufacturing and Design Science III
10.4028/www.scientific.net/AMM.271-272

A Chess-Playing Robot Based on SERCOS
10.4028/www.scientific.net/AMM.271-272.1536