THE APPLICATIONS OF OPC UA TECHNOLOGY IN MOTION CONTROL SYSTEM

Min Wang#, Hong Luo, Min Li, Jinmei Dong, Ruishi Mao, Tiecheng Zhao,
Institute of Modern Physics, Chinese Academy of Sciences,
Lanzhou 730000, China

Abstract

The establishment of data model is more abundant based on OPC UA (Unified Architecture) technology, which has good platform independence and high reliability. Thus it becomes a new direction in the field of data exchange of industrial control. In the paper, the motion control model based on redundant ring network is built by using NI 3110 industrial controller and servo motors. And the data structures used in parallel communication between the upper computer and multi-terminal motors are designed by using OPC UA technology. So the problem of inconvenient data exchange between the RT system of lower controller and the Windows system of upper computer may be solved.

INTRODUCTION

Motion control is widely used in modern industrial automation. The design is based on the application of the heavy ion accelerator in Lanzhou, China, which is firstly used in motion control of four beam diagnostic detectors. OPC UA (OPC Unified Architecture) technology with the advantages of good platform independent, security and so on, has become the main means to the data interaction in present industrial control, and is widely used in distributed control system.

THE ADVANTAGES OF OPC UA TECHNOLOGY

The traditional OPC specification is based on the COM/DCOM technology of Microsoft, which cannot meet the requirements of modern industrial automation in the aspect of interoperability, security, reliability and so on [1]. So OPC foundation released the newest unified method of data communication, namely OPC UA, which not only covers OPC DA, OPC HDA, OPC A&E, OPC security, but also expands many new functions [2].

Compared with the traditional OPC technology, OPC UA has the following advantages:

- Unified access approach, which means it can provide a unified address space and service model, but also has the function of semantic recognition, solves the problem that is the same information cannot be accessed in a unified way.
- Reliability, which means adjustable timeout can make the mistakes found and corrected, all this makes us deal with errors and failures of communication more easily.
- Security, which means the technology of underlying communication used for message transfer between the applications based on OPC UA provides the functions of encryption and marking.
- Independency of platform, which means OPC UA technology based on the SOA (Service-Oriented Architecture) of Web Service makes the application development no longer dependent on any particular operating system.
- Relevance of data, which means OPC UA provides the correlation functions of data nodes, rather than puts the node as a single data point [3].

#wangmin@impcas.ac.cn

THE ARCHITECTURE OF MOTION CONTROL SYSTEM

The motion control system is used for the motion control of beam diagnostic detectors of the cyclotron. The beam diagnostic detectors contain one SS (scintillation screens), one FC (faraday cup) in LEBT (low energy beam line), and another FC, one scraper in MEBT (medium energy beam line), so five servo motors are needed for the motion control (see Fig. 1).

Figure 1: Photos of the field in motion control.

Because Kollmorgen servo drive supports EtherCAT communication protocol [4], which has obvious advantages in topological structure, clock

ISBN 978-3-95450-146-5

synchronization, data transmission speed, construction cost and so on compared with other communication protocols, and the NI company has developed many LabVIEW software toolkits based on Kollmorgen servo drive, which can greatly reduce the difficulty and period of the development of upper application software, so Kollmorgen servo motor and NI 3110 industrial controller is selected for the motion control system.

The motion control system is composed of a plurality of servo drives and motors, a NI 3110 industrial controller and a client computer (see Fig. 2). Each servo drive is connected to a servo motor, servo drive communicates with each other through the EtherCAT interface. In order to build the redundant ring network of EtherCAT and improve reliability of the system, the first drive and the last drive are connected to the output and input of the EtherCAT interface ports of NI 3110 industrial controller, which makes NI 3110 communicate with each drive [5,6]. Because of the good platform independence and security of OPC UA technology, NI 3110 has the RT system is used as the server to display the real-time data and receive control commands from the client has the different Windows systems in the form of OPC UA.

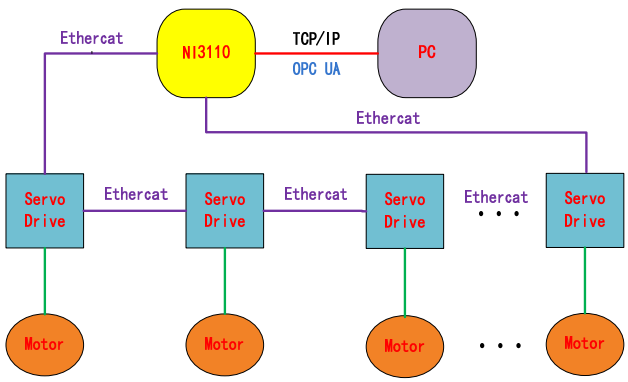

Figure 2: Architecture of motion control system.

The system can also be extended to a variety of topologies, such as star, tree and daisy chain type etc. Multiple NI 3110 can be connected through the EtherCAT protocol, and each NI 3110 also can be connected with any other hardware devices supporting the EtherCAT, not only the servo drives. The data from NI 3110 can be displayed by OPC UA, and the commands from multi-client computers are received by OPC UA too.

THE DATA STRUCTURE USED FOR MOTION CONTROL BASED ON OPC UA

Because OPC UA provides the correlation functions and the semantic characteristics of data nodes, so it is very necessary to design the data structure of the motion control system based on OPC UA, which will make the program easier to maintain, enhance the readability of the program, and facilitate the development of client program.

Based on the OPC UA application of LabVIEW, the data structure can be freely built according to the different applications. The data structure based on OPC UA for the servo motor named "motor_1" is designed, which is similar to the other servo motors (see table 1). The *Folder* node is used to distinguish each servo motor; the *Item* node is used to sort the data for motion control based on OPC UA, each *Item* node can define the name, access method, data type and description; the *Property* node belongs to the corresponding *Item* node, and also contains the definition of name, access method, data type and description, which do not appear in table 1 as a result of length reason.

Table 1: Data Nodes of the System

Folder Name: motor_1	**Item_1** Name: data Access: read only Datatype: float Description:	**Property_1** Name: position **Property_2** Name: scale_value
	Item_2 Name: command Access: read/write Datatype: int32 Description:	**Property_1** Name: enable **Property_2** Name: start **Property_3** Name: stop **Property_4** Name: home **Property_5** Name: distance
	Item_3 Name: status Access: read only Datatype: boolean Description:	**Property_1** Name: enable? **Property_2** Name: complete? **Property_3** Name: fwd_limit? **Property_4** Name: rev_limit? **Property_5** Name: scale_safe?

From the table 1, the *Item* of "motor_1" is divided into three categories, which are named with "data", "command" and "status". The "data" item includes two properties, which are named with "position" and "scale_value" – "position" means the position feedback from servo drive; "scale_value" means the value of displacement sensor acquired from the digital I/O of servo drive. Because the motor encoder is relative encoder, in order to improve the accuracy of position acquisition and bring convenience for calibration of detectors, resistance displacement sensor is used to get the correct absolute position of motion. The "command" item includes five properties, which are named with "enable", "start",

ISBN 978-3-95450-146-5

"stop", "home" and "distance". "enable" is used to enable the motor; "start" and "stop" is used to start and stop the motion tasks; "home" is used to find the motion origin; "distance" means the distance of motion tasks. The "status" item includes five properties, which are named with "enable?", "complete?", "fwd_limit?", "rev_limit?" and "scale_safe?". "enable?" means whether the motor is enable; "complete?" means whether the motion task has been completed; "fwd_limit?" means whether the positive limit switch is triggered; "rev_limit?" means whether the negative limit switch is triggered; "scale_safe?" means whether the value of displacement sensor has been out of the safe range.

The control interface of the client upper computer is shown in the Fig. 3. From the Fig.3, the motion control interface of SS in MEBT is divided into two parts. The left part named "Motor_1 Control" of the interface with the functions of enabling the Motor, finding the origin, starting and stopping the motion task, inputting the distance of motion, observing the feedback of motion, getting the value of displacement sensor and calculating the distance between the detector and the centre of chamber is used to complete motion tasks. The right part named "Motor_1 Status" is used to get the status containing "enable?", "complete?", "fwd_limit?", "rev_limit?" and "scale_safe?".

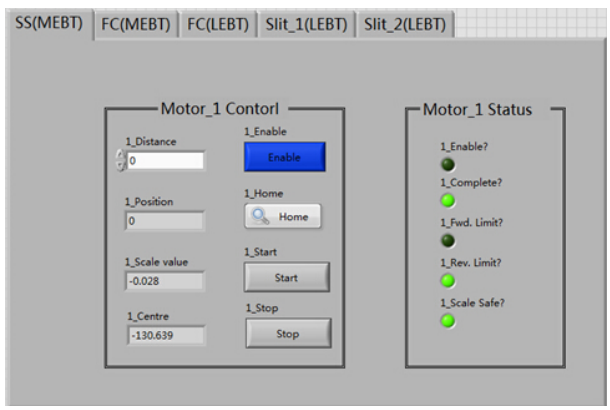

Figure 3: Interface of motion control system.

Thus it can be seen that the flexible and easy way of data organization of OPC UA can make the program of NI 3110 as the lower computer more easily to extend and maintain, but also make the development of program of the client as the upper computer more convenient .

CONCLUSION

With the development of industry control， the data communication needs the better reliability and safety, independency of platform and so on. In the paper OPC UA technology is selected for building the motion control system, the architecture of motion control system and the data structure based on OPC UA are designed, and the motion control system which can verify the advantages of OPC UA technology is being tested.

ACKNOWLEDGEMENT

We would like to thank my colleagues and Dianling Wang, Yao Li, Wenlong Jing form National Instrument for all of their help.

REFERENCES

[1] M. Wolfgang, L.S. Helmut, "OPC Unified Architecture – the future standard for communication and information modeling in automation", ABB Review, March 2009, p. 56-61(2009).

[2] Wikipedia website: http://en.wikipedia.org/wiki/OP-C_Unifie-d_Architecture

[3] L.S. Helmut, M. Wolfgang, "OPC UA Service-oriented Architecture for Industrial Applications", Software Trends, November 2006, ISSN0720-8928 (2006).

[4] J. Katzel, M.T. Hoske, "Industry Ethernet Protocol", Control Engineering Chinia, March 2007, p. 34-39 (2007).

[5] National Instrument website: http://www.ni.com/-white-paper/11700/en/#toc2

[6] EtherCAT Technology Group, "EtherCAT Physics and Slave Controller Integration", (2008).

ISBN 978-3-95450-146-5