

# The Real-Time Computer Numerical Control Based on RCS

Zhiman Liu<sup>1, a\*</sup> and Zhengyi Hu<sup>1</sup>

<sup>1</sup>Changchun Automobile Industry Institute, Changchun, China

<sup>a</sup>arnold0110@sina.com

**Keywords:** Computer numerical control (CNC); Real-time control system (RCS); Communication architecture; Communication management system (CMS); Neutral message language (NML)

**Abstract.** With the science and technology progress, CNC system is towards high performance, high precision, high speed, high flexibility and modularity. The traditional computer numerical control (CNC) system is mostly special equipment; users can't extend the function of the numerical control equipment, to meet their special needs. The reuse and reconfigurable ability for the control procedure is required in open CNC system. The Real-time control system library is applied in the software development process in numerical control system. In this paper, a CNC system based on RCS is design. Principle of communication based on RCS is firstly described, and then implementation of CNC based on RCS is completed. The modifications of the specific behavior of the machine tool and operating are limited on local control program, thus greatly improving the openness of CNC system.

## Introduction

With the science and technology progress, the development of numerical control technology is more and more fast, CNC system is towards high performance, high precision, high speed, high flexibility and modularity. Users want to be able to freely choose CNC device, driving device, servo motor, application software, such as numerical control system of each constituent elements, and be able to adopt the method of standard, simple combined these elements [1,2]. To satisfy with the open requirements, some research institutions and manufacturers have carried out the research since the 90s of last century. In Europe, America, Japan respectively established the research organization for the open numerical control system, such as OSACA, OMAC and OSE. The periodic progress has been made and announced the part of the research results from these three plans. For our country have no substantive design specifications and representative products in developing open CNC system of unified standard. So we put forward adopting the RCS method and the RTLinux platform for the open CNC system [3, 4, 5]. At present, the design scheme had been implementation in hardware platforms, and running in good condition, and prove that the attempt is successful [6, 7].

The RCS library is a C++ class library intended for multi-platform real-time distributed applications. It has been compiled and tested on several platforms including MS-DOS, MS Windows, and several UNIX Workstations.

The RCS library is developed by the American institute of standards and technology (NIST), it can be used to develop real-time control, planning, fault tolerance and adaptive control technology in the integration of complex control system. CNC is a kind of automation control system to higher degree, therefore, the RCS can be used to realize CNC system, so as to ensure the real-time and openness performance of CNC system. The structure of a single concurrent process module using NML is shown is Fig. 1 [8].

According to the RCS design method, the complex control problem is decomposed into the small subproblems to maintain and develop and modify easily. The hierarchical structure is designed, in order to reduce the complexity of the real time control system. Therefore, typical RCS design begins with the task analysis of the system. First of all, designers have to determine which subsystem can be decomposed into, the whole system can be decomposed into what is correspond to the subsystem of sensor and actuator, the subsystem can perform what tasks, which tasks are related to each other, what

is the flow of information or data flow between the subtasks, and select which task sequence to produce the desired performance and so on [9].

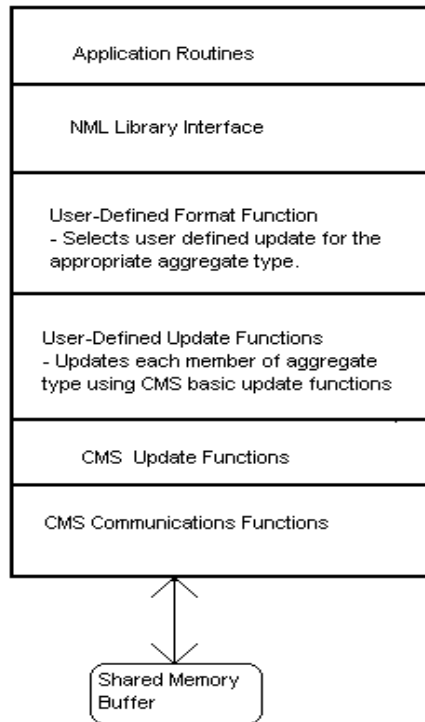


Figure 1. The structure of a single concurrent process module using NML

The remainder of the paper is organized as follows. In section II, principle of communication based on RCS is described. The implementation of CNC based on RCS is completed in section III. The several concluding remarks are made in section IV.

### Principle of Communication Based on RCS

The RCS design method provides a concise, standard architecture, the key is to use the CMS and NML communication tools in the RCS library.

CMS provides access to a fixed-size buffer of general data to multiple reader or writer processes on the same processor, across a backplane, or over a network. Regardless of the communication method required, the interface to CMS is uniform.

NML formerly known as the Neutral Manufacturing Language provides a higher level interface to CMS. It provides a mechanism for handling multiple types of messages in the same buffer as well as simplifying the interface for encoding and decoding buffers in neutral format and the configuration mechanism.

Fig. 2 illustrate the structure of a typical RCS application using NML. The application is distributed across three computers. Processes 1, 2, and 3 are able to write directly into the shared memory buffers they use because they are located in the same computer or backplane.

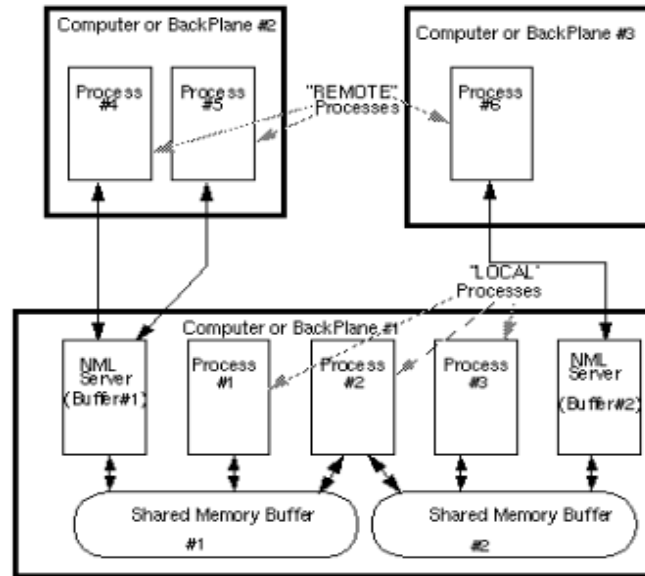


Figure 2. The structure of a typical RCS application

It is for this reason that they are labeled "LOCAL". Processes 4, 5 and 6 can only access the buffers through an NML Server and are therefore labeled "REMOTE". The description might need to be complicated in a system with buffers in more than one machine. Processes would then need to be described as local or remote with respect to a particular buffer.

### Implementation of Computer Numerical Control Based on RCS

According to task partitioning of numerical control, the CNC system is divided into man-machine interface module, and the task controller module, I/O controller module, motion controller module. The first three modules in non-real-time user space, motion controller module in real-time space. With non-real-time user mode between modules, the system adopts mature RCS library to build stable communication, and between non real-time controller module and real-time motion controller module, the shared memory way has been adopted for the internal communication.

**Definition of Communication Channel.** In order to pass information between RCS modules, inevitably there is a communication channel between the modules. To establish communication between processes, must have two conditions: determine the memory buffer can be shared and establish process to access the buffer. The communication channel between buffer and processes is founded by calling the NML functions, as for which RCS process access to which buffer is specified by NML configuration file. NML configuration files are ASCII file, the file is a list of buffer and processes, included in the list items can be the size of the shared memory buffer, buffer address, access to the buffer, which process access which buffer, control process for the buffer and so on[10].

The communication structure as shown in Fig. 3, the command messages from the upper module is passed to the lower modules, namely the upper module request the implementation of an operation. Status messages to their parent module from the underlying module, the message is reflect the characteristics of the underlying module current state of data information.

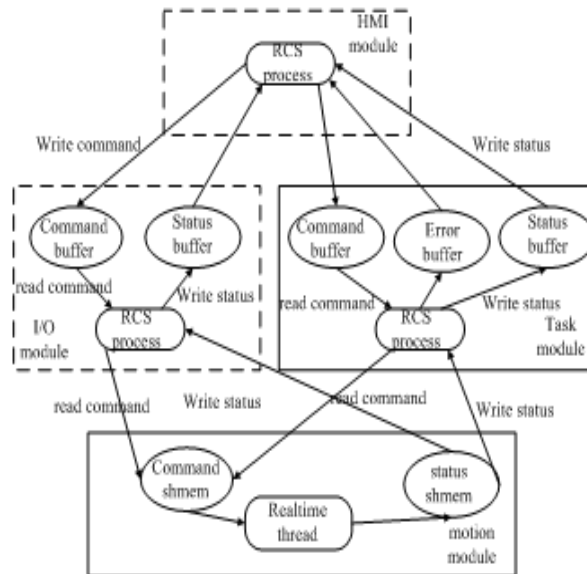


Figure 3. Communication model of the CNC

Non Realtime Communication. The three layer architecture is constituted with the RCS communication mechanisms: the top is the graphical user interface, the second is the task controller, and of the third layer is I/O controller and motion controller. To logically subdivided commands, status and error message passing, divides the five NML channels according to the functions:

- a) CmdBuffer, As the command channel, issued user commands by the interface to task controller;
- b) StatusBuffer, As the state of channel, dedicated to the state feedback from the task controller to the man-machine interface;
- c) ErrorBuffer, As the error channel, feedback error message from the task controller to man-machine interface;
- d) IoCmdBuffer, As the command channel, issued task commands to I/O controller;
- e) IoStatusBuffe, As the state of channel, dedicated to the state feedback from the I/O controller to the task controller;

By the above description, the task controller is the scheduling and decision center of the CNC system, and I/O controller is only responsible for the I/O status update, without any decision. So, the only task controller to the error channel between HMI.

Task controller is the core of the whole system, it respond for the task scheduling and the coordination of the of information transfer between the HMI module, the motion controller module and the I/O controller module. According to the different characteristics and requirements of the real-time communication between tasks, the RCS library based on the shared memory and the communication mechanism of NML have been used. Do not require a high communication rate between task controller module and man-machine interface, the average communication time of not less than 1/10 seconds, after taking into account for the development of the remote monitoring module, so the use of NML communication mechanism of RCS database in the system. NML supports the TCP/UDP communication protocol, multiple processes and support read the information in different location to satisfy the above requirements.

Realtime Communication. Motion controller is a real-time thread running in kernel mode. As a result of pipeline message communication mechanism for the RCS is only applicable to user mode between common tasks, so it cannot be used between motion controller and task controller. The shared memory mechanism provided by the RTLinux or RTAI kernel is adopted between motion controller and task controller.

**Make NML Application.** Define NML Message Vocabulary with C++ Class. In RCS design, designers need to determine the type of transfer of information between different modules. Typical message should include the following three parts: save the message size and a unique identifier (ID)

of variable data, data variables as the application details, and the update () function. The update () function is the function of tell NML how to update, encoding or decoding, and encoded can perform message identification number of read and write operations. The class diagram for status information of task controller is shown in Fig. 4. The class diagram for command information of task controller is shown in Fig. 5.

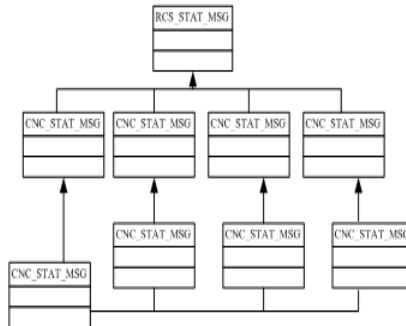


Figure 4. Class diagram for status information of task controller

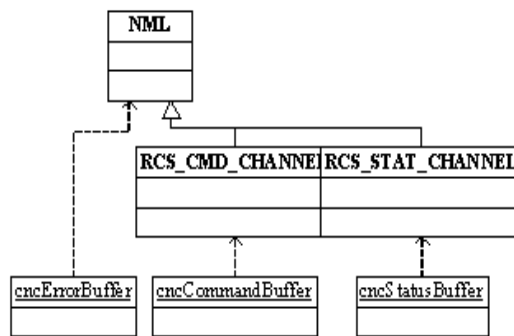


Figure 5. Class diagram for status information of task controller

Create NML Configuration File. Most of the options available in CMS are specified in a configuration file. There should be one line in the file for every buffer that is required and every process using CMS should have one line for every buffer it accesses. Currently there is no difference between a CMS configuration file and an NML configuration file. The files may contain only ASCII characters, and can be generated with any text editor.

Local processes can connect to buffers using GLOBMEM, SHMEM, FILEMEM or LOCMEM. Remote processes can connect using TCP, UDP or RPC.

SHMEM is intended for communications between tasks managed by the same operating system. The operating system allocates the memory to be shared so users do not need to specify one. Users have a choice of mutual exclusion techniques with include something similar to GLOBMEM's method, using an operating system semaphore or mutex, or disabling and enabling context switching or interrupts during the appropriate critical sections.

Create Connections to NML Buffers in Programs. Finally, create connections to NML buffers in the programs and call read and write methods in programs to transfer messages. The communication among various modules could be achieved by RCS for the CNC system.

At the same time of the task initialization, the channel of command, status, errors will be initialized based on the NML communication configuration file. The channels will be closed when the exit of the task controller. By the peek function call through command channel, the control information is read into the task controller; through write function call through state channel, state information will be transmitted out to the HMI of the CNC; by the write function call through the error channel, the error description string information will be sent out to the HMI.

## Conclusion

Recently, the reusable and reconfigurable capability is researched for the development of open architecture CNC system as an important performance. In this paper, a CNC system based on RCS is design. Firstly, Principle of communication based on RCS is described. Then, the implementation of CNC based on RCS is completed. Finally, the several conclusions are remarked in the last section. The principle of RCS, the development method is adapted to the task development of CNC system, which is convenient for numerical control system to realize the structural remodeling and function expansion. It has greatly improved the openness capability of CNC system.

## References

- [1] W.J.Wu, J. Zhang, Z.S.Peng. The realization to the communication between NC system modules based on the RCS Library [J]. Journal of Chinese Computer System, 2007, pp88-91.
- [2] M.Humphrey. Experiences using RT-Linux to implement a controller for a high speed magnetic bearing system. IEEE Real-Time Technology and Applications Symposium. University of Virginia.1999, pp: 121-130.
- [3] Huang H M. An Operator Experience with a Hierarchical Real-Time Control System (RCS)[R]. New York, USA: National Ins titute of Standards and Technology,1996.
- [4] Gazi V,Moore M L, Passino K M,et al. The RCS Handbook: Tools for Real-Time Control Systems Software Development [M]. New York: John Wiley & Sons, Inc., 2001.
- [5] Cheraghi SH, Waram V. Scheduling of multi-spindle CNC gantry mills [J]. Journal of Scheduling, 2002, 5(1): 3-22.
- [6] Mekid S, Pruschek P, Hernandez J. Beyond intelligent manufacturing:A new generation of flexible intelligent NC machines[J]. Mechanism and Machine Theory, 2009, 44(2): 466-476.
- [7] Morales-Velazquez L. Open-architecture system based on a reconfigurable hardware-software [J]. Journal of Systems Archi-tecture, 2010, (56): 407-418.
- [8] Kotani T, Nakamoto K, Ishida T, et al. Development of CAM system for multi-tasking machine tools[J]. Nihon Kikai Gakkai Ronbunshu, C Hen/Transactions of the Japan Society of Mechanical Engineers, Part C, 2009, 75(757): 2589-2595.
- [9] Group OAW. OMAC Baseline Architecture Functional Requirements Version1.0 [M].2002.
- [10] Shin SJ, Um J, Yoon JS, et al. Developing ISO 14649-based conversational programming system for multi-channel complex machine tools [J]. International Journal of ComPuter Integrated Manufacturing, 2009, 22(6): 562-575.