2009 International Conference on Artificial Intelligence and Computational Intelligence

Study on ARM-based Embedded NC System

Wang Jun
College of Mechanical & Electronic Engineering
China Jiliang University
Hangzhou, Zhejiang Province, China
Wangjun8762103@sina.com

Abstract

At present, study on open numerical control (NC) system is increasingly getting attention. Most of these studies concentrated their aims at realizing the open system structure based on PC or IPC (industrial personal computer). Few involved the category of embedded NC (ENC) system, but expectant goals could not be achieved at the performance of real time (RT) and reliability of system response. In this paper, a new ENC system based on ARM and programmable multiple axes controller (PMAC) is presented. ENC system, which enhanced performance and reduced cost, was chosen to replace traditional IPC one. Wherein ARM processor was chosen as master CPU, meanwhile PMAC2 was chosen as slave CPU, TCP/IP protocols was solidified with W3100A, and μ_c Linux was chosen as operating system (OS). The system was proved to be good at dealing with multiple tasks processing, RT and reliability of motion control. In future, ENC system may be expected to be a kind of important structure of open computer integrated machining system (CIMS).

1. Introduction

At present, machining industry is being transferred to China, and the machining features have been changed from the past big batch and low precision to small batch and high precision now. Therefore, high flexibility and precision is required in machining equipment. In order to meet this demand, more and more multi-purpose open NC equipments have emerged.

the study on open NC equipment is now concentrated on three fields: the first field is in interpolation algorithm, cutter compensation and synchronization, many scholars have set up algorithm models of various cutter machining states, and made great efforts to reduce machining errors [1-3]; the second one is in framework of NC system, at present, these studies are mainly focus on how to realize two level control based on PC or IPC [4]. It's advantage is familiar developing environment and quick developing speed, and disadvantages is expensive costs, bulking figure and bad RT performance; the third field is the study on ENC system based on MCU-PMAC configuration [5]. Because high performance demand of micro control unit (MCU) and real time operation system (RTOS) is expected, meantime, the PMAC is very expensive and the developing environment is relative strange comparing with windows, few studies have been done in this field.

In this work, we propose an ENC structure, which adopted ARM9 as upper computer, PMAC2 as lower computer and μ_c Linux as RTOS.

This system is a small CIMS, which may be summarized as network, ARM9 and PMAC2 three layers structure including transfers, control and operation. it may be controlled by remote host through network module W3100A; and the embedded system may be seen as an independent host to realize operations such as writing program, simulation, download and control etc.; the third layer is PMAC2, which realize some RT tasks such as interpolation algorithm, cutter compensation and synchronization, configuring above mentioned functions was the hardware task, while software configuration included two aspects, that is cutting out a set of OS and developing an application.

Finally, an experiment was carried out to testify the functions of system. A conclusion is draw that the system can deal with multiple parallel tasks and may be expected to become a kind of new NC equipment.

This paper is organized as below: Section 2 introduces the hardware structure of the ENC system; Section 3 presents ENC system software design; Section 4 presents performance testing and results; Section 5 presents discussion and conclusion; Acknowledgment and references are written as end of the paper.

2. Hardwire Structure of the ENC system

This system adopts special motion controller as lower computer. This configuration has below 4 advantages.

- Abbreviate developing period.
- Enhance system reliability.
- Enhance system control performance.
- Reduce software developing cost.

The S3C2410 processor produced by SANSUM was chosen as ARM main processor. S3C2410 is a kind of

978-0-7695-3816-7/09 $26.00 © 2009 IEEE
DOI 10.1109/AICI.2009.279

processor structured with 32 bits and RISC instruction set based on ARM9.With less volume, less power consumption, less costs and higher performance, it can support Thumb (16bits)/ARM (32bits) two kind of instruction set, and it provides a nice compatibility with 8bit/16bit devices. PMAC2 was selected as motion controller. PMAC2 is a kind of multi-axis motion controller produced by The Delta Tau Data Systems Inc. And it is one of the strongest motion controllers in the world. The PMAC2 is open controller. PMAC2 is essentially an integrated RT and multi-mission computer too. The PMAC2A-PC/104 selected in this paper is a compact version. It can control 8 servo motors at the same time. And it communicates with ARM9 by the USB interface [6].

Embedded Web module is mainly constructed by data processing and network interface. Network interface adopts 10M/100M self-adaptive Ethernet interface to join into internet. Data processing part is constructed by ARM9 and memorizer to realize network layer protocols and the functions of the application layer. The base of running Internet is TCP/IP protocols, and the reality of TCP/IP protocol stack is the key that the embedded system could join into the internet. In order to enhance the expanded capability and processing velocity of this module, the hardware reality that can provide TCP/IP protocol was adopted. The chip W3100A can realize protocols including TCP, UDP, IP, DHCP, ARP and ICMP etc. The advantages that use hardware to realize TCP/IP protocols is that the transportation velocity is more quickly, and this transportation does not occupy the resources of ARM9. So The ARM9 can deal with more works of application layer. There are many kinds of methods to link embedded system to Internet, among these methods; the method with Ethernet is commonly used. The standard protocol that this method carries on is IEEE802.3. A 10M/100M self-adaptive Ethernet interface was realized with a 10BASE-T/100BASE-T common interface chip that tally with IEEE802.3 protocol. It exchanges data with W3100A by a M Ⅱ standard interface. The schematic figure is showed as Figure 1.

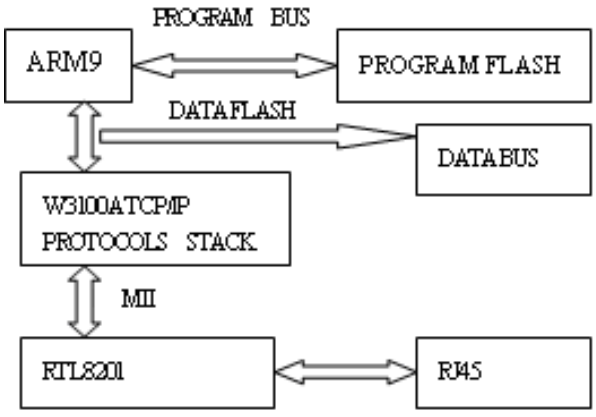

Figure1 Network interface schematic chart

The hardware of the system adopts master and slave two CPUs structure. The master CPU is ARM9 processor, which takes charge in management work such as keyboard, display and network communication etc. Whereas the slave CPU, that is PMAC2 motion controller, specially answers for the processing works of motion control. The communication between PMAC2 and ARM9 rely on reading or writing some address to transport instructions and data. The hardware structure of control system is showed as Figure 2.

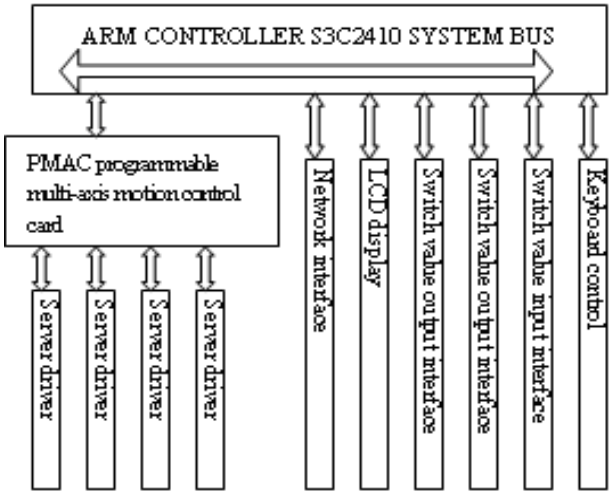

Figure 2 Control system schematic chart

3. ENC System Software Design

3.1. OS structure of the ENC system

μ_c Linux is a RTOS. It has excellent representation in reliability, RT performance and management efficiency. The control software were constructed where μ_c Linux was adopted as kernel, solidified TCP/IP protocol was adopted as network components, the algorithm of the motion controller was adopted as main application programs, and the third party plug MINIGUI was adopted as tool to write assistant communication program.

In short, there are following 4 characteristics for the OS of embedded system:

- Lower cost and short time to enter market.
- RT performance.
- Light weight.
- Safety and reliability.

μ_c Linux is gradually shaped through modifying and updating kernel according to different demands of the embedded system. The differences between μ_c Linux and ordinary Linux is the kernel: to suit with embedded

objective platform, kernel of the former is specially designed and modified from ordinary Linux, and kernel of the later is applied in general PC platform. In the history of the embedded system, many other general OS are also played as prototype of corresponding embedded system, for example, Vx-Works comes from 4.3BSD OS, and NT embedded comes from Windows NT.

By reconstruction, the application, formerly run on general OS, may quickly and conveniently be replanted to the operating environment of the embedded system, so does Linux. Linux applications which possess extensive open source codes have been replanted to the operating environment of embedded system, therefore, by using Linux to develop embedded OS, the developing speed can be accelerated, and the time coming to market can be shorted.

The essentials for designing embedded Linux mainly include that boot-loader, memory management for OS and processing management etc.

3.1.1. Bootloader firmware. Firmware, which completes system tasks cooperating with main programs, is programs stayed in ROM or Flash. It is solidified in ROM of target board to be used as starting up embedded software stored in RAM or ROM. Had been supported by CPU, it might be placed into RAM running during development. In embedded Linux developing, bootloader programs should be firstly developed to complete following tasks according to characteristics of target board:

Initialization CPU, Some configuration memories were initialized by bootloader. For instance, if MMU of S3C2410 was needed, it would be opened by using control command in bootloader.

Initialization necessary hardware, Necessary hardware of the board were initialized and configured with bootloader, such as initializing Memory, Flash, and ROM and interrupt controller. The interface device which downloads system image from host to hardware board is also initialized with bootloader.

Download system image, System image is just downloaded by bootloader. With command line provided by bootloader or reciprocal Shell interface the kernel image and file image may be indicated where they is downloaded, and the contents in memory of target board may be checked too. The bootloader of target board provide service program for accepting image, while the bootloader of host PC provide function to send and receive data through Ethernet card or serial communication port or other mode. After has send system image, if hardware allows, bootloader may provide command to writing downloaded image to Flash ROM. Generally, bootloader provide function to actuate Flash, provide command to support effacing and writing, thus, great convenience is provided to realize writing Flash and check it.

Initialization OS, An OS already downloaded may be started up by using bootloader. Bootloader may be designated to start up OS, no matter which is in RAM, Flash or anywhere.

3.1.2. Memory management. The memory management of OS is used to provide to OS consistent address mapping and some operation of memory page such as applying and releasing etc. These work, if there is not OS, should be done by application, that is, every application need manage its memory space well. Through embedded OS, the programmer assures that it can not getting over boundary for visiting memory and the safety of the OS consequently may be guaranteed. For example, every tasks just applies its local static memory space which can not be assigned to the others during running, the question of getting over boundary is thus not exist. But the static mode aggravates squandering of memory resources. According to their need, embedded applications may apply for memory spaces from OS, if it exists, and the management is conducted by the OS.

According to parameters provided by application when they apply for or release, the OS apply for memory to satisfy respective needs, subsequently, the developing procedure can be simplified, and the safety and reliability for running can be guaranteed. Some embedded CPU provides MMU, MMU, which provides function for memory address mapping and addressing, may conveniently manage memory of the OS. MMU, if it is provided, may be used to complete transfers from virtual address to physical address by the OS, therefore, all the applications may just address by virtual address. This addressing mode, which locates address of primary and auxiliary memory of the whole system, is named virtual memory in modern OS. MMU is the necessary condition for realizing virtual memory technique.

The management mode of virtual memory either allow system run application whose volume is bigger than that of physical memory or realize strategy that allot memory space according to need, either satisfy the demand of running speed of program or save physical memory space. By using 3 levels or 2 levels page-mode management corresponding to respective system structure, and MMU is utilized finishing the transfer from virtual address to physical address, Linux realize virtual memory mechanism. But many embedded system do not need virtual memory mechanism in the condition of demanding high RT performance, because that virtual memory mechanism may induce uncertain I/O time-blocking can not guarantee that running program end during allowing time; by redesigning and modifying memory management parts of Linux, Lineo company provides μ_c Linux to support memory management of micro processor unit without MMU. In order to sharing identical physical address for application, it adopts flat addressing mode to

complete address transfer, and modifies address of all addressing instructions of application loaded. So, for programmers, the questions of over boundary and memory protecting need not be thought over by themselves.

3.1.3. Tasks management. Many tasks dispatch may be realized by using OS. People who design system just need design tasks, multiple tasks dispatch may be finished subsequently by methods of the OS itself. General OS which realizes their tasks by processes or threads commonly does not allow changing its dispatch mode.

But embedded OS takes task as kernel, and task itself maybe ask system providing appropriate dispatch. For example, many embedded OS may provide dispatch mode of RT task while they can realize dispatch mode of non-RT too.

General Linux adapts time-sharing dispatch algorithm which does not allow to be occupied. Linux, which may distinguish the style of task by task description letter, adopts respective task dispatch algorithms to different dispatch. By adding strategy flag and correspondent task style, dispatching algorithm, new Linux dispatch algorithm will be realized too. In order to modify Linux to embedded OS of supporting RT task, two questions should be thought over.

The first question is interrupting processing. The response ability to exterior interrupt determines RT performance of system. In the kernel of OS, some operations need run under closing interrupt, and in this case, system response to exterior interrupt will be certainly delayed. This question can be resolved by double kernels mode which add one RT kernel between Linux kernel and interrupt, where, Linux kernel is dispatched as a task of the RT kernel. While RT kernel runs, operation of closing interrupt does not occur, and all the interrupt which need be responded real-timely will trigger RT kernel to run RT task, they will occupy Linux kern to run; other interrupt will be directly sent to general kernel to process.

The second question is process occupation. Linux is designed as general OS. Therefore, throughput is an important factor in designing. Especially in single CPU mode, because of the dispatching algorithms allowing occupation, so many operations will run to protect critical zone that extra spending happen to. Therefore, the throughput of OS is also greatly impacted, and subsequently, general Linux do not support occupying dispatch. But Linux which run in RT environment should discriminative deal with between RT task and general task by allowing RT task occupying ordinary task of lower PRI so that excellent RT performance can be obtained.

In order to enhance the ability that RT task occupies non-RT task, two kinds of methods that modify kernel are presented: the first method is occupation point method which settles a series of occupation points in the path passed by kernel to measure if RT need be run. In this case, the longest path between two occupation points is just the longest time that the highest PRI RT task is maneuvered. This method is realized in the kernel provided by Ingo Molnar of Red Hat Inc. The second method is that the kernel directly is reconstructed to occupational kernel. While higher PRI RT task is awakened, present task, if it is not in critical zone, will be directly occupied. This method is adopted in Hard Hat Linux of the Monta Vista Inc.

3.2. System application

System application which may be written with MINIGUI has friendly human-machine interface. The application stored in ARM9 realizes initialization, input, show and the communication function of the two CPU by using the functions of Pcomm32.dll of Delta Inc. its function framework is showed as Figure 3.

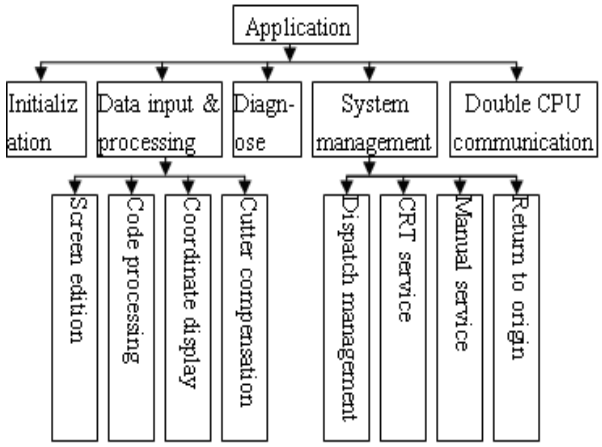

Figure 3 System software framework

By using download functions of dynamic link library Pcomm32 provided by Delta Inc, the initial module of system directly downloads system initialization file which is stored in NAND FLASH of ARM9 to PMAC2, initializes PMAC2 and activate PLC programs and RS-274G international standard(IS) code interpretation program. The OS input module completes input machining program, settles control parameters and compensation parameters etc.

Both storage mode input and NC mode input are supported and both key board and disk may be used as input terminal. System complete following works such as null code deleting, code testing and code transforming etc. after having finished input. Then, download function of Pcomm32 is maneuvered to download code to PMAC2. The display module which visits registers of DPRAM by maneuvering functions of Pcomm32 dynamically display parameters such as machine tool coordinates, workpiece coordinates, trace errorness and electrical motor state etc. Meanwhile, according to parameters and input data, the

motion trace of cutter may be calculated and the model of workpiece may be dynamically simulated too.

The communication module which may maneuver communication functions of Pcomm32 serves for modules above mentioned.

3.3. Control software of system

The control software includes interpolation module, decoding module, position control module and I/O processing module. The control software stored in PMAC2 is executed by PMAC2 to guarantee RT performance of system. Its function modules are showed as Figure 4.

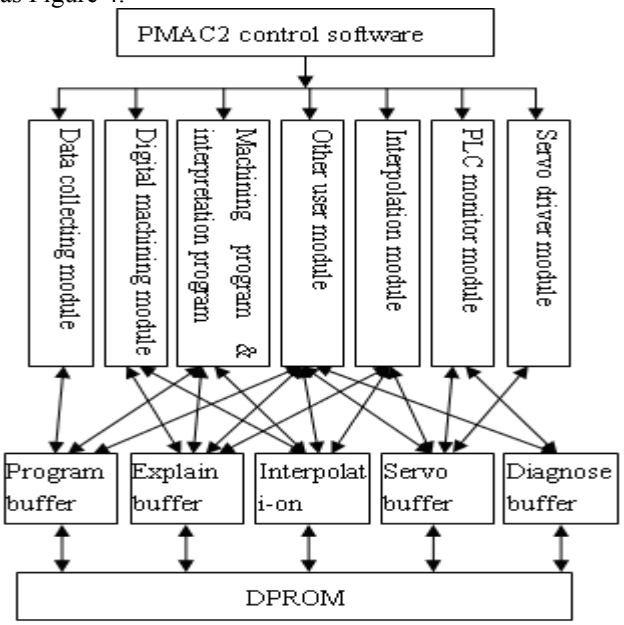

Figure 4 Topology of control module

Interpolation module includes following interpolation mode such as linear interpolation, arc interpolation and spline interpolation etc. PMAC2 also provides position-velocity-time (PVT) motion mode which may control graphics trace more efficiently and compactly. User may choose and mix discretionarily some work modes above mentioned, meanwhile, acceleration and deceleration may be realized automatically by the system.

Decoding module includes G code interpretation program, M code interpretation program and T code interpretation. Decoding module managed by the application edits and downloads motion program to fixed memorizer space of PMAC2. After having incepted codes which come from upper ARM9 and accord with RS-274, PMAC2 seeks automatically interpretation program from memorizer and execute correspondent operation.

The position control module, which controls actual position of the system according with order position, realizes position control by PID provided by PMAC2. Meanwhile, PMAC2 provides following functions for synchronization between electrical motor and external affairs such as position tracing, time-based control, position catching and position comparing etc.

3.3.1.PID control in servo motor. In a mechatronics system, in order to obtain excellent stability and dynamic performance, control loop of the system need be emended and adjusted. Its influence is great. Therefore, after basic characteristics (including machine transmission and electrical motor choosing etc) is ascertained, in system that PMAC2 is regarded as core, PMAC2 provides control loop algorithm PID+velocity/acceleration feed forward+NOTCH filter (notes: the effect of NOTCH filter is: because of some question existed in system such as lagging, stillness friction and curliness etc, the system generates harmonious wave vibration and the performances of system will therefore be degenerated. PMAC2 forms a filter which can prevent harmonious wave vibration by settling some parameters, and it almost satisfies requests in any occasion.

3.3.2. Coding of motion programs and PLC programs.

PMAC2 can support 256 motion programs. In any time, any where, any one of these programs can be executed even though the same one is being executed in another coordinate. The programs number that PMAC2 can execute in the same time is as many (no more 8) as the number of defined coordinates in the card. A motion program may take any one other motion program as its sub program no matter whether or not the variables are carried.

Indeed, PMAC possesses powerful function, through programming, motion and compensation of following motion modes can be realized such as linear mixture motion, touching off motion of motion programs, arc interpolation motion, PVT (position-velocity-time) mode motion, spline motion, synchronizing motion, tool radius compensation, axis transition matrix etc [7].

PMAC contains 64 non-synchronizing PLC programs (There are 32 compiler languages and 32 non-compiler languages), they, which can repeatedly run with very quickly speed, have many same function with hardware PLC. PLC programs have many same logical structures as the motion programs but, they have not sentences of motion type. PLC programs apply in some operations which are not synchronizing with motion, such as monitoring analog signal input and digital signal input, setting output, sending off messages, monitoring motion parameters. Because PLC programs may completely access PMAC2 variables and I/O, as well as its non-synchronization, PLC becomes a powerful accessory.

Table 1 Axis Address

A4	A3	Axis address
0	1	X axis
1	0	Y axis
1	1	Z axis

Table 2 Each Axis Command Buffer and I/O Adress

A2	A1	buffer	address
1	1	come	Command address
1	0	otpw	I/O address
0	1	Bufw0	Motion parameter low 16bits
9	0	Bufw1	Motion parameter high 16bits

Through bus, ARM processor write commands into motion controller, and make it finish motion control. Motion controller buffer address bus possess 4bit, whose interface address is showed in table1 and Table 2.

Instruction controlling process includes three kinds of modes. The first mode is to writing lower 16bit motion parameter into buffer parameter address; the second mode is to write higher 16bit motion parameter into buffer parameter address. The third mode is to write commands into buffer command address. After writing command in command address, The chip write motion parameters into corresponding registers (such as object position register, initial velocity register, motion velocity register etc) according to commands(that is register addresses). Each register need to be written one time as above. The last mode is to write starts motion command. After all motion parameters have been written, start command is written into command address, then chip begin generating pulse. The process of reading motion parameters include as follows:

- Writing address of motion parameters addresses;
- Reading lower 16bits of motion parameters;
- Reading higher 16bits of motion parameters.

4. Performance Testing and Results

A test had been carried out to verify ENC system performance. After ENC system was operated, an interpolation program was run; meanwhile, the second interpolation program was downloaded from the network. We observed that the first program continued to run steadily to the end when the second one was been downloaded into ARM9 and then into the BUFFER of PMAC. Simulation result is showed in Figure 5.

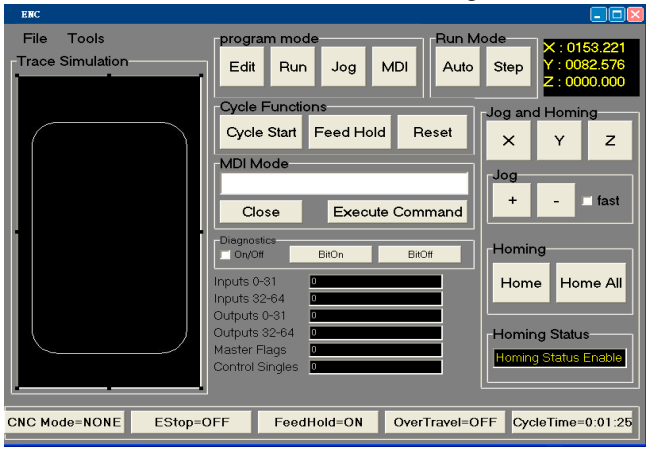

Figure 5. The main interface of the application

5. Discuss and Conclusion

Primary factors that effect performance of the system are summarized as following:

5.1. The continuity of control command

The situation that commands buffer is empty should be avoided. On the basis of buffer, PMAC2 may preprocess multiple traces. Buffer mechanism may guarantee that big undulation of the speed and acceleration of each segment trace do not appear, therefore, machining quality can be enhanced. The key that solve this question is effectively monitoring what happen to buffer, and rationally maneuver motion instructions.

5.2. Servo zero drift phenomenon

Following fact is discovered in system operating that general errorness is 13~15 pulses if zero drift elimination function provided by PMAC2 is not used. Machining quality can not therefore be guaranteed. Zero drift may be eliminated by timely measuring and dynamic modifying zero drift value.

5.3. Adjusting PID parameter

PID algorithm is rather matured and widely applied, but adjusting parameters need abundant practicing experiences. The system, if its parameters are well adjusted, displays excellent static-state characteristics, high machining precision and low running noise. In practice application, preferable PID parameters may be measured by test program, and be modified by system configuration program. A developing direction in parameter configuration is automatically online measured and adjusted. It is very practical for using system in different machine tools and work condition; it is expected to finish in this system as well.

As a powerful motion controller, PMAC2, which possess excellent 8-axis control and interpolation functions, plays an important role in ENC system. whereas ARM9, which has 32bits data width, with RISC instruction set, small bulk and lower power consumption, is a high-performance processor. With motion control card and ARM, we can largely reduce researching and developing time and improve R&D speed. So we can obtain high-performance CNC system in a short time. Owing to the network module adopted in the ENC system, the speed of information transportation is faster than ever, and in addition, the system reliability is improved. Meanwhile we added RTOS μ_c Linux into ENC system,

therefore, ENC system carries on multi-mission processing with high performance, and guarantees the real time property in motion control. It will replace traditional CNC system in future.

Acknowledgment

This work has been supported by National Natural Science Foundation of China (50175037) and Science and Technology Department of Zhejiang Province (2006c21127).

References

[1] E. L. J. Bohez, N. T. H. Minh, B. Kiatsrithanakorn, P. Natasukon, H Ruei-Yun, L.T. Son, "The tencil buffer sweep plane algorithm for 5-axis CNC tool path verification," *Compute-Aided Design,* vol. 35, no. 12, pp. 1129–42, 2003

[2] B. Lauwers, P. Dejonghe, J. P. Kruth, "Optimal and collision free tool posture in five-axis machining through the tight integration of tool path generation and machine simulation," *Compute-Aided Design*, vol. 35, no. 5, pp. 421–32, 2003.

[3] D. Sun, J. K. Mills, "Adaptive synchronized control for coordination of multi robot assembly tasks," *IEEE Transactions on Robotics and Automation,* vol. 18, pp. 498–509, 2002

[4] F. W. Mi, G. F. Chen, "The Application of Electromechanical Integrated Experimental Device," *machine tool & hydraulic pressure,* no. 12, pp. 129～131, 2004. (in chinese).

[5] S. Jin, J. Z. Fu, Z. C. Chen, The Design of Control System Based on ARM and High speed Pulse Generator, " *Mechanical & Electrical Engineering Magazine,* ," vol.22, no. 2, pp. 56~58, 2005. (in chinese).

[6] The delta tau.com homepage (2006, February 10) [Online]. Available: http://www.deltatau.com

[7] J. P. Wang, A. Wang, Z. L. Jing et al, "Design and Application of an Anti-Windup State Feedback Controller," *Journal of Xi'an Jiaotong University,* vol. 36, no. 4, pp. 365~368, 2002.