International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015)pp34867-34872

Automation of a CNC Lathe Using Open Architecture

Aji Joy[1], B. Vinod[2]

1. Assistant ProfessoratDepartment of E.C.E, Mar Athanasius College of Engineering, Kothamangalam, Kerala. Research scholar at PSG College of Technology, Coimbatore, Tamil Nadu.
2. Associate Professor at Department of E.E.E, PSG College of Technology, Coimbatore, Tamil Nadu.

Abstract
In this paper a system is proposed to retrofit a two axes Computer Numeric Control machine with open software and hardware architecture. The main objective of the paper is to analyses the performance of an open controller and the corresponding hardware. The overall control is done by open software developed in Linux. The machine operation is tested by sending a sample data written in G-code to a two axis CNC lathe, retrofitted with open hardware and interfacing to LinuxCNC operating system. The machine input commands issued by the LinuxCNC are interpreted by the motion control card and trajectory control signals are generated.These signals are converted to motor drive pulses by an FPGA module. The I/O signal generated by the FPGA module is fed to the AC servo motors using servo drives. The motion control card communicates with the LinuxCNC based personal computer using PCIExpress bus. That will assist the machine tool designers to develop a more flexible and versatile open system and help the end users to reconfigure and restructure the system as per their own requirement at lower cost.

Keywords: Open Architecture, CNC Lathe, LinuxCNC, Motion Control Cards, Servo Drives, Servo Motors, Open Architecture Control.

1. INTRODUCTION
Since the launch of first Numeric Control (NC) machine by MIT in 1952, Computer Numeric Control (CNC) technology has gone through leaps and bounds in different areas[1]. Most of the advancements were limited in usage because it was employed in proprietary fields. During the last three decades there are significant changes in CNC by the introduction of open source architecture. Due to the globalization of the economy, high competition in the market, the demand for the new technology innovation and need of short time to market, CNC also demanded the flexibility offered by the open architecture.

In the traditional NC systems, application programs, system software and the hardware were tightly coupled. This led to inflexibility, complexity and long development time for such systems. Due to the high innovation speed in the processing and the communication technologies during past two decades, it become necessary for companies to develop hardware independent software to stay competitive. In order to cop-up with the market requirements, frequent updates are needed in the motion control also. To accomplish this Open Architecture (OA) is the only solution.

Open architecture control (OAC) is a well-known terminology in the field of machine control. In a similar fashionto the manner in which openness has revolutionised the PCindustry, it also has the potential to revolutionise the CNC industry[1]. An open the system should have the features like modularity, portability, interoperability, extensibility and scalability. It should be vendor free, consensus driven, standardised, freely available, user customizable and upgradable. The above features will enable the system builders and end users to operate on a more versatile machines[2].

Although the research work of CNC based on Linux is still not as active as the equivalent Windows-based research [3, 12, 13], the software-oriented CNC base on Linux/ RTLinux is an open source software (OSS) solution, which has many advantages that Windows system cannot, and willnot, offer. OSS is becoming a valuable part of themanufacturing system in which the users,integrators, and even developers can benefit through theflexibility of decision making, including which features toinclude, what extensions to make, and when to implementthem. The large quantity of software offered by theworldwide open source communities avoids the potentialfor repetition of research work [3].

The organization of this paper is this. Section 2 explains about Open Architecture Control (OAC) and an overview of LinuxCNC is given in section 3. In section 4 the existing system is explained. The section 5 is the proposed system and section 6 is the results and limitations.

2. OPEN ARCHITECTURE CONTROL
The concept of OA was introduced in the early 80's in order to incorporate customer specific requirement based on market demand and cost. Many research facilities, universities and consortia worked their way into it and different systems were developed world wide. Unfortunately there were no mutual compatibility among these systems. National Institute of Standards and Technology (NIST) in USA developed measurements and standards for intelligent control systems relevant to manufacturing industries[4].

The NIST had foreseen the future and they had developed the Real-Time Control System (RCS) as a standard reference model for building real-time intelligent control systems[5]. RCS evolved over many years and it led to the development of numerous controller applications. Enhanced Machine Controller (EMC) was one of the prominent among them. EMC offers real-time control based on open source and

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015)pp34867-34872

community software development. It is suitable for a variety of equipment such as machine tools, robots, co-ordinate measuring machines, etc. [6].

The EMC public domain software system was originally developed by NIST for controlling milling machines using computer systems. It was based on RS-274D (G-Codes) and is considered as the reference design for the implementation of OA for CNC. In June 2000 NIST relocated the source code to sourceforge.net for public access. This enabled the public to make additions and modifications to the EMC system and it was renamed as EMC2. The licensing of EMC2 is placed under GNU General Public License (GPL)[6]. Further a new layer known as Hardware Abstraction Layer (HAL) was introduced to EMC2. This was for interconnecting different functions easily without altering C-code or recompiling. It also includes a virtual oscilloscope to examine the operation in real time. Finally EMC2 evolved as LinuxCNC[6]. It uses Linux kernel with real time extensions (RTAI) for the control operations.

LinuxCNC can offer flexibility and openness along with various intelligent manufacturing functions and they can be incorporated very easily[6]. It has additional features like faster design cycles, low down time, increased productivity, low maintenance time, increased processor speed, better user interface, etc.

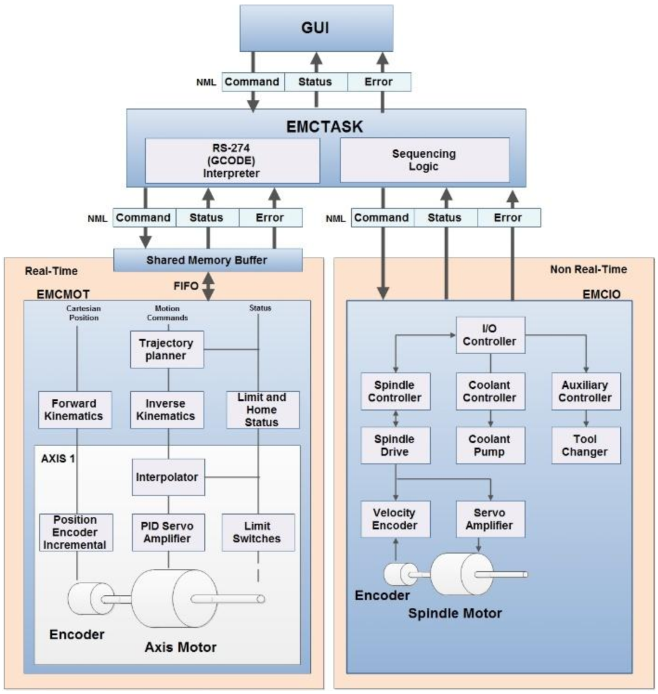

Figure 1. LinuxCNC Architecture.

3. LinuxCNC OVERVIEW

LinuxCNC is an open source software under GPL for CNC machines. It can control up to 9 axes in a CNC machine. It uses the real time kernel of the Linux system and supports both servo and stepper motors. Drivers for LinuxCNC is available from various vendors. Since it use HAL, developers can mix and match various control boards and output control signals thorough parallel and serial ports[7].

Many enhancements were incorporated by NIST in the advanced versions of the LinuxCNC. The user can select Graphical User Interface (GUI) according to his preference for machine control. The tool management helps to interface tool changes. Alternative part programs also can be used with LinuxCNC. Machine inspection can be done online by checking the status of different system parameters. Provision is also provided for thermal geometric error compensation[8,9,10].

The general block diagram of LinuxCNC architecture is shown in Figure 1. The major blocks in the architecture of LinuxCNC are motion controller, discrete I/O controller, task executor and user interfaces. The Motion Control Card (MCC) receives commands from user space modules via a shared memory buffer, and executes those commands in real-time. The status of the controller is made available to the user space modules through the same shared memory area. The motion controller interacts with the motors and other hardware using the HAL[10].

3.1. Motion Controller

The motion controller used in LinuxCNC is EMC Motion Controller (EMCMOT). It executes the executable modules in a loop to perform real time operations and performs trajectory planning, direct and inverse kinematic calculations and computation of a desired output to the motor control subsystem. This process includes sensing of controlled axis positions, computation of the next trajectory point and interpolation between these trajectory points. It supports both hardware limits and programmable software limits. The hardware limits will be sensed by the axes limit and home switches where software limits are assigned as numeric values in the codes[10].

3.2. Discrete I/O Controller

EMC I/O (EMCIO) Controller is the Discrete I/O controller module that handles all I/O functions which are not directly related to the actual motion of machine axis. It is implemented as an I/O controller consisting of a hierarchy of subordinate controllers. It controls the main spindle, coolant pump, aids in automatic tool change and auxiliary functions like emergency stop, lubrication, etc. [10].

3.3. Task Executer

The task executer (EMCTASK) coordinates EMCMOT, EMCIO and the user interface. It includes two modules - the G-code interpreter and sequencing logic. It receives and analyzes the commands, either from the operator through GUI or from another process. The commands are interpreted into Neutral Messaging Language (NML) messages and dispatches them to EMCMOT, EMCIO or EMCTASK at appropriate times [10].

3.4. User Interfaces

AXIS is the standard GUI interface that comes along with LinuxCNC. It can be added with virtual control panels like PyVCP(Python based Virtual Control Panel) and GladeVCP(Glade based Virtual Control Panel) for better control operations [10].

4. EXISTING SYSTEM

The existing SBL110CNC lathe is equipped with DC servo drives and motors to control the vertical and horizontal axes. The drive shafts are connected to the motors using belt and pulley mechanism. DC servo motors are equipped with built-in rotary encoders for feeding back the position information of axes to the PLC. An induction motor is used to drive the spindle and for sensing the speed of the spindle an external encoder is used. The specification of the axes details are given in the Table I.

Table 1. Specification of the axes.

Sl No	Parameter	Values
1	Feed Range	1-2000 mm/min
2	Rapid Traverse rate	5 mm/min
3	Max. threading pitch	2 mm
4	Positioning Accuracy	± 0.01 mm
5	Repeatability	± 0.05 mm
6	Spindle Speed	1000-3000 rpm

In SBL110CNC the control signals for driving the motors are generated from a PLC. The PLC is interfaced to the motor drives and different I/Os using daughter boards. The daughter boards contain opto-coupler circuits and relays for electrical isolation. Also the Human Machine Interface (HMI) is equipped with a keypad and CRT monitor.

In the present scenario, due to the advancement in technology, improvement in the performance of motion control can be achieved. The parameters like resolution of axes movement, spindle speed and torque and electrical isolation can be improved. Further enhancements such as flexibility in control, reconfigurability and better HMI can be achieved by automation of the existing machine. This will also reduce the power consumption and improve the overall efficiency of the machine. Above all, as the hardware and software used are not based on open technology, it is not possible to reconfigure, repair or replace the system as individual modules [11-18].

5. PROPOSED SYSTEM FOR RETROFIT

In the proposed system effort is taken to retrofit an old two axes SBL110CNC lathe with OA. This will help to overcome the limitations of the existing machine and will provide better machine control operation. Compactness and accurate control is achieved by replacing the PLC with MCC. Resolution and torque are improved by replacing the DC servo motors and induction motor by AC servo motors. By using PC instead of HMI, the flexibility in user interface and configurability of the system are enhanced.

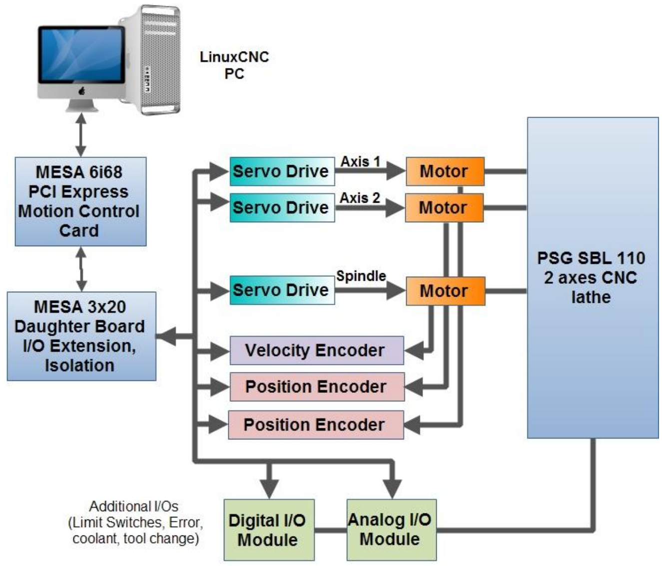

Figure2. Block Diagram of the Proposed System

5.1. Motion Control Card and Daughter Board

The MCC selected for the proposed system is MESA6i68 PCI Express Card. The main reason for the selection of this particular card is its compatibility with the PCI Express bus. The MESA6i68 Card works with the system in real time and provide flexibility for better configurability. The daughter board selected is MESA3x20, which is compatible to the MESA6i68 MCC.

The input from the HMI are decoded into machine control signals by the MESA 6i68 PCI Express Card and send as I/O signals to the drives with proper isolation. This will be done by a daughter board MESA3x20, which operates with Spartan-3 FPGA[19]. The MESA3x20 is a high I/O density external FPGA card with 144 user I/O ports. It requires only 3.3V external power supply. PCI bridge allows FPGA programming via the host and does not require any FPGA bootstrap. It supports a host data transfer rate up to 150MB/Sec [19].

In the proposed system, due to the use of microcontroller and FPGA instead of PLC, the problems like limitation of available I/Os, restricted reconfigurability, high power consumption, etc. are eliminated.

5.2. Drives and Motors

The selection of feed motors and drives are very important because it determines the precise axes movement. In the proposed system AC servo drives and motors with built-in encoders are used.

The drive used in the system is Sanyo Denki RS2A03A0AL0. Depending upon the input signal and the signal fed back from the encoder, speed and direction of the motor rotation is controlled. The drive itself is provided with a digital panel with which the operator can set different parameters for the configuration of drive. It is also possible to perform test operations such as JOG with different parameters like speed and direction of rotation, the torque produced by the motor at different speeds, etc. [20]. The drive's maximum capacity is 1000W and can deliver pulses at the maximum rate of 5Mpps.

Out of the two motors used to control the axes movement, the z-axis motor does not require a brake while the x-axis motor requires. The z-axis is uses Sanyo Denki AC Servo Motor R2AA08075FXH00W and the x-axis uses Sanyo Denki AC Servo Motor R2AA08075FCH00. The maximum rated output delivered by both servo motors is 750W. They have rated maximum torque of 2.39 Nm and rated maximum speed of 6000 rpm. A serial incremental encoder is used and the number of partitions per rotation is 1,31,072 (17 bit) [20].
In the proposed system, dynamic configuration of axes position and spindle velocity is possible by varying different parameters. This enables the use of same drive for axes movement and spindle rotation. Also, machine operations for different load conditions are effectively carried out since the PID values for different load conditions can be assigned. Based on the type of feedback, the encoder can be configured for incremental and absolute values.

6. RESULTS AND DISCUSSION

SBL110CNC machine is retrofitted with open controller and the performance is evaluated by analysing various parameters. The DC Servo system and the induction motor are replaced with AC servo system for better and precise control. For flexibility the controller is replaced with an open controller. The HMI is replaced with a PC to make the system more user friendly. The retrofitted machine is shown in the figure. 4.

Figure 3.The SBL110 CNC machine retrofitted with open controller.

The simulation result of the CNC operation resulted from the machining of the tool shown in the figure. 5 is given below in figure. 6. The stroke shown in blue colour is the homing of the axes. The Rapid traverse shown in purple and the tool path is shown in black. The finished product is shown in the figure.5.
Improvement is shown in machining operation since the feed rate and rapid traverse rate and shown much improvement. Since the ball screw used is same there is no difference in maximum threading pitch. Machining errors like positioning accuracy, repeatability and backlash are minimised in the new system. The parameters like spindle speed and axes holding torque are shown good improvement.Many functionalities like temperature compensation, flexible JOG operation, file input from CAD software are added. Lot of improvements are made in functionalities by fully automating coolant pump, lubricant pump and tool changer. Real time performance is also improved much since the LinuxCNC is capable of handling real time tasks.

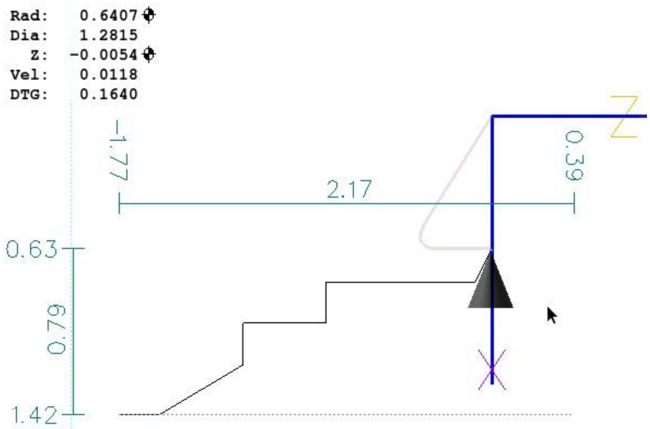

Figure4. LinuxCNCsimulator(AXIS) screen shot.

The comparison of the features of old system with the retrofitted system are given in the Table. 2.

Table 2.Specification of the axes.

Sl No	Parameter	Existing System	Retrofitted System
1	Feed Rate	1-2000 mm/min	1-5000 mm/min
2	Rapid Traverse rate	5 mm/min	10 mm/min
3	Max. threading pitch	2 mm	2 mm
4	PositioningAccuracy	± 0.01 mm	± 0.04 mm
5	Repeatability	± 0.05 mm	± 0.09 mm
6	Backlash(vertical)	± 0.05 mm	±0.02 mm
7	Backlash(horizontal)	± 0.03 mm	±0.01 mm
8	Spindle Speed	1000-3000 rpm	0-6000 rpm
9	Holding torque of axes	1.0 Nm	2.3 Nm
9	Temperature Compensation	Nil	Available
10	Tool changer	Manual	Automatic
11	Coolant pump	Manual	Automatic
12	Lubricant pump	Manual	Automatic
13	JOG operations	Available	More flexibility
14	Loading Characteristics	Low torque	More torque
15	CAD, CAM, G codes	Only G codes	All formats possible in PC
16	Real time tasks	Not always	Executed in real time

9. CONCLUSION

The SBL110 CNC machine is successfully retrofitted with LinuxCNC and open hardware. Machine tool operations are performed as per the expectations of the evaluator. Even though the performance was satisfactory many interfaces were untested due to the absence of support from the LinuxCNC or due to the absence of driver for the hardware. Even though the system operation is done with two axes machine, the operation can be extended to multi axis platform.

The LinuxCNC satisfies the educational and light operations level machining operations. Due to the robust open source real time OS, LinuxCNC promotes expansion possibilities, scalability, interoperability and reconfigurability of open characteristics.

One of the main disadvantage noticed was it lack interface to the motion controller through Ethernet. Due to the compatibility issues in modules developed in two different real time environment like RTAI and RTLinux the Ethernet support is not possible in real time operation. But it is found that many proprietary controllers support Ethernet interface.

The auxiliary functions like coolant pump, lubricant, tool changer, emergency stop, are also supported by LinuxCNC system with standard functions or auxiliary functions.

REFERENCES

[1] Ambra C et al, 2002, “The design of a high performance modular CNC system architecture,”In: Proceedings of the 2002 IEEE international symposium on intelligent control, British Columbia, Canada, 27–30 October 2002, pp 290–296.

[2] O. L. Asato, E. R. R. Kato, R. Y. Inamasu, A. J. V. Porto, 1999, “Analysis of Open CNC Architecture for Machine Tools,” 15th Brazilian Congress of Mechanical Engineering ~ COBEM 99~, November 22-26, 1999, São Paulo. SP. Brazil.

[3] Pritschow G et al, 2001,“Open controller architecture-past, presentand future,” Ann CIRP 50(2):pp463–470.

[4] Fred Proctor, John Michaloski, 1993, “Enhanced Machine Controller Architecture Overview,” National Institute of Standards and Technology, US, 1993.

[5] EMC Team, 2011, “EMC v2.4 User manual,” http://www.linuxcnc.org, 2011.

[6] Reuven Katz, Byung-Kwon Min, ZbigniewPasek, 2000, “Open Architecture Control Technology Trends,” ERC/RMS Report 35, September 2000.

[7] Zhang CR et al, 2003, “An USB-based software CNC system,” JKMater Process Technol 139, pp286–290.

[8] Park S, Kim S-H, Cho H, 2006,“Kernel software for efficientlybuilding, re-configuring, and distributing an open CNC controller”,Int J Advanced ManufTechnol 27 pp788–796.

[9] Ferraz F, Coelho RT, 2005, “Data acquisition and monitoring inmachine tools with CNC of open architecture using internet”, Int JAdvManufTechnol 26,pp90–97.

[10] Bartos FJ, 2004,“Is Linux at the gates of the factory?”, Control Eng51(5), pp32–36.

[11] Dennis B, 2004,“Open up with open source”, Control Eng 51(5), pp56.

[12] MA Xiong-bo, HAN Zhen-yu, WANG Yong-zhang, FU Hong-ya, 2006, “Development of a PC-based Open Architecture Software-CNC System,” Elsevier, Chinese Journal of Aeronautics, November 2006, pp. 272-281.

[13] Suhua Xiao, Di Li, Yizong Lai, Jiafu Wan, and Shouting Feng, 2007, “An Open Architecture Numerical Control System Based on Windows CE,” IEEE International Conference on CONTROL AND AUTOMATION, May 30 to June 1, 2007, Guangzhou, CHINA.

[14] TomislavStaroveški, DankoBrezak, TomaUdiljak, DubravkoMajetić, 2009, “Implementation Of A Linux-Based CNC Open Control System,” 12th International Scientific Conference On PRODUCTION ENGINEERING~CIM 2009~, June, 17-20, 2009 Biograd, Croatia.

[15] Hua Ji& Yan Li &Jian Wang,2008, ” A software oriented CNC system based on Linux/RTLinux,” The International Journal of Advanced Manufacturing Technology, Volume 39, Issue 3-4, October 2008,pp 291-301.

[16] Chi Yonglin,2005, ”An evaluation space for open architecture controllers,” The International Journal of Advanced Manufacturing Technology,” Volume 26, Issue 4, August 2005,pp 351-358.

[17] DraganMilutinovic. Milos Glavonjic, Nikola Slavkovic, ZoranDimic, SasaZivanovic, BrankoKokotovic, LjubodragTanovic, 2011, ”Reconfigurable robotic machining system controlled and programmed in a machine tool manner,” The International Journal of Advanced Manufacturing Technology, Volume 53, Issue 9-12, April 2011,pp 1217-1229.

[18] Kung-Lin Ho, Chiu-Keng Lai, 2005,“PC-based Real-time Motion Controller Design using the FPGA Chip,” Proceedings of 2005 CACS Automatic Control Conference Tainan, Nov 18-19, 2005, Taiwan.

[19] Technical Specification of MESA cards available from http://www.mesanet.com.

[20] Technical specifications of Sanyo-Denki drives and motors available from www.sanyo-denki.com.

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 14 (2015)pp34867-34872

Aji Joy received B. E. Degree in Electronics and Communication Engineering from Mangalore University, Karnataka, India in the year 1996 and M. E. in Power Electronics and Drives from PSG College of Technology, Coimbatore, Tamil Nadu, India in 2008.His major interest is in embedded systems, and automation and the research area is Design and Development of Open CNC controllers.

B. Vinod received B. Tech. Degree in Electrical and Electronics Engineering from Calicut University, Kerala, India in 1991, Post-Graduation in Applied Electronics from Bharathiyar University, Coimbatore, India in 1997 and doctorate from Bharathiyar University, Coimbatore, India in 2007.His major fields of interest are Automation, CNC controllers and robotics.