To be published at the 5th International Symposium on Robotics and Manufacturing, Maui, HI, August 14-18, 1994.

# The Enhanced Machine Controller Architecture

**Dr. Ronald Lumia**
*National Institute of Standards and Technology*

**Abstract**

Nearly half of the world's machine tools were manufactured in the US in 1980. Today, the US market share has fallen to 10%. Domestic machine tool manufacturers are no longer competing effectively with their European and Asian counterparts. NIST is meeting this challenge by developing the Enhanced Machine Controller (EMC). The EMC is being developed to test generic interfaces and demonstrate the benefits of an open architecture for machine tools. The EMC will demonstrate how to reduce life cycle costs of controllers, improve performance and accelerate the incremental integration of new technology. This paper will describe the EMC architecture and the plans for its implementation.

## 1. Introduction

It has long been recognized that system architecture plays a major role in systems integration. The research community, for example, is very diverse in its approach to general architectures. Saridis was the first to suggest the use of intelligent control [1] while Brooks took quite a different approach with the subsumption architecture [2]. More recently, Miller has taken a more pragmatic approach in developing robot control systems [3]. Architectures have also been developed for specific domains. In the area of vehicle control, Lowrie describes the Autonomous Land Vehicle project [4] and Shafer reports on a mobile robot project [5]. Robot architectures include the work by Khatib [6] and Khosla [7].

The role of architecture in systems has been a focus within NIST since the mid 1970s when Barbera implemented the first version of the Real-time Control System (RCS) [8]. A slightly modified version of this architecture was using in the early 1980s as the basis for the Automated Manufacturing Research Facility (AMRF) [9]. Since then, RCS has been used as the basis for many projects including the Army Field Materiel Handling (FMR) Robot [10], the Multiple Autonomous Undersea Vehicle (MAUV) project [11], the Flight Telerobotic Servicer [12], a coal mining project [13], a remotely driven vehicle [14], and others.

Architecture has recently played a role in the development of machine tool controllers. Martin Marietta produced the Specification for an Open System Architecture Standard (SOSAS) for the Next Generation Controller project [15]. Recently, several commercial controller vendors have adopted the open architecture approach.

The problems associated with machine tool controllers can be grouped into two classes, excessive cost and lack of openness. High cost is manifested through initial insertion cost of the technology, training, and maintenance, as well as the lack of volume to justify investment. Lack of openness prevents the sophisticated user from modifying the control system to suit the application.

## 2. Project Goal

Given the two underlying problems of excessive cost and lack of system openness, NIST has embarked upon the development of the Enhanced Machine Controller. The goals of the project are to:

- Reduce life-cycle cost to users of machine tool controllers by providing public domain open system software.
- Demonstrate the open system on machines for two user groups: retrofit and high-end.

The EMC will initially focus on controllers for metal removal machines such as mills, lathes, grinding machines, and lasers. However, there is fundamentally little difference between the EMC and other controllers, such as robot controllers, and eventually the EMC will be used in other domains.

## 3. Technical Approach

NIST will develop the EMC and demonstrate how machine tool builders, integrators and users can:

- Construct a baseline controller using an initial set of components, which are described in the next section
- Extend the initial set of components with new ones
- Modify the controller to handle new applications or integrate new technology
- Access low-level functions and data
- Improve machine diagnostics
- Communicate with other factory systems

The Host Machine Executive (HME) provides the basic machine tool controller functionality. The HME can be any combination of software and hardware, depending on budget and performance desired. The ability to scale the design will be demonstrated through two types of implementations. One is aimed at the retrofit community where the goal is to put a new controller on an old machine, and thereby enhance its capabilities. The second implementation is aimed at the high-end users who need to integrate controllers onto new machines for challenging applications. The HME configuration toolkit will be created late in the project to automate essential administrative and configuration operations.

Throughout the course of the project, NIST intends to use a consortium of machine tool builders, machine tool users, and controller builders for advice and project direction.

## 4. Architecture Description

As shown in Figure 1, the HME forms the central part of the controller and performs all of the basic functions. An Application Programming Interface (API) is developed for each of the basic functions. The API represents the most generic interface between the system kernel and the basic function so that functionally equivalent modules can be configured into a system. For example, one of several possible trajectory generators may be chosen for a specific controller depending on the accuracy desired. The HME includes both the APIs and an instantiation of each basic function.The user can keep the default instantiation of the function or substitute another if so desired. Furthermore, modules may be implemented using any combination of software or hardware, so long as they satisfy the API.

Not all of the basic functions of the controller need to operate with the same cycle time. Figure 2 illustrates a hierarchy which shows the relationship between the rates required for the basic components. Each level in the hierarchy represents roughly an order of magnitude difference in the required execution rate of a module. The Workstation Planning and Workstation Management modules do not typically reside within the machine tool controller. They simply illustrate the relationship between the machine tool and the factory in which it works. The Plan interpreter accepts a part plan, e.g., RS-274D, BCL, NCL, etc., and coordinates the continuous and discrete activities. Following the continuous leg, the output from plan interpretation produces commands for the trajectory generation module, e.g., G01, G02, etc., which are sent to servo control where any number of algorithms could be executed, e.g., 3rd order subinterpolation for a PID loop. On the PLC side, the discrete input/output module decomposes input commands in a similar fashion and eventually sends commands through the discrete equivalent of a servo control for actions such as fluid on/off, tool changes, spindle on/off, etc. The operator interface allows human interaction at any level of the hierarchy. One can modify the execution of the part plan, such as changing the feed rate. However, it is also possible to interact with the hierarchy at a much higher level, selecting the next part program for the machine tool, for example.

## 5. Conclusion

The expected benefits include reducing the life-cycle cost of controller implementations, improving the competitiveness of the U.S. machine tool industry, creating a market for controller subsystem entrepreneurs, and developing controller specifications that can be used for DoD procurement actions.

System architecture has been shown repeatedly at NIST and elsewhere to be a critical element in all systems, including machine tool controllers. The EMC embodies the experience gathered from 15 years of research at NIST as well as the work of many other projects, such as the Next Generation Controller project. This paper has described the EMC architecture, a plan to implement the architecture, and the benefits derived from using it as the basis of machine tool controller.

## References

[1] Saridis, G.N., Foundations of the Theory of Intelligent Controls," *IEEE Workshop on Intelligent Control,* Troy, NY, August 1985.

[2] Brooks, R.A., "A Robust Layered Control System for a Mobile Robot," *IEEE Journal of Robotics and Automation*, Vol. RA2, no. 3, March 1986.

[3] Miller, D.J., Lennox, R.C., "An Object-Oriented Environment for Robot System Architectures," *IEEE International Conference on Robotics and Automation*, Cincinnati, OH, May 1990.

[4] Lowrie, J., et. al., "Autonomous Land Vehicle," *ETL-0412*, Martin Marietta Denver Aerospace, Denver, CO, July 1986.

[5] Shafer, S.A., et. al., "An Architecture for Sensor Fusion in a Mobile Robot," *IEEE International Conference on Robotics and Automation*, San Francisco, CA, April, 1986.

[6] Khatib, O., "A Unified Approach for Motion and Force Control of Robot Manipulators": The Operational Space Formulation," *IEEE Journal of Robotics and Automation*, Vol. RA3, no. 1, February, 1987.

[7] Stewart, D., Schmitz, D., and Khosla, P. K., The CHIMERA II Real-Time Operating System for Advanced Sensor-Based Robotic Applications, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol 22, No. 4, July, 1992.

[8] Barbera, A.J., Albus, J.S., and Fitzgerald, M.L., "Hierarchical Control of Robots Using Microcomputers," Proceedings of the *Ninth Symposium on Industrial Robots*, Washington, DC, March, 1979.

[9] Simpson, J.A., Hocken, R.J., Albus, J.S., "The Automated Manufacturing Research Facility of the National Bureau of Standards," *Journal of Manufacturing Systems*, Vol. 1, no. 1, 1982.

[10] McCain, H.G., Kilmer, R.D., Szabo, S., Abrishamian, A., "A Hierarchically Controlled Autonomous Robot for Heavy Payload Military Field Applications," *International Conference on Intelligent Autonomous Systems*, Amsterdam, The Netherlands, December, 1986.

[11] Albus, J.S., "System Description and Design Architecture for Multiple Autonomous Undersea Vehicles," *NIST Technical Note 1251*, NIST, Gaithersburg, MD, September, 1988.

[12] Albus, J.S., McCain, H.G., Lumia, R., "NASA/NBS Standard Reference Model Telerobot Control System Architecture (NASREM)," *NIST Tech. Note 1235*, NIST, Gaithersburg, MD, July, 1987.

[13] Huang, H.M., "Hierarchical Real-Time Control System for Use with Coal Mining Automation," *Fourth Conference on the Use of Computers in the Coal Industry*, Morgantown, WV, June 1990.

[14] Szabo, S., Scott, H.A., Murphy, K.N., Legowik, S.A., "Control System Architecture for a Remotely Operated Unmanned Land Vehicle," Proceedings of the *5th IEEE International Symposium on Intelligent Control*, Philadelphia, PA, September, 1990.

[15] Martin Marietta Corporation, *Next Generation Workstation/Machine Controller Specification for an Open System Architecture Standard* (Draft), Volumes I through VI, Document No. NGC–0001–13–000–SYS, March 1992.

## Acknowledgments

This work was supported by the Navy Mantech Program through the NIST Automated Manufacturing Research Facility. The author is grateful for the competent and dedicated staff working on the EMC project including John Michaloski, Fred Proctor, Simon Frechette, Brad Damazo, Bob Gavin, and Charles Yang.

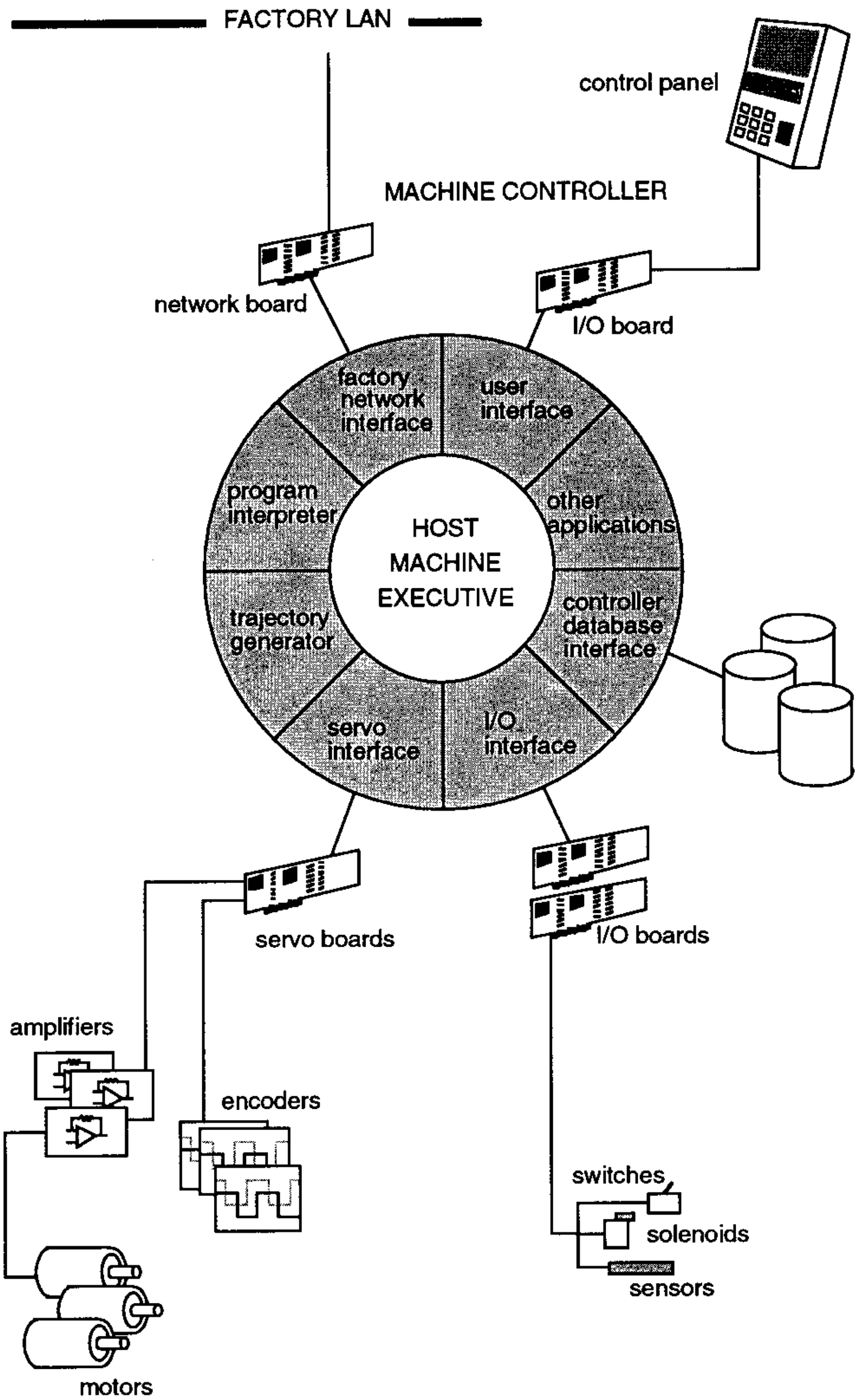


**Figure 1. Host Machine Executive controller**

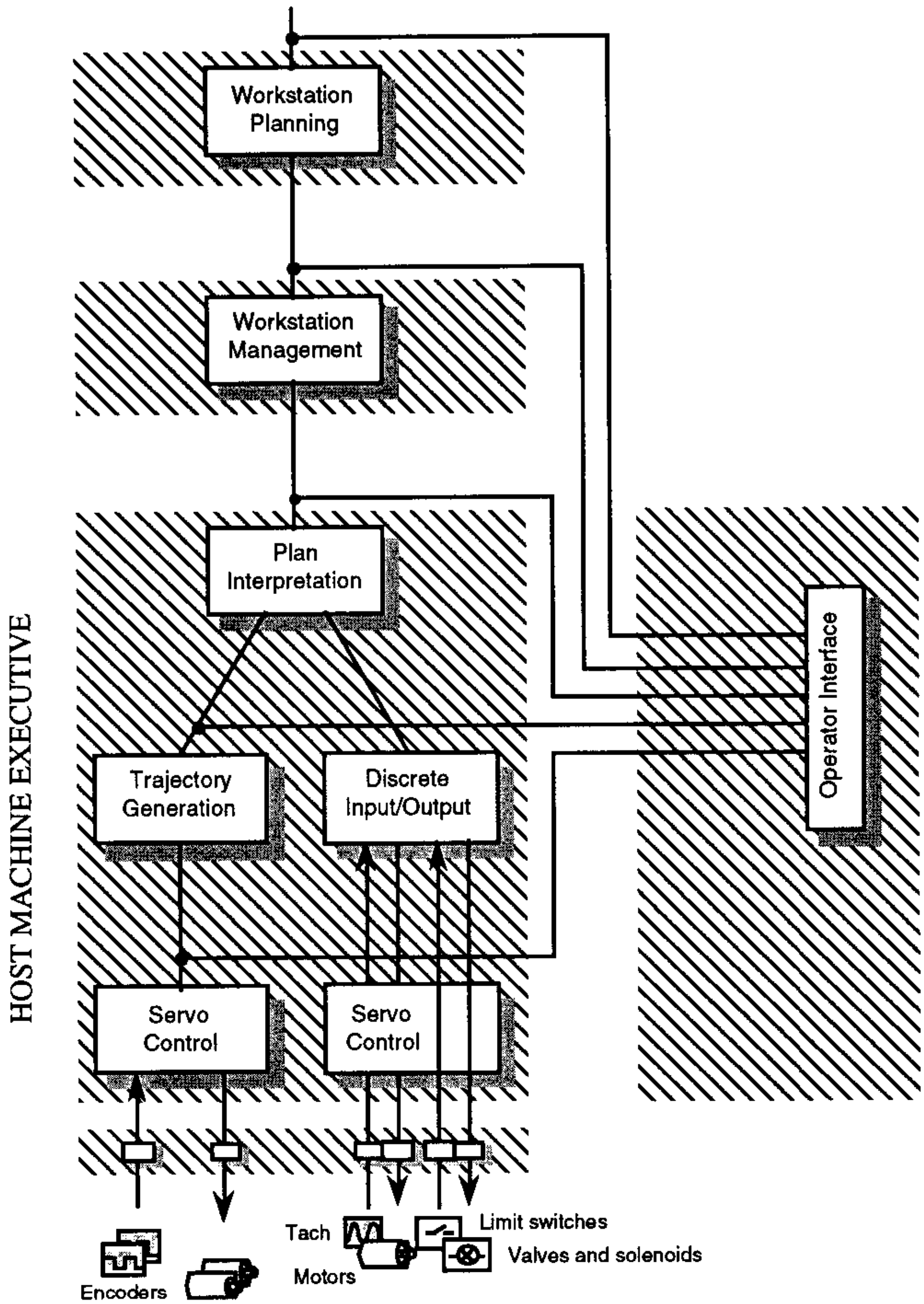


**Figure 2. Baseline EMC hierarchy**