CONTROLS AND SENSORS
FOR ADVANCED MANUFACTURING

Frederick Proctor
National Institute of
Standards and Technology
Gaithersburg, Maryland

ABSTRACT

This paper describes research into improving the performance of machine tools and robots. Techniques for improving accuracy, such as thermal compensation, automatic teaching, and force-based surface and edge finishing, are described. These techniques have been demonstrated to greatly improve the performance of machine tools and robots, but their application has been limited due to the proprietary nature of most industrial robot and machine tool controllers. The paper addresses the issue of proprietary architectures in the light of recent programs to standardize interfaces in the controller industry.

INTRODUCTION

Conventional machine tools and robots are position-controlled devices whose accuracy depends on the quality of their components, performance of the servo control, and the resolution of their position sensors. Component quality is manifested in the straightness of linear axes, orthogonality of successive axes, the eccentricity and backlash of gears, and consistency in the pitch of the screws. Designers can choose to improve the fabrication of these components, or to calibrate their inaccuracies and build compensation into the control of the machine. While improving the manufacturing of components reduces the need for calibration, such improvements are costly. Many vendors find this cost unjustified, and choose instead to calibrate the errors. As an example of this, the reader may be familiar with “lead screw comp,” compensation for inconsistencies in the pitch of lead screws that has been built into many commercial machine tools. Solutions of this type are termed static calibration, since the measurements are performed once for each model of the machine, or for each particular machine if the results vary significantly.

Certain repeatable inaccuracies in machine tools and robots are not static, but vary with the state of the machine. A classic example is thermal growth, in which the device actually expands as it warms up. Other nonrepeatable inaccuracies are introduced due to the unknown geometry of the part within its tolerances, and the errors introduced by operators during fixturing. When the part is made entirely on a single machine, the tolerance problem is reduced, since the resident part program contains the actual choices made within the tolerance band. Errors are more obvious when the part has been machined on another device, whose tolerance choices reside in a program inaccessible to other machines. Machinists must take great care when indicating a part into a machine tool, so that the location of previously machined features can be captured.

In general, every technique for improving accuracy through calibration or *a priori* measurements will be confounded by errors whose values cannot be predicted with certainty. Instead of predicting errors, however, one can measure them directly and gain a much higher confidence in their values. These measurements rely on sensors which capture the error information. For example, a machine tool controller may limit the force on a cutter by limiting the feed rate, based on previous measurements which determined the feed rate which caused excessive force. A margin of safety is built in so that feeds do not approach this upper limit. Unfortunately, this approach does not account for tool wear, which causes the force to rise, and also unnecessarily limits the feed rate for sharp tools. By using a sensor which measures the machining force directly, one may optimize both the feed rate and the cutter lifetime, resulting in increased throughput and reduced tooling cost.

In the next few sections, techniques akin to the one described above are detailed to emphasize the benefits of using sensor-based control to improve quality, shorten cycle times, and reduce cost. Ultimately, the application of these techniques have been limited by the difficulty in incorporating them into today's proprietary controllers. The concluding section examines what is being done to open up the machine tool and robot controller through standards efforts.

THERMAL COMPENSATION

Researchers in the Automated Production Technology Division of the National Institute of Standards and Technology (NIST) have developed a three-level approach to machine tool accuracy enhancement, as part of a Quality in Automation (QIA) program [1, 2, 3]. The foundation of the QIA program is a control architecture which features three sensor-based controllers which operate at decreasing cycle times: a real-time control loop, a process-intermittent control loop which relies on fast probing, and a post-process control loop using dimensional information generated by a coordinate measuring machine. The QIA architecture has been implemented on both a vertical machining center and a turning machine. This section concentrates on the real-time control loop, which modifies machine tool trajectories during machining to effect improved accuracy.

The real-time control loop relies on a geometric-thermal model of the machine tool which estimates the various components of the tool's systematic errors for given positions and temperature gradients. Before this system can be employed, both a geometric thermal model and a kinematic model of the machine tool must be developed. These models are used to determine how the thermally-induced errors in various components will affect the overall error in the tool tip position. The development of these models is analogous to calibration, which is done only once and is used continually thereafter. The machine tool is instrumented with temperature sensors which are monitored by a host computer, which computes tool tip errors based on the geometric-thermal model and the kinematic model. Practical difficulties arise when attempting to feed this information back into the machine tool controller, since most controllers do not provide access to the servo controllers which easily support such feedback.

Typically, three approaches may be used to apply computed error compensation to a machine tool. One is to inject the error compensation signal directly into the servo control hardware as an analog voltage. This method was applied to the real-time error compensation of a Brown and Sharpe[1] vertical machining center at NIST. The technique is suitable for controllers which implement servo control using hardware, but in many systems servo algorithms are computed in software or firmware, and the injection of an analog signal is inappropriate. In these cases, the compensation signals are represented digitally, and are input to the controller via ports and written to registers that are read during the servo computations, such as the registers which contain the following errors. At NIST, this second method has been applied to the enhancement of a Hardinge turning center. Alternatively, one can insert a real-time error corrector (RTEC) between the position feedback element of the axes of the machine tool and the machine tool controller [4]. This device independently counts the unaltered signals from the feedback element, and alters the signals before they are provided to the machine tool controller. The value of the alteration depends upon calculations made in real time based on models of the machine tool and sensor measurements. At NIST, this third method has been applied to a Monarch Metalist turning center, with thermal feedback.

The advantage of these methods is that such error correction is transparent to the higher levels of the controller; that is, the part programmer or the machinist overseeing operations are not responsible for any additional tasks, and are unaware of the presence of the compensation (except, of course, for the improvement in performance). The disadvantage of these methods is that the

---

[1] Certain commercial equipment is identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the equipment specified is necessarily the best for the purpose.

"black box" which is used to capture, alter, and reinsert the axis sensor data must be built from scratch if is is to be used with another vendor's controller.

Test results of thermal compensation on the Hardinge turning center demonstrated significant improvements in the accuracy in diameter, length, taper, and squareness for a cylinder fabricated from mild steel. For example, for a nominal diameter of 41 mm, the uncompensated machine tool generated an oversize of 57 µm, while the compensated machine tool improved to an undersize of 3.8 µm, an improvement of almost 15 times. The improvement in length was more pronounce for a nominal length of 88 mm, the uncompensated length was oversized by 130 µm, while the compensated length improved to an undersized 6.4 µm, a factor of almost 21 times.

FORCE-BASED FINISHING

In this section, we examine the use of force information for finishing operations such as deburring, chamfering, and buffing. Three systems are detailed to demonstrate the improvements realized when force feedback is used to compensate for systematic errors such as part misplacement, kinematic inaccuracy, and part tolerances. The force feedback in these systems is applied to industrial robots, which are used as the tool carriers.

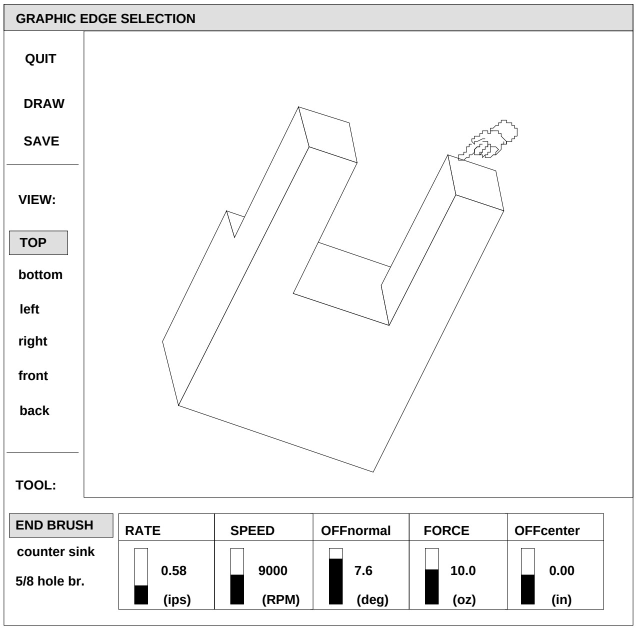
Deburring and chamfering are light machining operations which remove burrs, the rough thin ridges which form on part edges during the bulk machining processes. They are typically followed by buffing to produce a final radius. Because these are often the final manufacturing steps, it is critical that they be performed reliably, since all the value added to a part will be lost if the part is scrapped. Curiously, finishing is often performed manually, introducing an unpredictable human factor at a most critical stage. The reason for this is the common prevailing notion that minor flaws, such as burrs or poor surface finish, will be magically removed at the finishing stage. Because of this, a mixed bag of defects accumulates until only a human, with a capability to reason and adapt, is capable of performing the myriad of operations necessary to remove the defects. Unfortunately, with a human's flexibility comes a tendency to make mistakes, grow bored or tired, or suffer from repetitive stress disorders. In order to improve the quality of chamfering, particularly precision chamfering on complex parts, it is highly desirable to automate it.

One approach to automation is to finish the part on the machine tool immediately after cutting. This method has two advantages. First, there is no downtime incurred by transferring the part. Second, any systematic errors such as geometric inaccuracy and thermal expansion will be similar for both the machining and finishing passes, so their effects will be masked. However, this approach is not always feasible. For example, a part produced on three-axis horizontal and vertical milling machines will in general contain edges that require five axes of position and orientation control to deburr. In cases where five-axis milling machines are available, shop managers may find it hard to justify using them for finishing when there is a backlog of complex machine jobs which cannot be completed any other way. Furthermore, the fine dust generated by chamfering may ruin normal machine tool seals, which are designed for much larger chips.

A second approach is to move the parts to a robotic workcell specialized for finishing. Robots are normally chosen over dedicated part-specific automation systems because they may be reprogrammed to handle a changing inventory of parts. Moving parts to such a workcell presents its own problems, however. The most severe is in registration, where the location of the part edges is known only approximately due to tolerances, inaccuracy of the machine tool in the previous workcell, errors in fixturing, and kinematic and dynamic errors in the robot. The tolerance problem can be overcome by passing the particular choice of dimensions from one workcell to its successor. Machine tool accuracy, part fixturing errors, and robot kinematic and dynamic errors present more difficulty. One way to reduce their effects is to precisely measure the location of the edges, using probes or cameras in a mapping pass. This method may also be extended to detect anomalies such as burrs. Alternatively, the position of the edges may be determined in real time, as the chamfering tool traverses the edges, using force feedback, acoustic emission, or vision information.

It is advantageous to generate robot coordinates based on part drawings or computer-aided design (CAD) files, as they are for machine tool coordinates, instead of using the method of teaching commonly employed in industry. Teaching requires that the robot be brought off the production line, and it also requires a human programmer for each part to be finished. In contrast to teaching, robot coordinates can be interactively generated off-line using a natural interfaced based on graphics, as indicated in Figure 1.

Edges and their finishing parameters may be interactively generated by a user in a matter of minutes, and the resulting robot coordinates can be computed in a matter of seconds using the knowledge of the part geometry present in the CAD file and the location of the part relative to the robot. However, computed coordinates are only as good as the accuracy of the robot. Because the design of robots typically includes articulated joints with a large range of motion, their accuracy is severely limited, especially when compared with machine tools. Because of this, it is crucial that some means to accommodate for this inaccuracy be built in to the robot finishing workcell. Calibration of the kinematics is a first step, which can be improved with models of the backlash characteristics of the joints, or stiffness and inertia models which can predict dynamic quantities such as overshoot. Unfortunately, generating the data from which to develop these models is an exceedingly difficult task. In many cases, the use of proper sensory feedback provides a natural way to overcome the limits of robot accuracy, particularly when they stem from several unrelated or poorly-understood sources.



RATE	SPEED	OFFnormal	FORCE	OFFcenter
0.58 (ips)	9000 (RPM)	7.6 (deg)	10.0 (oz)	0.00 (in)

Figure 1. Graphic Edge Selection. An operator selects the side of the part to be viewed, the deburring tool to be used, and the tool parameters such as feed rate, speed, and force. A mouse pointing device aids in the selection.

The Cleaning and Deburring Workstation

The Automated Manufacturing Research Facility (AMRF) is small-batch manufacturing testbed at NIST, funded by the U. S. Navy. One of the components of the AMRF is a workcell which allows a user to graphically specify part edges which require deburring and the parameters such tool type and feed rate which are to be used. Users can also specify how the part is to be buffed following the deburring operations in a similar manner. This system, the Cleaning and Deburring Workstation (CDWS) [5], is shown in Figure 2.

The robot trajectories for deburring and buffing are generated automatically from the user's finishing requests, without the need for teaching. It is known in advance that the robot coordinates resulting from this off-line programming will not be satisfactory, due to robot kinematic error, part misplacement, and part tolerancing. To accommodate for these errors, a method known as self-teaching has been developed. In self-teaching, the off-line programmed coordinates are used as a baseline for the trajectory, but are modified slightly based on run-time force information. In the case of deburring, the off-line coordinates are approached by the robot, which monitors a force sensor as it brings the tool and part together (for deburring, the robot carries the tool to a stationary part; for buffing, the part is carried to the stationary buffing wheel). Once the desired force level is attained, the robot coordinates are captured and stored, and the next point in the deburring trajectory is approached and taught in the same fashion. This process shown in Figure 3. After all computer generated points are updated by self-teaching, the robot proceeds to deburr the entire part. This process may be repeated for every part in the batch, or periodically to accommodate for tool wear. The key improvement is that human intervention is not required once the edges and their parameters have been selected. For parts requiring hours of human teaching, only minutes of automatic teaching are needed.

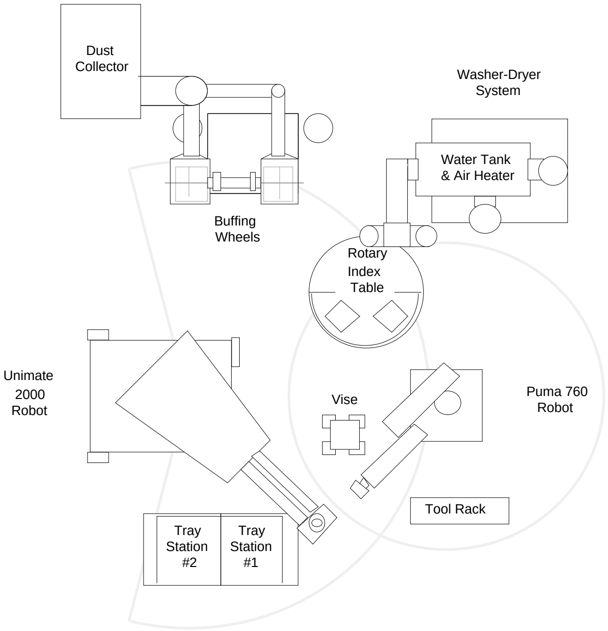


Figure 2. The Cleaning and Deburring Workstation.

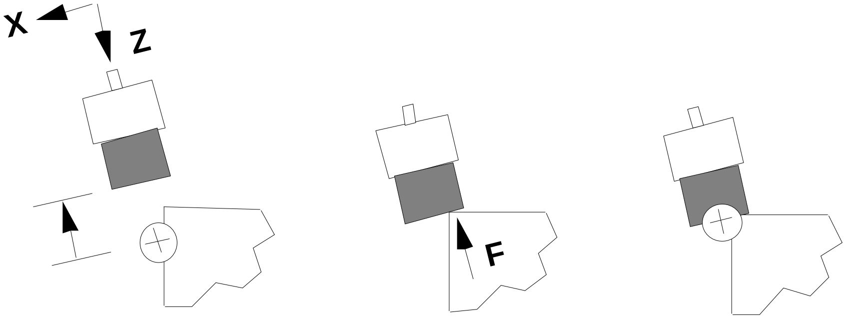


Figure 3. Self Teaching. The robot first approaches the off-line programmed point, monitoring a force sensor until the desired force is attained. The controller then records the new point.

In the case of buffing, a similar strategy is followed. First the robot approaches the buffing wheel, and then monitors a force sensor as it brings the part and wheel together. Unlike the self-teaching method used for deburring, however, the buffing pass stops after the desired force is reached, and all points on the buffing trajectory are offset by the difference between the computed first point and the sensed beginning. This is successful since the part is moved linearly along the face of the buffing wheel, and no complex orientation changes occur. The self teaching accommodates for robot inaccuracy as well as buffing wheel wear.

One enhancement to the self teaching method allows the user to preview how the graphically selected parameters such as force and feed rate will affect the part finish. Using a table of measurements developed previously, a program determines the degree of buffing which will result from the user's selection of buffing force and feed rate across the wheel. This predicted surface finish is then displayed as a mosaic of color-coded facets on the part face, with overbuffed areas appearing in a "hot" color such as red or orange, and underbuffed areas appearing in a "cool" color such as blue or green. If the part finish is perceived by the user to be unsatisfactory, one may reselect buffing force, feed rate, or part position within the wheel and reevaluate the surface finish. This is shown in Figure 4.

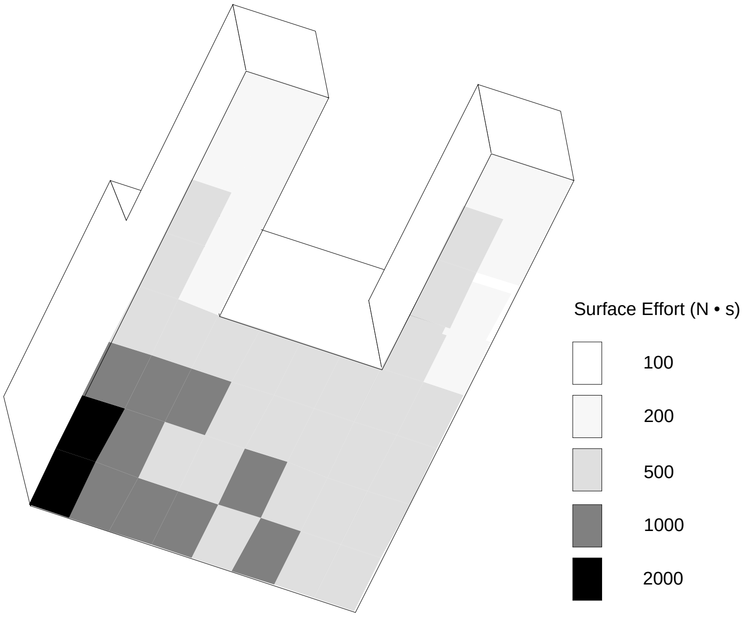


Figure 4. Mosaic of buffing results, which indicates the degree to which the surface would have been finished based on the user's selection of force and feed rate.

In order to control the CDWS robots based on real-time force feedback, the original vendor-supplied controllers had to be replaced with custom controllers developed by NIST which had the capability of integrating force feedback with robot motion control. This has the serious disadvantage in that the techniques developed as a result of this program could not be disseminated to industrial or government users without a major expense in both time and money for controller retrofit. This problem, coupled with the hesitance of any outside organization to rely on non-commercial unsupported products for their production, has meant that these techniques have not found their way out of the laboratory factory and into production.

A second disadvantage of the CDWS is that it is only capable of brushing and buffing operations, which are limited to soft metals such as aluminum and brass. In order to finish harder metals such as titanium and inconel, much harder tooling such as carbide rotary files must be used. The U. S. Navy has a need to process these materials as well, as they form the bulk of the critical engine components for jet aircraft. Unfortunately, the application of hard tooling to edges made of such hard material results in large forces for even small position perturbations. The periodic self-teaching which is sufficient for softer materials must be replaced with high-frequency force control, during which forces are monitored continually as the tool traverses the edge. The inability of conventional robots to respond to force information at the frequency necessary for high-speed force control has led to the development of a second-generation finishing workcell in the AMRF. While this workcell addresses the high-speed force control problem, work is also being performed with an industry partner in order to validate interfaces which may someday form the basis for controller standardization. In the next section, the second generation workcell is described. The standardization efforts are discussed in the final section.

The Advanced Deburring and Chamfering System

Researchers at NIST and United Technologies Research Center are working jointly to develop an automated chamfering workcell, to be applied to the finishing of U. S. Navy aircraft engine components made from titanium and inconel metals. This system is known as the Advanced Deburring and Chamfering System (ADACS) [6, 7]. The strategy is to use a six-axis robot as a coarse positioning device, which carries an actively-compliant chamfering tool to the part edges. The chamfering tool consists of a carbide rotary file, a high-speed electric spindle, force transducers, and an actuated housing driven by stepper motors and lead screws. The tool is controlled independently from the robot, and can be reprogrammed almost instantly to emulate a wide range of stiffness and damping in both the normal and tangential edges directions. Fine motion capabilities allow the tool to track edges based on force feedback, so that edge contours can be traversed and precise chamfer depths maintained in spite of robot inaccuracies, deviations in part geometry, and fixturing errors. The tool is shown in figure 5. This strategy is known as the around-the-arm solution, in contrast to the through-the-arm solution in which the robot itself is given frequent position updates in response to force feedback. The around-the-arm solution has proven to be a more effective method, primarily because robot controller delays, joint backlash, and link inertia limit the control bandwidth to be far below that required to maintain a consistent chamfer depth without breaking into oscillations or limit cycles.

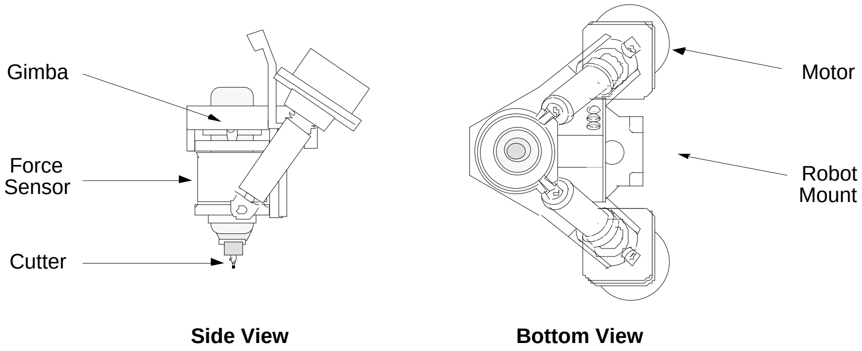


Figure 5. The Adaptive Deburring Tool. The tool consists of a high-speed spindle and rotary file mounted in a two degree-of- freedom motorized housing. Force transducers inside the tool monitor forces each millisecond, and a dedicated controller computes tool tip motion required to maintain a constant force.

The expanded capability of the ADACS puts a burden on the part programming system to generate commands for both the robot and tool. Furthermore, new chamfering strategies need to be developed which make use of the ability of the robot and tool to work together to finish an edge with a precise force. As examples of the complexity of processing available in this new system, users now have the ability to program guarded moves, map the part edge prior to processing, and perform micromachining during which the robot is stationary while the tool moves in a constrained region. The expanded capability is far beyond the capabilities of the vendor-supplied robot controller. Presently, a custom controller is used which presents a control hierarchy to the user. At each level in the hierarchy, a set of tasks is available which may be selected to form the basis of a program. At a high level, such a program may perform feature chamfering, allowing the user to simply select the desired features to be chamfered. At a lower level, a program exists to chamfer a single edge. At an even lower level, programs which move the robot to approach points and perform guarded moves are available. While the custom controller still makes technology transfer to industry very difficult, several efforts are being undertaken to improve the possibility for technology transfer. First, the work is being performed in conjunction with an industrial partner, United Technologies Research Center, which was selected from a competitive pool of respondents. The interaction with an industrial robotics user and manufacturer insures that research will be guided by the needs of industrial users and not by the ideals of laboratory researchers. Second, the work is being conducted in cooperation with several attempts to standardize the controller industry. The ADACS serves as a testbed which points out the need for interfaces to robot controllers that will enable force control to be fully utilized. The hope is that more programs of a similar nature will identify a comprehensive baseline of sensor techniques and desired controller capabilities that can be used to specify a reasonable set of interfaces that can be standardized. This topic is the subject of the following section.

OPEN ARCHITECTURE CONTROLLERS

All of these techniques previously described have been developed and tested, and have demonstrated real improvement in the accuracy of finished workpieces. An immense obstacle to applying these techniques on the factory floor is the proprietary architectures of most industrial machine tools and robots. A solution to this problem lies with open architectures. An open architecture is the definition of the components of a system, their function, the interfaces with which they communicate with each other and the external world, and the structure and meaning of information which crosses those interfaces. By that definition alone, all existing controllers can be made to present an open architecture simply with the addition of a comprehensive set of documentation. A standard open architecture is an open architecture whose definition is adhered to by some reasonably large portion of an industry (a *de facto* standard), or one that is formally standardized by a standards body (a *de jure* standard). However, standard open architectures may fall short of specifying interfaces which are required of some innovative application, in which case the standard open architecture is no better than a proprietary one. In order for the specification of an open architecture to be considered valuable, the specification must be made rich enough to support the broad variety of applications which exists for the specified system, as well as any which may be anticipated to exist in the near future.

In the author's experience, complaints against open architectures have been lodged mainly by the vendors of control systems, while praise for open architectures have been voiced by end users and developers. Each of these arguments by users and developers has been given to the author in support of open architectures:

- open architectures give users multiple and competitive sources for replacements and enhancements to their equipment;

- users have more confidence in their decisions to purchase open architecture controllers, since the disappearance of one vendor does not mean the immediate obsolescence of their equipment;

- third-party developers of enhancements (such as thermal compensation) now have a vastly expanded market for a single product which previously required expensive customization for disparate platforms.

The classic example of the benefit of open architectures to users and third-party developers is the personal computer. The personal computer has been attributed to have launched a revolution which transformed business, science, and education by bringing computational power to every desktop. This transformation was accomplished through specifications which allowed third parties to develop a wide variety of software and hardware, spanning a broad price and performance range, allowing users to tailor computation to their particular problems and budgets. However, this revolution had a severe negative impact on the providers of larger computers who were not poised to enter the personal computer market. Many of these companies found themselves facing bankruptcy with almost no market share, or forced into alliances with their competition. Ever mindful of the threat to the delicate balance of business and the market, controller vendors can hardly be faulted for approaching the open architecture issue with circumspect. Each of the following arguments by controller vendors has been given to the author in support of proprietary architectures:

- in-house development of both hardware and software gives the vendor control over safety and reliability, and the peace of mind that they will not be liable for malfunctions of other vendor's products;

- developing custom hardware allows the vendor to optimize the component selection and not pay recurring costs for unused features of all available equivalent third-party hardware;

- proprietary hardware interfaces (such the computer bus), and proprietary software interfaces (such as the operating system) provides the vendor with a market niche and the opportunity to sell supporting products.

The first argument is difficult to resolve. It cannot be answered based on an examination of the example presented by the personal computer industry, since damage to products, personal injury, or death are almost entirely foreign to that domain. However, an intelligent formulation of an open architecture should address the safety issue, and allow controller vendors to limit the degree to which third party enhancements can affect the operation of the machine. Watchdog safety systems, for example, provide such a guarantee. Watchdog systems operate independently from the machine tool or robot, monitoring the operation and immediately safing the machine in the event of a safety violation. While true watchdog systems do add significant cost, their function can be built in to the vendor's controller in a way which conforms to the architecture specification while remaining inviolable.

The basis for the second argument may actually be eliminated with the adoption of an open architecture, since the limited offerings from third-party vendors which may have forced internal development would be expanded upon the increase in market. It may be true, however, that one may always realize a reduced cost by designing hardware to fit exactly around the application to be supported, or by comparing development costs with the profits charged by third-party suppliers of the equivalent component. In fact, an open architecture specification should reward vendors who provide custom hardware which optimize cost by providing them a market for their product which did not exist previously. This may be an answer to the concerns raised in the third argument. Ultimately, the choice rests with the vendor and one's business analysis.

The problems presented to controller vendors need to be balanced with the benefit to users of the controller. This is the focus of the Next Generation Controller program, which is an Air Force-funded project which will define a Specification for an Open System Architecture Standard (SOSAS) for machine tool and robot controllers [8, 9]. A complementary effort, the Low End Controller (LEC) program, is a consortium of machine tool controller vendors funded through the National Center for Manufacturing Sciences and their own in-kind contributions. The goal of of the LEC is to develop a specification for a machine tool controller architecture of reduced scope. These programs are currently underway, and will generate specifications that will address the needs of both controller vendors and end users of machine tools, foster a fertile community of third party enhancement vendors, and regain some of the market share lost to international competition.

REFERENCES

1. Donmez, M. A., D. S. Blomquist, R. J. Hocken, C. R. Liu, and M. M. Barash, “A General Methodology for Machine Tool Accuracy Enhancement by Error Compensation,” Precision Engineering, Publication No. 0141-6359/86/040187-10, 1986.
2. Donmez, M. A. (editor), “Progress Report of the Quality in Automation Project for FY90,” NIST Internal Report NISTIR 4536, March 1991.
3. Donmez, M. A., Kang Lee, C. Richard Liu, and Moshe M. Barash, “A Real-Time Error Compensation System for a Computerized Numerical Control Turning Center,” Proceedings of the IEEE International Conference on Robotics and Automation, San Francisco, CA, April 1986.
4. Yee, Kenneth W., Herbert T. Bandy, Jack Boudreaux, and Neil Wilkin, “Automated Compensation of Part Errors Determined by In-Process Gauging,” NIST Internal Report 4854, June 1992.
5. Murphy, K. N., R. J. Norcross, and F. M. Proctor, “CAD Directed Robotic Deburring,” Proceedings of the Second International Symposium on Robotics and Manufacturing Research, Education and Applications, Albuquerque, NM, November 16-18, 1988.
6. Murphy, K. N., and F. M. Proctor, “An Advanced Deburring and Chamfering System,” Proceedings of the Third International Symposium on Robotics and Manufacturing, Vancouver, B. C., Canada, July 18-20, 1990.
7. Proctor F. M, and K. N. Murphy, “Keynote Address: Advanced Deburring System Technology,” presented at the American Society of Mechanical Engineers Winter Annual Meeting, San Francisco, CA, December 10-15, 1989. Published in Mechanics of Deburring and Surface Finishing Processes, PED-Vol. 38, ASME, New York, NY, 1989.
8. National Center for Manufacturing Sciences, “Next Generation Workstation / Machine Controller Requirements Definition Document,” Document NGC-0001-011-000-RDD, August 1990. Available to U. S. citizens and companies through United States Air Force WRDC/MTPM.
9. Martin Marietta Astronautics Group, “Next Generation Workstation / Machine Controller Architecture Definition Document,” Document NGC-ADD-01, June 1991. Available to U. S. citizens and companies through United States Air Force WRDC/MTPM.