

The Vertical Machining Workstation of the AMRF:  
Software Integration

Charles R. McLean

Factory Automation Systems Division  
Center for Manufacturing Engineering  
National Bureau of Standards  
Gaithersburg, MD 20899

Presented By:

Charles R. McLean  
ASME Symposium on Intelligent and Integrated Manufacturing  
Chicago, Illinois  
December, 1986

Bibliographic Reference:

McLean, C. R. "The Vertical Machining Workstation of the AMRF:  
Software Integration", Proceedings of the 1986 ASME Symposium on  
Intelligent and Integrated Manufacturing, Anaheim, CA, December  
1986.



The Vertical Machining Workstation of the AMRF:  
Software Integration

Charles R. McLean

Group Leader  
Production Management Systems  
Center for Manufacturing Engineering  
National Bureau of Standards  
Gaithersburg, MD 20899

ABSTRACT

The software architecture of the Vertical Machining Workstation (VWS) in the Automated Manufacturing Research Facility (AMRF) at the National Bureau of Standards is presented. The prototype system demonstrates flexible computer-integrated manufacturing for a family of prismatic parts. The workstation software components include: a feature-based design system for defining part geometries, an automatic process planning and NC code generation system, a state machine-based hierarchical control system which executes process plans, a diagnostic tools package, and mailbox communications software. The mechanical components of the workstation, described in a companion paper [1], include: a CNC vertical machining center, a robot and locally implemented gripper system, a pneumatic vise, a vacuum chip removal system, local part storage, and tray roller tables for a robot cart materials delivery system. The system is capable of running in stand-alone mode as a single station computer-integrated manufacturing system, or remotely under the AMRF cell controller.

The workstation software has the following features: (1) The user interface to the system is based on a series of prompts and menus which make it easy for a novice user to design and manufacture complex parts. Graphics displays provide constant feedback to the user. (2) Most of the system software is implemented in Lisp and supports the incremental addition of new knowledge or intelligence about manufacturing processes. (3) The process plans have a format that is uniform across all AMRF control systems. Plans stored in the neutral file format may be generated locally in the VWS, or remotely at process planning workstations. (4) The control software executes process plans directly without translation, sequencing the major activities of all mechanical systems within the workstation. (5) New activities or capabilities can be quickly added to the workstation library by defining new work elements which are invoked from process plans. (6) A large portion of the workstation planning and control software is generic and independent of the manufacturing

application. The same body of software developed for the VWS is currently being used to implement a material handling system and a cleaning/deburring workstation.

## 1. INTRODUCTION

The work described in this paper is part of a larger ongoing research effort in factory automation standards at NBS, referred to as the Automated Manufacturing Research Facility (AMRF) project. The U.S. Navy Manufacturing Technology Program co-sponsors the AMRF and is a major partner in its work. This effort is also sponsored by other government agencies, industry, and the academic community.

### AMRF Project Goals

A major goal of the AMRF project is to establish a testbed small batch manufacturing system at the NBS Gaithersburg, Maryland site that can be used by government, industry and academic researchers to develop, test and evaluate potential interface standards [2,3,4,5]. The results of AMRF research are being transferred to the private sector via the NBS Industrial Research Associate Program, participation in factory standards organizations, conference papers, technical reports, and various technology transfer programs that have been established with project sponsors and other participants.

A portion of the AMRF testbed system became operational in 1983 and was subsequently demonstrated to the public during four integration test runs between 1983 and 1985. These test runs demonstrated both the automated production of small batches of machined parts and the unique nature of the AMRF system. The NBS testbed differs from virtually all other flexible manufacturing systems in the variety of "off-the-shelf" components that have been integrated into a single coordinated operation. A significant level of technical effort was required to integrate the subsystems obtained from different commercial vendors. The incompatibility of these systems and the costs associated with

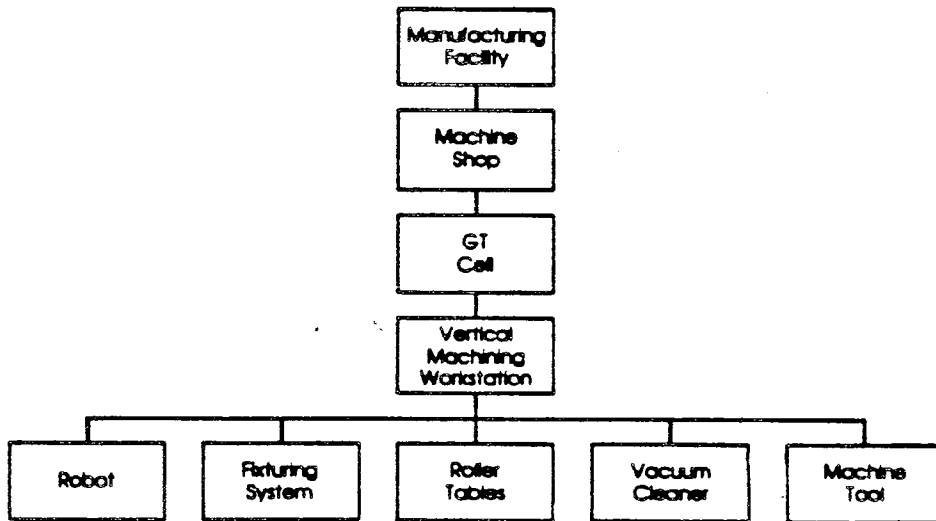


Fig. 1 The Vertical Machining Workstation in the AMRF control hierarchy

integration efforts currently keep automation out of reach of the small job shops that would benefit the most from it. The development of a generic factory architecture and the specification of standard interfaces between manufacturing subsystems would alleviate this problem by increasing the potential for "plug compatibility" between commercial products offered by different vendors [6,7].

### Overview of Paper

This paper describes selected software of the Vertical Machining Workstation (VWS) and its functional capabilities. Section 2 identifies project objectives and system functional requirements that the workstation satisfies. Section 3 presents an overview of the internal structure of the workstation in terms of several major software modules. A number of other subsystems are identified, but not discussed. A scenario in section 4 illustrates workstation operations in stand-alone mode. Section 5 summarizes concepts, system implementation status, and current directions.

## 2. WORKSTATION REQUIREMENTS

The control of production activities at the workstation level is the primary required capability of the Vertical Machining Workstation software. Within the AMRF manufacturing management

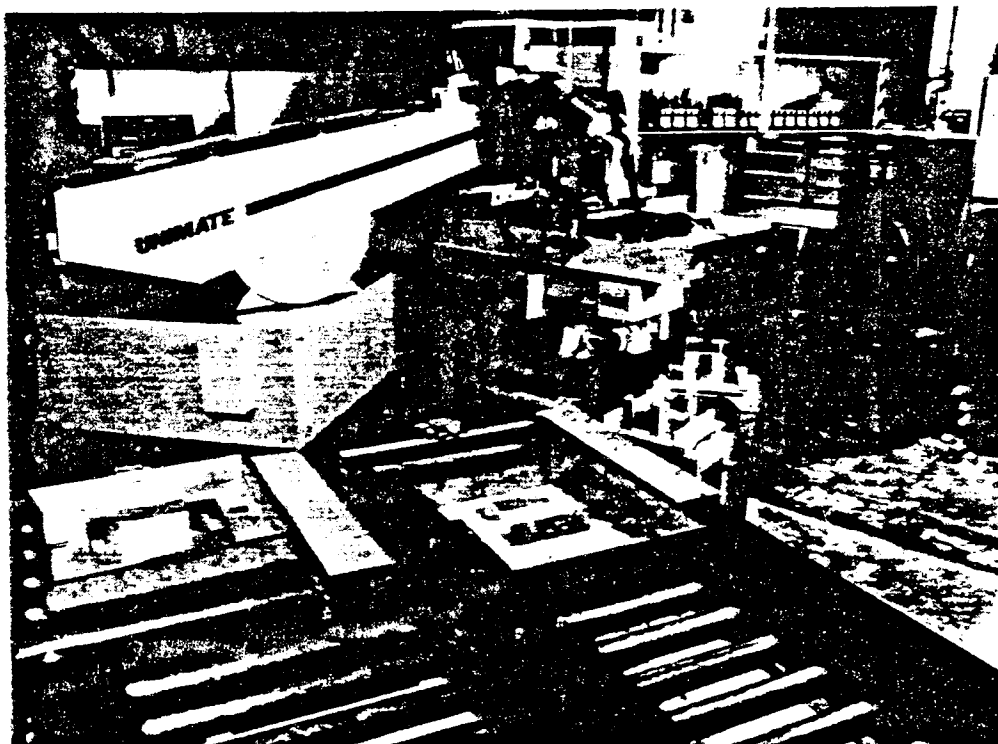


Fig. 2 Photograph of the hardware systems of the VWS

hierarchy, workstation control is the second lowest level (Figure 1). Workstation controllers are responsible for coordinating the operations of small functional groups of industrial equipment. The equipment managed by the VWS control system pictured in Figure 2, includes: 1) a robot manipulator equipped with an interchangeable gripper system, 2) a numerically controlled vertical machining center, 3) two tray roller tables, 4) a vacuum cleaner which is used as a chip removal system, and 5) two fixturing devices: a pallet mounting system and a flexible vise. A more detailed discussion of this equipment and its front-end device controller is presented in a companion paper [1].

#### Preparation of Manufacturing Data

A second major requirement of the software is the incorporation of a local manufacturing data preparation system to permit stand-alone operation of the system. Data preparation is defined to include the specification of: 1) geometry data, 2) process operation sequences, and 3) part programs for NC machine tools. A feature-based design system is to be provided for the entry of all required part geometry data. All part specific machining, handling and support operations are to be defined in workstation and equipment level process plans. These process plans are to be written in the uniform format developed by the AMRF Process Planning Project. Process plans must be internally verified to ensure that no unsafe operations attempted. NC programs are to be generated automatically from feature-making operations specified in machine tool process plans. The data preparation system must be flexible enough to support a diverse and changing part mix.

#### Other Major Objectives

Other major objectives of the workstation software include: 1) the total integration of all component subsystems, 2) one person or unattended operation in stand-alone mode, 3) a friendly user interface based on menus and graphics displays, 4) simple error recovery techniques, and 5) the flexibility to add in new processing capabilities, as library routines, without modifications to the basic system. The software must provide interfaces to support the integration of the workstation with the AMRF cell control system, the communications network, and the distributed data administration systems. The workstation must be able to run unattended, under the remote control of the cell. It must also be possible to remotely program the workstation by generating process plans externally. When the workstation is running in integration mode, it must retrieve all required data from the AMRF common databases. Finally, the workstation control system must support interfaces to the material handling system for the receipt and shipment of materials and tooling.

### 3. SYSTEM DESCRIPTION

The Vertical Machining Workstation controller is a software package developed at NBS that directs the activities of subordinate equipment level control systems, provides manufacturing data preparation functions, and integrates the workstation with the rest of the AMRF manufacturing system. The architecture of this system is based upon a number of the real-time control concepts developed by the NBS robotics project [8,9]. The system also incorporates many extensions to these concepts developed by the AMRF cell control [10,11], process planning, and hierarchical control system emulator (HCSE) projects [5]. The VWS

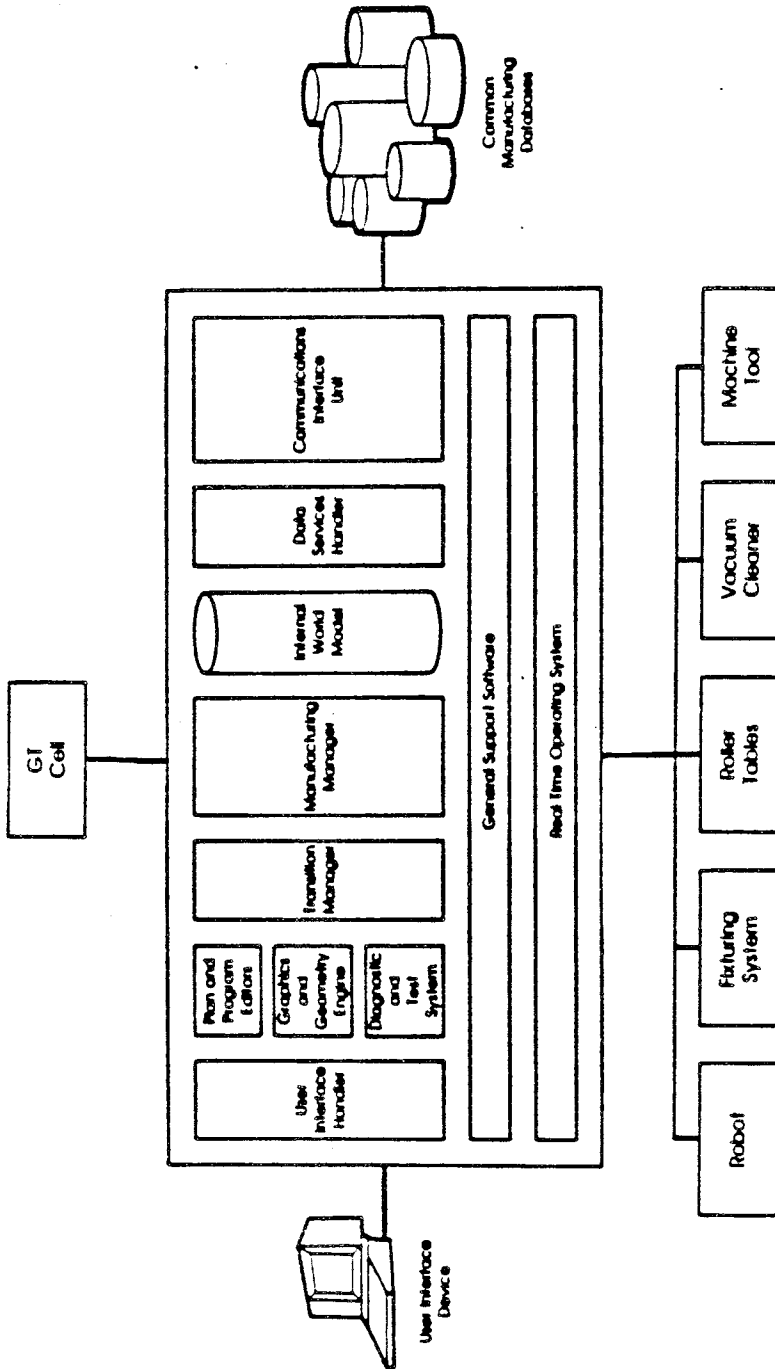


Fig. 3 The major software modules of VMS control system

software is implemented primarily in LISP, a symbolic programming language that is used extensively for artificial intelligence applications. The Franz Lisp environment, in which the VWS control software is programmed, runs under the Unix (tm) operating system on Sun computer workstation. The SunCore (tm) graphics package was used to program the VWS visual displays. Communications interfaces to the device control computer are implemented in the C programming language.

The software modules of the workstation controller that are addressed in this paper include: 1) an interactive process planning system for specifying sequences of machining and support operations, 2) a manufacturing management system, based on a hierarchy of state machines, that sequences work order processing within the workstation and 3) an state machine operating system that sequences the execution of control modules. Other VWS modules, not detailed in this paper, include: 1) a feature-oriented part design system, 2) an automated system for generation of process plans from feature-based part specifications, 3) an automated system for generation of numerical control code from feature data, 4) the library of functions or subroutines, called work elements, that provides basic manufacturing capabilities, 5) the internal world data model, 6) a transition manager, which sequences the system through major evolutions, such as: initialization, restart, error recovery and shutdown, 7) a data services handler, which manages interaction between the internal world model and the global database, 8) a communications interface unit, 9) a diagnostics and test system, 10) graphics display generators, and 11) general support software which provides access to data files, builds new versions of the system, etc. A discussion of these modules will be the subject of another paper. Figure 3 illustrates the decomposition of the vertical machining workstation controller into its major component modules.

A brief overview of selected modules of the VWS software follows. Each overview first presents applicable concepts and definitions. It then describes the functions that have been implemented in the version of the system.

### Process Planning System

Concepts and Definitions. Within the AMRF, process planning is used to specify both the material processing and the support operations that are necessary to transform raw material into a finished product. The work orders that are entered locally or assigned by cell control, the vertical machining workstation's supervisor, refer to process plans. The VWS controller obtains the referenced process plan from the local computer file system or the AMRF database to determine the steps that are required to carry out the order. Although the data contained within a process plan is system dependent, the same basic plan structure is used by all controllers. Every process plan contains the following major sections:

**Parameters** - lists of variables that are used within the plan and are bound at execution time. Default values, if applicable, are also specified.

**Header** - identifies the plan, version number, planner name, part name, identification number and revision, important dates, group technology codes, customer identification, and engineering changes, if applicable.



Requirements List - specifies all tools, fixtures, materials, programs and process plans that are required to carry out this process plan.

Procedure Specification - defines the steps necessary to carry out a production process or a support activity. Each step has a step number, a work description (consisting of a work element and its associated parameters), precedent steps (steps that must be carried out prior to the execution of this work element), and an estimated duration for the activity.

A hierarchy of process plans has been defined that corresponds to the AMRF levels of control. The levels and the names given to their associated plans are:

- Cell level: Routing Slip
- Workstation level: Operation Sheet
- Equipment level: Instruction Set

Operation Sheets and Instruction Sets may be prepared for the vertical workstation using either the local VWS plan editor or the central AMRF process planning system.

Plan Editor. The plan editor is activated by selecting the process planning option from the main VWS command menu. Upon entry to the editor, three windows are displayed on the screen (Figure 4): 1) an menu options/data window on the right hand side, 2) a command window in the lower left, and 3) a part display window in the upper left hand corner. The main process planning menu, displayed on system entry, offers command options such as:

- Plan file manipulation: list, edit, clear all, delete, create, rename, display, copy, and save files.
- Step editing: change, insert, delete, and resequence steps.
- System operations: quit editor, enter break package, clear screen and display command menu.
- Graphics and verification: turn graphics display on/off, turn verification testing on/off, locate position on screen using mouse, redraw screen, draw or flash a selected feature, and part display to a picture catalog.

Many other menus and prompts appear at appropriate times during the plan editing process.

If the edit plan option is selected, the user is asked to select an operation sheet or instruction set. By selecting an instruction set and then the insert step option, the user is prompted for information that is required to specify a new feature-making operation. First, the user is asked to select from one of the following machine tool work element types: INIT (initialize machining sequence), GROOVE, POCKET, TEXT, HOLE, COUNTERSINK, THREAD, CHAMFER\_IN, CHAMFER\_OUT, SWEEP, or CLOSE (end machining sequence). Next, if it is appropriate, the user is asked to select

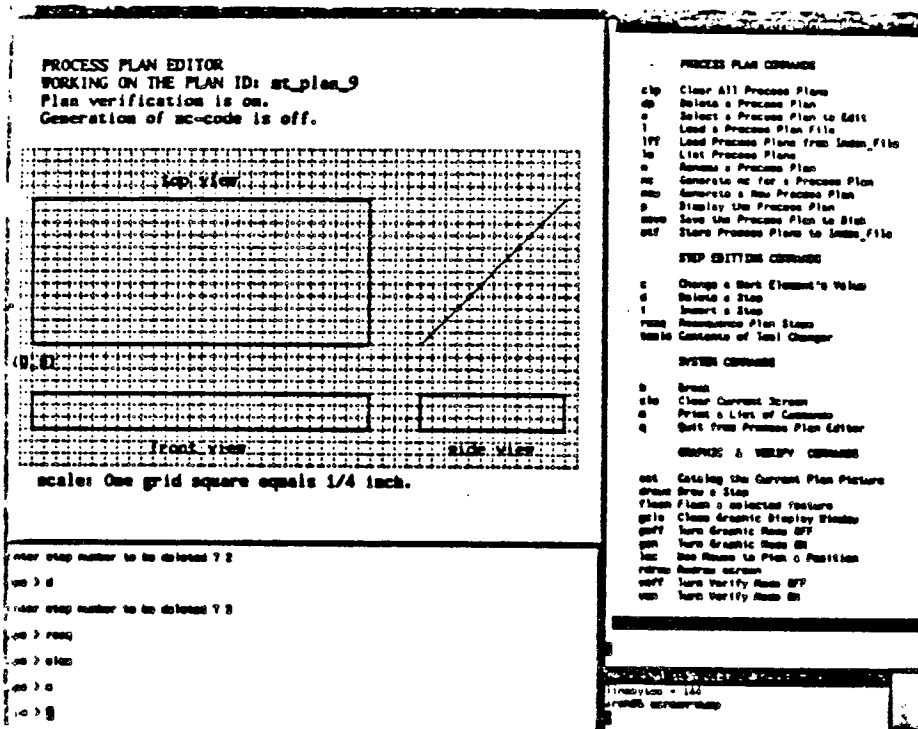


Fig. 4 Entry display screen for local process plan editor

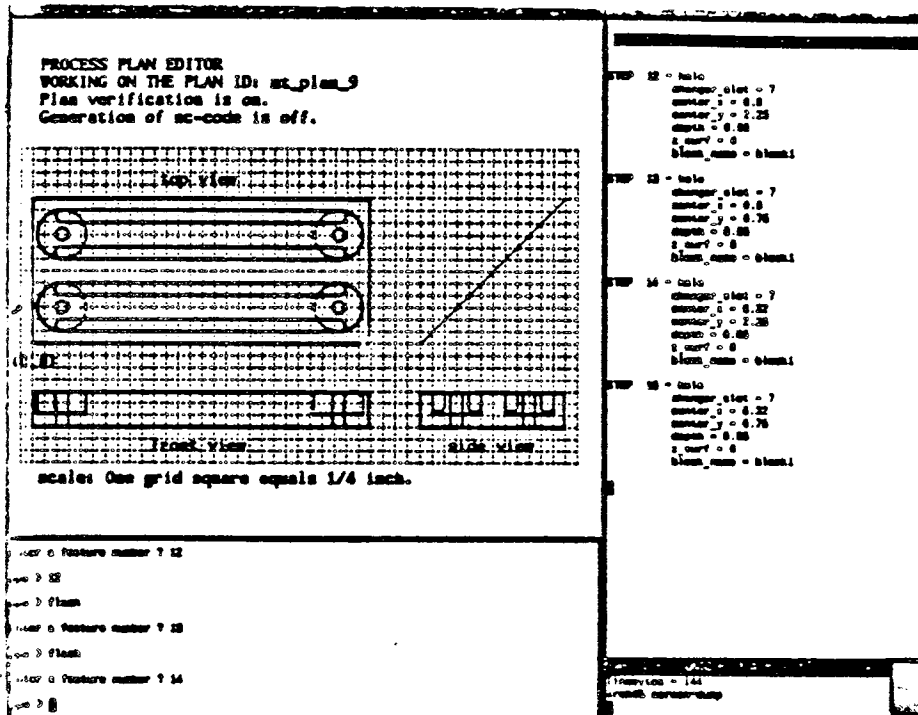


Fig. 5 Process plan editor screen showing completed part

a cutting tool from a catalog of tools in the changer. The catalog specifies a tool number, a tool type and a tool diameter. The prompts that follow tool selection ask the user to enter the following types of information: feature location, dimensions, corner radii, and z-surfaces for initial positioning of the cutting tool. After all required data for a step is entered, the work element parameters are verified; the feature is graphically represented on the part; and the steps of the current plan are displayed in menu/data window (Figure 5).

If the verifier is turned on, it will perform a number of logic and safety checks that are dependent on the operation type. Some examples of verification checks are: 1) feature placement: containment of the feature in a safe work volume within and around the part, maintenance of minimal part wall thicknesses, 2) tool and feature interactions: compatibility of feature corner and tool radii, minimum feature dimensions sufficient for tool clearance, tool type appropriate for specified operation, and 3) feature dependencies: required reference features exist, and the type/size of reference features is correct. When all desired features have been entered and verified, the process plan is usually saved to a disk file. At this point, the user may exit the process planning system, return to the maintain control menu, and order the workstation to produce the part.

#### Manufacturing Management

Concepts and Definitions. A protocol has been established within the AMRF by which supervisors and subordinates exchange control information. There are two basic types of control messages that are exchanged: commands and status. A supervisor uses command messages to tell a subordinate what to do. A subordinate uses status messages to report progress in carrying out commands. These control messages carry pointers to work order records that are maintained internally or within the AMRF database.

Within the AMRF, two types of work orders have been defined: jobs and tasks. A job is an order that is assigned to a control system at its top level, typically by its supervisor. The controller refers to a process plan to decompose a job into lower level work orders, called tasks. These tasks are either carried out directly by the controller or assigned to subordinates for execution. From the subordinate's point of view, these tasks may be complex jobs, which it may in turn decompose into simpler tasks. Jobs and tasks are stored in work order tables. All order tables have the same data elements:

Order Number - a unique integer identifier assigned to the order by the supervisory controller.

Work Description - a structure, derived from a process plan, consisting of a work element and values which parameterize the work element. Parameters may identify: subsystems to carry out work, process plans which decompose the work order, tools and materials to be used, etc.

Precedent Orders - work orders which must be completed before this one can be initiated.

Priority - the priority level of the work order assigned by the supervisory control system.

Duration - an estimated time to complete the activity specified in the work description.

Critical Start and Completion Times - the earliest and latest times at which the work order must be started and completed. These times are used to create a critical path network.

Scheduled and Actual Times - the times at which the order was, or will be, started and completed, as appropriate.

Current Action - a parameter indicating the current action to be taken, such as: PLAN, EXECUTE, STOP or CANCEL.

Current Status - a parameter to indicate the progress that has been made on the order, such as: ACKNOWLEDGED, BUSY, DONE, ERROR, STOPPED, or CANCELLED.

Some fields, that is data elements in the table, are written by the supervisory control system, while others are written by subordinates. The supervisor writes "action" fields which tell the subordinate how or when to perform the work and reads "status" fields written by the subordinate. A subordinate reads the "action" fields and writes "status" fields which report its progress in carrying out the order. A given work order table may be referred to as either a "task" or a "job" table, depending on whether the table is currently being accessed by the system or by its supervisor, respectively.

Manufacturing Manager Module. A generic system has been developed which performs the work order processing functions described above and determines the sequence of production activities that a controller performs [10]. This system is comprised of two major groups of subsystems: a library of work elements and a production control module (PCM). The work element library contains subroutines or functions that implement basic manufacturing operations, such as the machining operations presented in the discussion of process planning. A more detailed discussion of the VWS work elements is beyond the scope of this paper. The PCM is implemented as a three level hierarchy of finite state machines (FSM) modules (see Figure 6). The FSM modules are named, from the highest to the lowest level: Production Manager (PM), Queue Manager (QM), and Dispatch Manager (DM). The procedure that is used by the PCM to process work orders is described below.

The Production Manager resides at the top level of the production control module hierarchy. It is responsible for: processing command messages that originate from the user or the supervisory control system, decomposing assigned job level work orders into tasks as specified by process plans, passing these tasks to the Queue Manager for execution sequencing, determining overall job status from the individual task status data provided by the QM, and generating status messages that reporting job progress to the user or the supervisor. The VWS Production Manager currently accepts only one job from the user or the supervisor for processing at a time.

The Queue Manager resides at the second level of the production control module hierarchy. It is responsible for coordinating the execution sequence of tasks created by the Production Manager, maintaining task critical path networks which

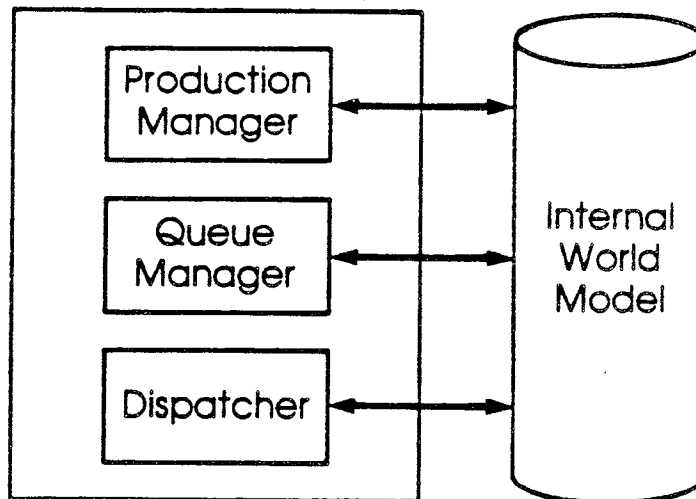


Fig. 6 Layers of the Production Control Module

define precedent and timing constraints on tasks, monitoring the status of system resources, generating schedules on the basis of task constraints and the availability of required resources, releasing ready tasks to Dispatch Managers for processing, monitoring DM progress reports, and providing status to the PM. The current VWS Queue Manager coordinates linear task sequences, but does not yet incorporate timing data in its decision process.

The Dispatcher Manager resides at the third and lowest level of the production control module hierarchy. It is responsible for managing the activities of one subordinate control system, initializing the subordinate control system, monitoring its readiness condition, generating subordinate command messages for tasks released for execution by the Queue Manager, monitoring the progress of task execution by the subordinate, and reporting task and resource status to the QM. The VWS Dispatch Manager currently performs all defined functions.

#### Control State Sequencing

Concepts and Definitions. In the AMRF, control modules are arranged in a hierarchy which distributes and defines factory management responsibilities. Modules at each level in the hierarchy must be able to adapt to carry out assigned work orders under constantly changing conditions. A controller's adaptive behavior must be predictable, i.e. deterministic. State machine concepts, derived from automata theory, provide a mechanism for implementing deterministic behavior within control modules.

State graphs are used to specify all possible inputs, outputs, states and state transitions for a control system. State tables are programming structures that are used to implement state transition graphs within computing systems. A state table is divided into condition and action sections. Each line in the table associates conditional tests, that define system states, with actions to take when the system is in that state. Conditional tests are based on such information as: current system state, supervisor's command, sensory input, subordinate feedback, and world model checks. Actions that are defined for each state include: updates to the internal world model, commands to subordinates, feedback to the system's supervisor, and the setting of state variables.

A time interval, called a control cycle, is defined for each control module, which determines how its table is processed. In order to achieve real-time control, a module's control cycle must be short enough to maintain system stability -- that is, it must identify its current state and generate outputs before the behavior of the system deviates beyond acceptable limits. The state table processing operations that must occur during a control cycle include: sampling inputs, preprocessing state variables, locating the current state in the table, handling error conditions, executing procedures associated with the current state, postprocessing state variables, and generating outputs. Two major error conditions include: 1) a no match condition, where a match for current state variables cannot be found in the state table, and 2) a multi-match condition, where more than one line in the table matches. In the AMRF, control modules that are based upon state table structures are called "finite state machine" or FSM modules.

State Machine Operating System. A special production management operating system (PMOS), has been implemented to run the finite state machine modules in the Lisp environment of the Vertical Machining Workstation. Each FSM control module has the following sections: a module identification header, variable declarations, preprocess procedures, a state table, no match procedures, multi-match procedures, postprocess procedures, and local function definitions. The state machine modules are defined by LISP property list structures. A full-screen state table editor is used to develop and modify the finite state machine control modules.

The PMOS software performs the following functions: 1) system initialization, 2) loading of module files, 3) compilation of modules into a streamlined, run-time format, 4) scheduling of module execution cycles, 5) synchronization of data transfers into and out of the module, and 6) the sequencing of operations with a execution clock cycle.

The PMOS software determines which modules are scheduled to execute during a given control cycle. It copies data, as requested by each control module, from common memory into module data buffers. After transferring all required data into the module, it is executed by the following steps: internal module data bindings are setup within the LISP environment, preprocess function calls are made, the state table is processed, postprocess function calls are made, and internal module data bindings are removed from the LISP environment. Finally, when each scheduled module has been executed during a control cycle, outgoing data is transferred from internal data buffers into common memory. This read/write synchronization of common memory is used to effectively create parallel execution of modules on a single processor, and to prevent

race conditions from developing when control modules execute asynchronously.

#### 4. OPERATIONAL SCENARIO

The AMRF divides manufacturing activities into two major phases: data preparation and production operations. The current implementation of the Vertical Machining Workstation provides a number of different options for carrying out these two phases [13]. The following scenario briefly describes the sequence of activities that must be performed to make a part and identifies some of the options that are available to a user of the system. A short discussion of error recovery is also included.

##### Data Preparation

A feature-based design system is used to graphically design and specify part parameters. The design system supports designs based on rectangular prismatic part blanks that include the following feature types: straight and contour grooves, pockets, chamfers, holes, threaded holes, and text. Although many feature types remain to be implemented, an infinite variety of parts can be generated from this limited feature set. After a part is interactively specified and verified using the design editor, an option is offered to automatically generate machine tool process plans and associated NC part programs.

During process planning, operation sheets and instruction sets are prepared which identify all production and support operations that the workstation must coordinate. The work elements specified in the operation sheet may include: 1) the robot loading of parts, 2) fixture sequencing, 3) NC program generation, 4) program

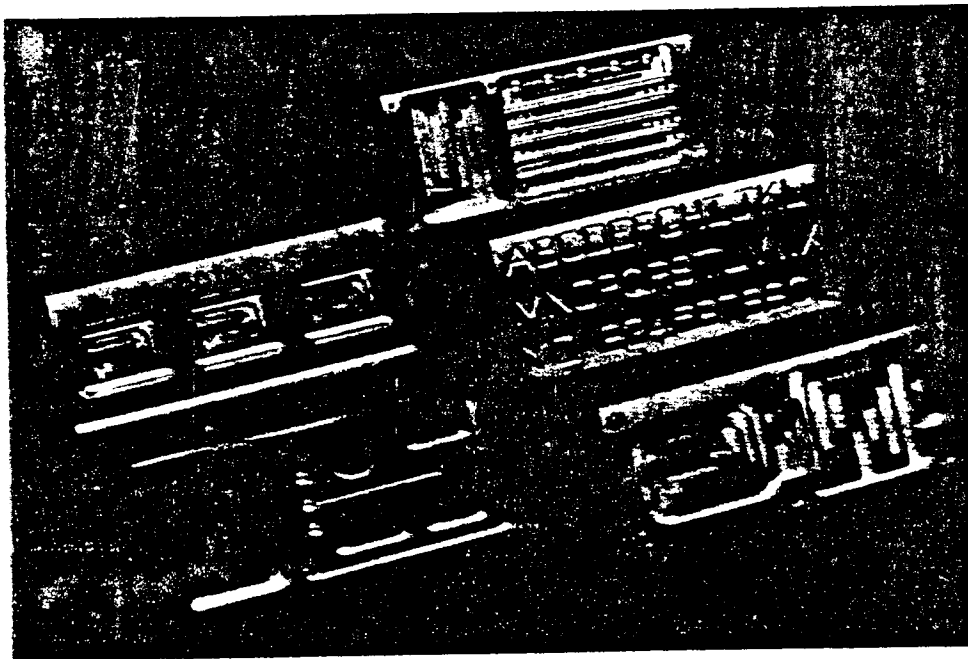


Figure 7. Test parts produced using automatic NC code generation from process plans during the first test run of the VWS in June 1985

downloading to the machine tool, 5) NC program execution, and 6) robot part unloading. A workstation level work element may have parameters that refer to equipment instruction sets. Machine tool instruction sets specify the parameters required for generating machinable features. The NC code generation system can create a part program either during the data preparation phase or during the production phase of the manufacturing operation. Figure 7 illustrates some of the parts that were produced using the process plan editor and automatic NC code generation during the first test run of the system in 1985.

A third data preparation option permits the preparation or generation of process plans and part programs external to the Vertical Machining Workstation. Plans may be edited using the AMRF process planning system and transferred to the VWS via the communications network and the common databases. Similarly, part programs may be prepared using a commercial CAD system and may be transferred to the workstation across the same network. Work elements are provided to download these programs to the machine tool.

### Part Production

When all data has been prepared by one of the options described above, the user returns to the execution subsystem to enter the order to produce a part. An order is entered into the system through a series of order entry prompts on the main menu. These prompts ask the user to specify the operation sheet and machine tool instruction set to be used in making the part.

The Manufacturing Manager subsystem of the VWS retrieves the specified process plan and creates orders for each step in the plan in its internal task table. Tasks without precedent tasks are ready to begin immediately. The Production Control Module releases the next task that is ready, and puts others in wait states. As the equipment level subsystems report that they are ready for new work, by sending status feedback messages, the workstation generates new commands with parameters determined by process plan parameters. These activities are stepped forward based on the clock cycle sequencing of the PMOS state machine subsystem.

### Error Recovery

Occasionally during the production process, a system error may occur. One error encountered during the initial runs of the VWS system involved part fixturing. At that time, an automated chip removal system was not available. A stray chip would sometimes prevent a part from seating properly in the automated fixturing system. Sensors connected to the device control computer would detect this situation, stop further execution of the current command, and provide a feedback error message to the workstation controller software.

A technique was developed to quickly recover from this type of error. After the error message is displayed on the screen, the user enters a command to enter a special diagnostic subsystem. From this subsystem, work element commands can be manually selected from menus to step the system out of the error condition. First, a command is sent to open the fixture. Next, chips are manually cleared and the part blank is returned to local part storage area. Finally, the system is manually stepped ahead to a normal completion of the part loading operation. When normal completion feedback of the step is received from the device controller, the diagnostic mode is exited. The Manufacturing



Management system now detects normal completion of the work element and continues the execution of the process plan, as if nothing had happened.

## 5. SUMMARY

A system architecture has been presented which integrates the component subsystems of the Vertical Machining Workstation within the Automated Manufacturing Research Facility prototype small batch manufacturing system. A number of components which have been integrated were obtained from different system vendor. Other major components were implemented at NBS. The level of automation and integration achieved in the system is very high.

The key architectural concepts that facilitated integration included:

- 1) the decomposition of the manufacturing management hierarchy into well-defined levels,
- 2) the uniform treatment of work order decomposition across the hierarchy,
- 3) the definition of standard work elements for all control systems,
- 4) a consistent data structure for process plans at all levels, and
- 5) uniform interfaces to system services, such as: network communications and data administration.

Future efforts of the Vertical Machining Workstation project will focus on completing missing interface links, defining new work elements (especially for the robot manipulator), and investigating the application of artificial intelligence-based problem solving techniques to process planning and operations scheduling within the workstation.

## ACKNOWLEDGEMENT

The author acknowledges the efforts of the following people who contributed to the design and implementation of the AMRF Vertical Machining Workstation software and hardware: Clara Bird, Robert Bunch, J. Jun, Tom Kramer (Catholic University), Denver Lovett, Ed Magrab, Fred Rudder, Alton Quist (General Dynamics - Pomona), Wendell Wallace, and Carl Wenger.

## REFERENCES

- (1) Magrab, E., "Vertical Machining Workstation of the AMRF: Equipment Integration", (Submitted for publication).
- (2) Simpson, J., Hocken, R., and Albus, J., "The Automated Manufacturing Research Facility of the National Bureau of Standards", Journal of Manufacturing Systems, Volume 1, pp 17-32.
- (3) Hocken, R., Nanzetta, P., "Research in Automated Manufacturing at NBS", Manufacturing Engineering, Vol. 91, No. 4, October 1983, pp. 68-69.

(4) Nanzetta, P., "Update: NBS Research Facility Addresses Problems in Setups for Small Batch Manufacturing," Industrial Engineering, June 1984, pp. 68-73.

(5) Furlani, C., et al., "The Automated Manufacturing Research Facility of the National Bureau of Standards", Proc. of the Summer Simulation Conference, Vancouver, BC, Canada, July 11-13, 1983.

(6) McLean, C., Mitchell, M., Barkmeyer, E., "A Computing Architecture for Small Batch Manufacturing," IEEE Spectrum, May 1983, pp. 59-64.

(7) Jones, A., McLean, C. "A Proposed Hierarchical Control Model for Automated Manufacturing Systems", Journal of Manufacturing Systems, Vol. 5 No. 1, 1986, pp. 15-25.

(8) Albus, J., Barbera, A., and Nagel, R., "Theory and Practice of Hierarchical Control", Proceedings of 23rd IEEE Computer Society International Conference, September 1981.

(9) Barbera, A., Fitzgerald, M. and Albus, J., "Concepts for a Real-Time Sensory-Interactive Control System Architecture", Proc. of the 14th Southeastern Symposium on System Theory, April 1982.

(10) McLean, C., Bloom, H., Hopp, T. "The Virtual Manufacturing Cell", Proc. of Fourth IFAC/IFIP Conference on Information Control Problems in Manufacturing Technology, Gaithersburg, MD, October 1982.

(11) Jones, A. and McLean, C., "A Cell Control System for the AMRF", Proc. of 1984 ASME International Computers in Engineering Conference, Las Vegas, Nevada, 12-15 August 1984.

(12) Jones, A. and McLean, C. "A Production Control Module for the AMRF", Proc. of Summer 1985 ASME Conference, Boston, Massachusetts, August 1985.

(13) Kramer, Thomas R., "Integrated Automated Manufacture of Metal Parts", Proceedings of National Aerospace & Electronics Conference, IEEE, Dayton, Ohio, May 1986.

This is to certify that the article written above was prepared by United States Government employees as part of their official duties and is therefore a work of the U.S. Government and not subject to copyright.

The identification of software and hardware systems was done to adequately specify the system's capabilities. Such identification does not imply recommendation or endorsement by the National Bureau of Standards, nor does it imply that the systems identified are necessarily the best available for the purpose.