THESIS REPORT
Master's Degree

SYSTEMS
RESEARCH
CENTER

Supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), Industry and the University

Development of a CAM System for the Rapid Prototyping Environment

by: D.C. Hudson
Advisor: J.A. Kirk

M.S. 89-1
Formerly TR 89-3

Development of a CAM System for the Rapid Prototyping Environment

by

Daniel Cory Hudson

Thesis submitted to the Faculty of the Graduate School
of The University of Maryland in partial fulfillment
of the requirements for the degree of
Masters of Science
1988

Advisory Committee:

Professor Dr. James A. Kirk, Chairmen/Advisor
Professor Dr. Davinder K. Anand
Professor Dr. Edward B. Magrab

Abstract

Title of Thesis: *Development of a CAM System for the Rapid Prototyping Environment*

Name of Candidate: Daniel Cory Hudson, Masters of Science, 1988

Thesis directed by: Dr. James A. Kirk, Professor, Department of Mechanical Engineering, University of Maryland, College Park

The goal of this research was to develop a Computer-Aided Manufacturing system for rotational parts for application to small-scale manufacturing. The developed software, DPCAM, was to provide an intermediate step between the total automation of a Rapid Prototyping Center and current commercially available systems. This was achieved by applying technologies already developed for the total automation problem to the intermediate solution. Modules included in the DPCAM package include:

-Design Retrieval

-Part Evaluation

-Estimation

-N/C Code Generation

-Equipment Management

The information from the modules is presented to the user through a series of "user friendly" graphics windows. While each of these functions is a "state of the

art" system, incorporating techniques such as Group Technology and Manufacturability Analysis, the major contribution is their integration into a single complete CAM package. The finished system is implemented on a "PC" style computer and tested at the University of Maryland's Flexible Manufacturing Laboratory.

Acknowledgements

I wish to thank Dr. Kirk for his help throughout this project. Without the guidance both he and Dr. Anand provided, this research would not have been possible. I would like to thank Dr. Magrab for his assistance and for sitting on my Thesis defense committee. I wish to express my gratitude to the Systems Research Center for the funding and support they provided for this research.

Thanks also goes to all members of the Flexible Manufacturing Laboratory, Magnetic Bearing Laboratory, Magnetic Spindle Laboratory and Space Station Laboratory for their help, suggestions and support throughout the development of DPCAM.

Special thanks to Dachen Pang, my cohort in the development of DPCAM, and Lisa Wilson, for her help in the preparation of the thesis text.

Table of Contents

List of Figures

List of Tables

Chapter One: Introduction

What is manufacturing? Websters defines manufacturing as:

> ...making of goods by machinery...

From this we can infer that both skilled craftsmen creating individual custom products and groups of robots and men working to make millions of like items are performing a manufacturing process. Since man first formed a bowl to hold his food or sharpened a spear to defend his land, he has sought better ways to manufacture the goods that surround and fill his life.

The goal of this thesis is to improve productivity in one area of manufacturing. This area, known as Rapid Prototyping, deals with the automated production of components in the small batch setting. The work combines advances made in manufacturing practice and computer technology to create a state-of-the-art CAM system called DPCAM. Included in this package are functions for: Design Retrieval, Part Evaluation, Estimation, N/C Code Generation and Equipment Management.

Short History of Manufacturing and Automation

Since the dawn of time, man has sought to supplement his own strength by using human ingenuity, creating mechanisms to perform jobs otherwise done by manual labor. To construct these machines, our ancestors developed basic manufacturing skills, such as cutting and forming. As years passed, succeeding generations refined old techniques and developed new ones. Still, production primarily consisted of a single craftsmen, working on an individual piece from start to finish; not what is looked upon as manufacturing today.[2] All of this began to change rapidly, however, in the 18th century.

A series of advances in agriculture techniques throughout the 18th century resulted in heretofore unknown food surplus'. With extra food, populations and aggressions both grew as nations jostled for land and power. To meet the rising tides of aggression, governments pushed their craftsmen for more and more wares of war; weapons, uniforms, ships, etc. In an attempt to satisfy the increasing demand for goods, radically new manufacturing techniques were developed. Chief among these developments, in what is now called the Industrial Revolution, were the beginnings of automation and interchangeablity.[31]

The development of Interchangeability was a key step in moving production out of the hands of a single individual and into the multitudes of the transfer line. Up to this point, as long as the final product fulfilled the objective of the original design it was considered acceptable. With the advent of interchangeable parts, however, a product had to be made exactly to design specifications. This insures that pieces from other craftsmen or shops could be used in later design changes or repairs. In addition, it became apparent that to best control these tight standards, parts should to be made in large centralized locations; thus the beginnings of todays factories.[2]

The next major evolution in manufacturing came 150 years after the Industrial Revolution and is continuing today: Manufacturing Automation. The term "Automation" was coined in the 1950's at Ford Motor Co. It described the new practice of automatically transporting material to different locations on the shop floor.[6]

Modern Automation was first proposed in the 1950's. Various interests, both commercial and governmental, sought to reduce human involvement in areas of manufacturing by replacing man with machines. It was theorized that machines could perform a variety of tasks with a better factor of "repeatability" then man. They could also take care of new jobs too hazardous for manual labor. To meet this need, the US Air Force contracted the Massachusetts Institute of Technology to research the problem. In 1952 they returned with a demonstration of the first numerically controlled "programmable" machines. Since then, advances in computers and robotics have combined with the NC machine to form what is today considered modern automated manufacturing.[25]

The Scale of Production and the Rapid Prototyping Center

The techniques employed to make a part are directly related to the projected volume of it's production. (figure 1) If millions of a particular item are to be produced, a transfer line or special system is created to make and assemble the pieces. These techniques involve the formation of a set of machines and men devoted solely toward making that one item. The initial

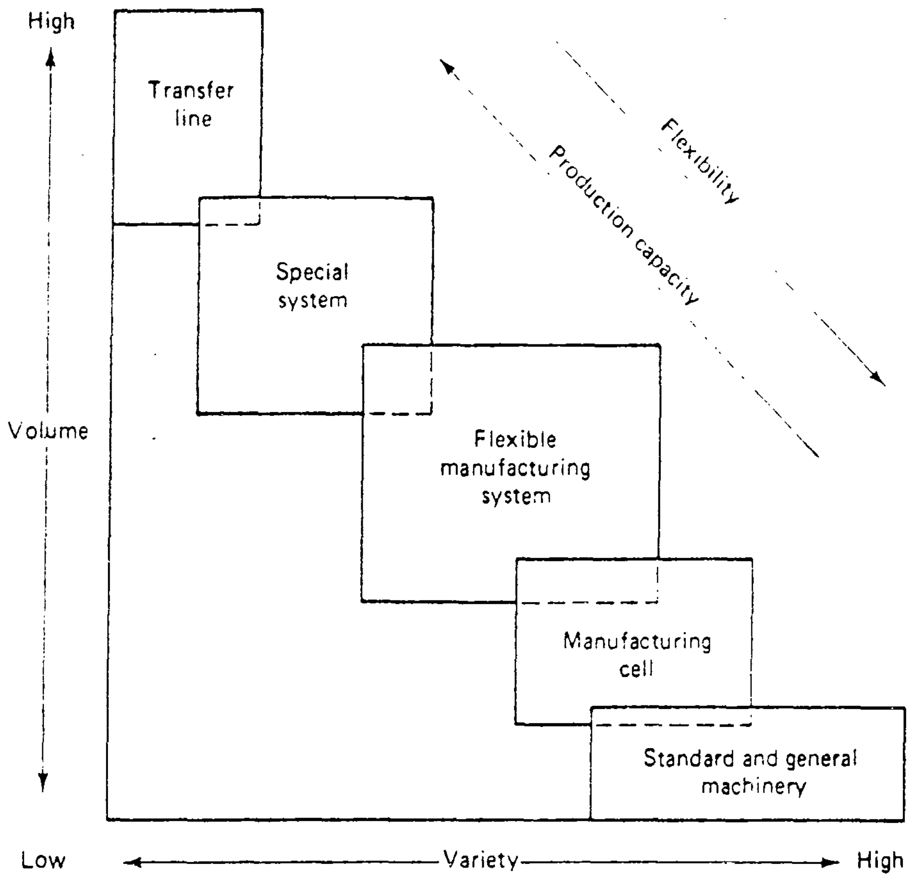


Figure 1: The Scale of Production [2]

investment on a transfer line or special system is high, but in the long run, these dedicated processes make both higher quality and less expensive products.

If an item needs to be produced on a large scale, but not on that of a transfer line, a Flexible Manufacturing System or Manufacturing Cell is commonly used. These techniques use the same type of automated equipment as the transfer line (NC Machines and Robots) but do so on a much smaller scale and incorporate more flexibility. The major drawback to these methods is the high cost of the changeover needed to make each different part. This expense is due to the level of knowledge needed to communicate with and reconfigure today's automated equipment. To use a Flexible Manufacturing Cell, enough parts need to be made to offset these costs and still make a profit.

When an item's production level dips below that where a Flexible Manufacturing Cell is still profitable, it is made in an individual or "small batch" fashion on general machinery in a machine shop. Although NC equipment is beginning to appear in these shops, the techniques used are little different than those used in the 1940's and 50's. This fact take on an even darker hue when it is shown that up to 75% of production in the United States today takes place in the small batch arena.[33]

To bring the machine shop up to the current level of technology, a new type of shop is proposed. It encourages a migration of the automation techniques used in the Flexible Manufacturing Cell and the Transfer Line to that of small batch production.[18] It is called the Rapid Prototyping Center (RPC). (figure 2)

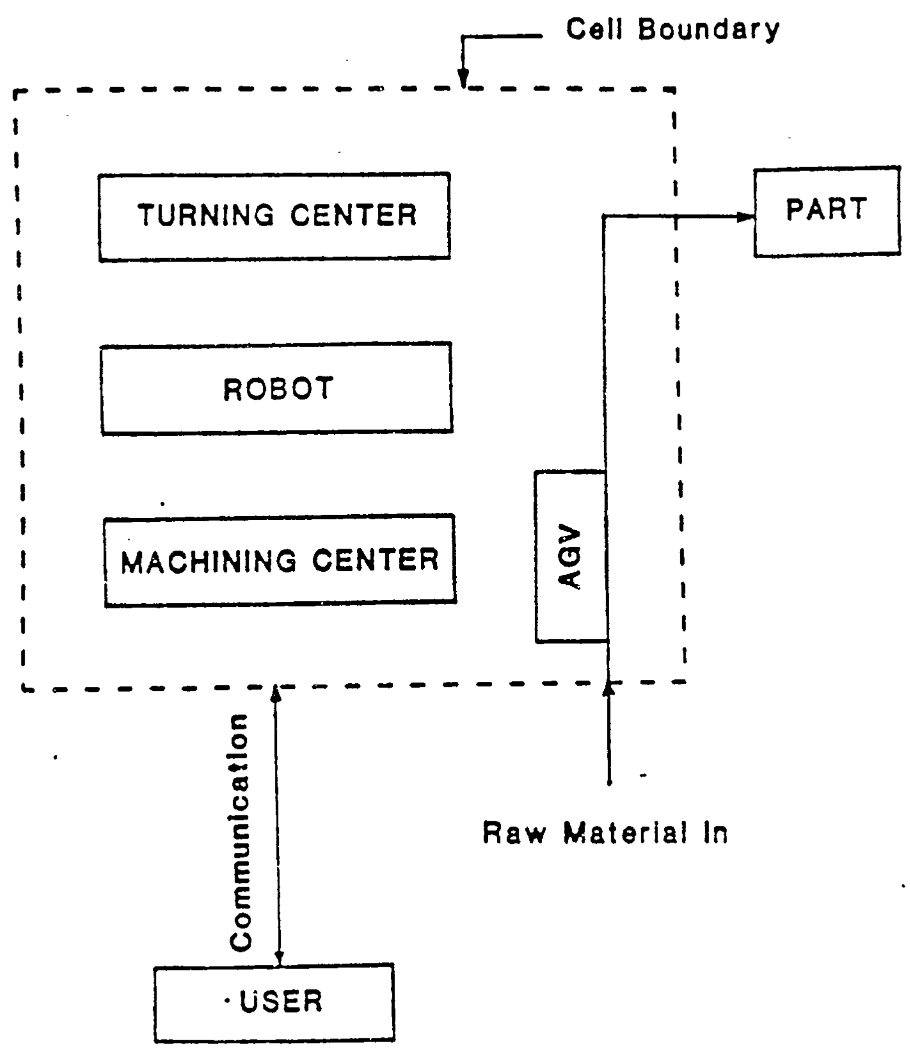


Figure 2: The Rapid Prototyping Center

Research Toward the Rapid Prototyping Center

The goal of the Rapid Prototyping Center is to make use of programmable machines, robots and other modern devices, normally found on the shop floor of a large factory, as common in the average machine shop.[15] As stated earlier, one of the main, obstacles toward this goal is the expensive changeover of both software (machine and robot programming) and hardware (fixtures) needed each time a new part is made. To lower these costs, it has been proposed that the changeover process should also be automated. This technique is embodied in the Flexible Manufacturing Protocol (FMP) being developed at the University of Maryland.[3,33]

FMP embraces the idea that existing technologies, such as Computer Aided Design (CAD), Computer-Aided Manufacturing (CAM), Computer Aided Process Planning (CAPP) and the Flexible Manufacturing Cell (FMC) should all be joined into a single unified process. (figure 3) Only in this way will the costs of the needed flexibility be lowered to a reasonable level. The FMP can be divided into five major areas:

-Design
-Manufacturability
-Process Planning
-NC Code Generation
-Cell Control

While great strides have been made in each of these areas, a complete working model has yet to be developed. But just because the FMP is not ready does not mean that the machinist has to go unaided.

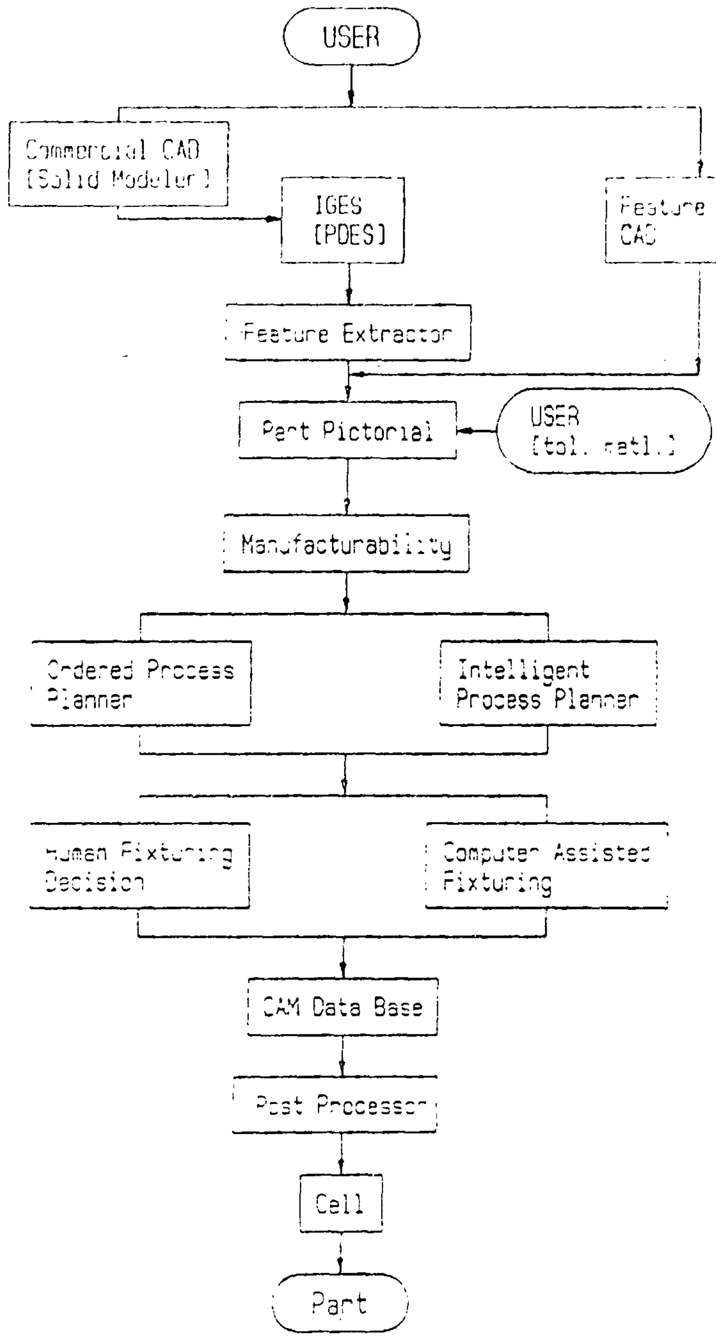


Figure 3: Flexible Manufacturing Protocol

Current Tools Available Toward the RPC

Today's machinist finds himself flooded with a bevy of products to help him begin the journey down the path toward the RPC. Most are marketed under the moniker "CAM system" but a closer analysis reveals them to be much less. State-of-the-art commercially available CAM systems, like Point Control's SmartCAM, provide the user with a powerful NC code generation facility, but little else. In order to get help at the desk, a machinist would need to use a totally different program dedicated to the appropriate task. An example is Manufacturers Technologies' Costimator, used for cost estimating. For another task, for instance, fixturing, yet another piece of software, i.e. Fixture Pro by Jergins, Inc. would need to be used.[10,11,12]

In short, to come up with a system approaching what could be considered true Computer-Aided Manufacturing the user is required to employ a multitude of individual, incompatible, programs.[11,12] As one could imagine, many machinists throw up their arms in frustration at this muddled patchwork of programs. The result is a system of machine shops lagging behind the times because the complete system is not yet available and current alternatives do not meet expectations.

Alternate Route Toward the RPC

With FMP and other like systems years away from operational status and a current system that is an unfinished software "patchwork quilt" at best, it would appear that the flexible Rapid Prototyping Center will have to wait. This would not be so if available technology and advances already made in FMP research were combined into an intermediate CAM system useful to the machinist now. Such is proposed and implemented in this research under the name DPCAM.

Chapter Two: The Structure of DPCAM

The ultimate goal for the DPCAM software is to provide a machinist with an intermediate solution to the problem of total shop automation for the RPC. To achieve this, DPCAM must be a state-of-the-art CAM system; providing a more complete product then current, commercially available systems.

Objectives of a CAM system

In general. Computer-Aided Manufacturing can be defined as:

> ...the use of computer systems to plan, manage and control the operations of a manufacturing plant...[8]

As such. CAM application fall into one of two categories:

1) Computer Monitoring and Control (CMC) - These are functions where the computers exercise direct control or monitoring over a process.

2) Manufacturing Support Applications (MSA) - These are functions necessary for production, but where no direct contact between computer and process occur.

Computer Monitoring and Control areas are those where the computer exercises a degree of direct control over the manufacturing process.[8] This can be all-encompassed by one broad field of study called Cell Control. In a CAM system, cell control takes as input "part orders". These come from upstream in the system from a process planning unit. The Cell Controller then organizes them into the required sets of operations to satisfy the system's requirements. Once the appropriate grouping is achieved, the controller directs the various floor devices available into a coordinated symphony of action to complete the ordered part. Likely contents of a "part order" would be:

1) Process Plan--Contains material requirements, tool requirements, job priority and part destination.

2) Machine Codes--NC tool paths used to drive the machine tools.

If the Computer Monitoring and Control area of a CAM system requires process plans and machine codes as input, it is only logical that this type of information be part of the Manufacturing Support Application's output.[8] As CMC provides for assistance on the shop floor, MSA aids the machinist in "paper" oriented tasks. Typical MSA functions include:

NC Part Programming--control programs for NC machine tools

Computer-Aided Process Planning--creation of the machine sequences for a particular part.

Computer-Generated Work Standards--generation of production time for a particular part or operation.

Input to the MSA section of a CAM system commonly comes in the form of an IGES file from a linked CAD system or directly from blueprint information.

From this breakdown of CAM into CMC and MSA operations, one can concluded that a truly comprehensive CAM system should seek to span from the office to the shop floor. Aid should be available in such diverse areas as cost estimation and machine control. No single current CAM system can provide all of these functions.

Objectives of DPCAM

As seen in the previous section, developing a complete CAM system is a formidable task, requiring a knowledge of both office and floor operations to provide a truly comprehensive system. In laying out the objectives of DPCAM, we will attempt just that.

The first thing most people think of when they consider "manufacturing" is the physical fabrication of a part--the cutting of metal. As a result, it is no surprise that virtually every CAM system available today supports the creation of NC machine cutting codes. To insure at least parity with existing systems, DPCAM must, as a bare minimum, include provisions to create cutting codes.

The term "do not re-invent the wheel" is a common cliche in the English language. It indicates the propensity of man to recreate past discoveries because of poor management and bookkeeping or lack of willingness to spend time examining past work. Great amounts of time and money could be saved if people performed even a short search of prior achievements before setting off on their own.[1] One of the major areas of research in the field of Computer-Aided Process Planning is the use of a computer's data management abilities to store previously created processes for future use. Should a similar part ever need to be made, the old process plan can be resurrected and retraced steps saved. This methodology is known as the Variant Approach and applies a grouping principle called Group Technology to link "like" families of parts together.[2] An offshoot of this work, known as Design Retrieval, would be a useful addition to the DPCAM system.[7]

A key element in the manufacture of any part is the creation of a cost estimate. Before any contractor makes a commitment to a particular company or shop, he wants to know what their "bottom line" price is. As a result, whether a shop survives or fails hinges on their ability to create timely and accurate cost estimates.[21] If the part desired is of a "straight forward" nature, estimation is a routine exercise. Unfortunately, this is not often the case. Today's machinist is confronted with parts combining unusual and complex geometries with stiff tolerances, and material requirements encompassing the latest space-age composites. In these situations, the average estimate is more akin to a "stab in the dark" then a scientific calculation.[21] Such a poor guess could result in either a gross underestimate and financial disaster or an overestimate and the unnecessary loss of a job. To make matters worse, a varying range of experience in machinists at a shop, usually results in a corresponding inconsistency in the estimates generated. To provide a solution to all of these problems, Computer-Aided Cost Estimating (CACE) has been developed. These programs provide both a "shining light" down paths that a machinist has yet to traverse and a common reference that results in a more uniform series of estimates irregardless of the estimator.[10] CACE is a key element of support in any CAM system and should be included in DPCAM.

Among the numerous developments for the FMP is a module that deals with the "Manufacturability" of a part. Items considered in manufacturability are tolerances, feature irregularities and requests to the manufacturing cell that are not possible with current equipment.[5] It would be helpful to the machinist if a module containing some of these features were incorporated into the DPCAM system. A Part Evaluator could interact with the user to provide information on the "manufacturabilty" of a part. It could also provide assistance in making the decision whether a part is producible given the machinist's skill level and the available equipment.

One of the main problems of current CAM software packages is their lack of flexibility. In many cases, the moment a new piece of equipment is brought into the machine shop, the company's CAM package becomes obsolete, unable to accommodate the needs of the new device. As such, a goal of DPCAM should be to remain as flexible as possible, allowing for changes in shop equipment. One way to achieve this is through the incorporation of lists of shop equipment into an editable database management system. Such a system might be a wise addition to the DPCAM package.

Maybe the most important element of any software package is the issue of user friendliness. No matter how complex or elegant the internal operation of a piece of software is, unless it provides the user with a tractable interface, it will go unused. Graphics have proved themselves to be one of the most effective ways to communicate information to a user (Remember "a picture is worth a thousand words!").[13] The addition of color makes graphics even more effective. Another issue to consider is the intense dislike many users have toward looking through thick manuals to figure out a program's mode of operation. With this in mind, a measure of "on-line" documentation and error messaging should be included to help the user move more quickly through the program. All of these factors should be kept in mind when creating the DPCAM user interface.

Conclusions

The philosophies of DPCAM have been formulated and laid out. (figure 4) The completed system attempts to join current CAM technologies to new "state-of-the-art" ideas developed for the yet to be completed Rapid Prototyping Cell. The completed package will include modules for the functions:

-Design Retrieval
-Part Evaluation
-Estimation
-NC Code Generation and Transmission
-Database Management

and will be incorporated under a system of user-friendly graphics interfaces.

One final note. To complete work on such a large project in a short period of time, the research has been divided into two sections: Prismatic parts

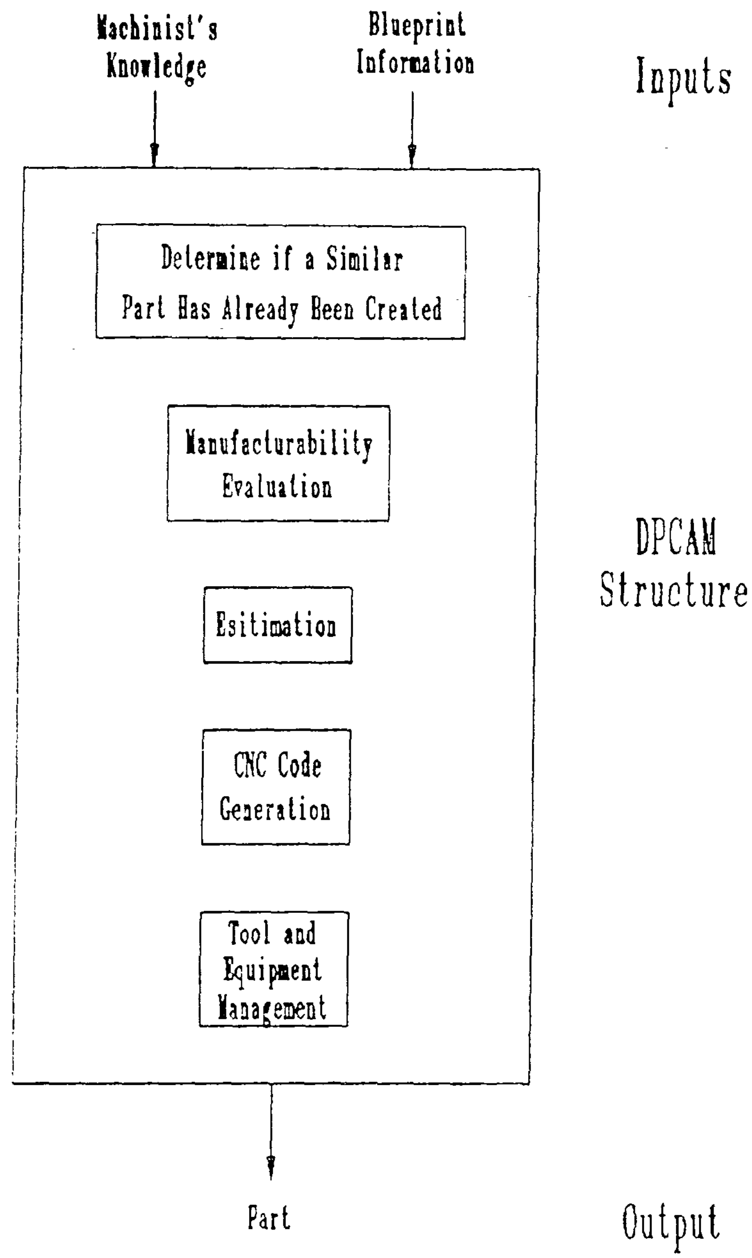


Figure 4: Layout of DPCAM

and Axi-Symmetric parts. This thesis will discuss strategies and solutions for Axi-Symmetric components. Solutions for the problems for prismatic parts are presented in a thesis by Dachen Pang.[22]

Chapter Three: Design Retrieval Through Group Technology

The major stumbling block in the path to a truly automated manufacturing center is the need for flexibility. Today, the cost of flexibility is a slow system; the result of reordering the cell for each different part that comes along. With this type of setup it is inevitable that the same items and procedures will be recreated; in affect "reinventing the wheel" again and again. This waste of time and material could be reduced if a mechanism was in place that would allow a machinist access to previously created plans and estimates applicable to a current job. Such a Design Retrieval system is currently available in a new manufacturing philosophy called Group Technology.

Group Technology

Group Technology is a manufacturing philosophy in which similar parts are identified and grouped together to take advantage of their similarities in manufacturing and design.[8] For manufacturing, Group Technology allows for a more efficient layout of tools and equipment on the shop floor. The benefits of this are increased efficiencies in setup time, inventory, scheduling, process control and planning.[8] Group Technology helps the designer by allowing him easy access to existing drawings and estimates. By searching through the old data, an existing part that could be modified to meet current needs, might be located.[7]

Central to the Group Technology idea is the organization of parts into family groups. These families can be created from either a design or a manufacturing standpoint. A design family is characterized by sets of components that have a host of geometric similarities (figure 5). Manufacturing families are comprised of a collection of parts that, though they do not appear to be related, are created through similar sets of manufacturing processes (figure 6). Unfortunately, these families are not necessarily the same, since the requirements of design and manufacture are often different. In response to this limitation there has been an effort to create family groupings that reflect both sets of characteristics.[2] Creation of family groupings is usually carried out in one of three ways:

1) Visual Inspection
2) Production Flow Analysis
3) Part Classification and Coding System

Accepted as the most accurate and the most widely used of the three techniques is Part Classification and Coding.[8]

Part Codes consist of a set of numbers and/or letters which attempt to represent the character of a particular part. The accuracy of a coding system is a function of the methodology used and the code length.[7] The three methods used in part coding are: Hierarchical, Chain and Hybrid.

The Hierarchical technique allows for a detailed representation in a compact code. This is achieved by making the current code digit qualified by the preceding entries.[2] This characteristic makes hierarchical systems take on

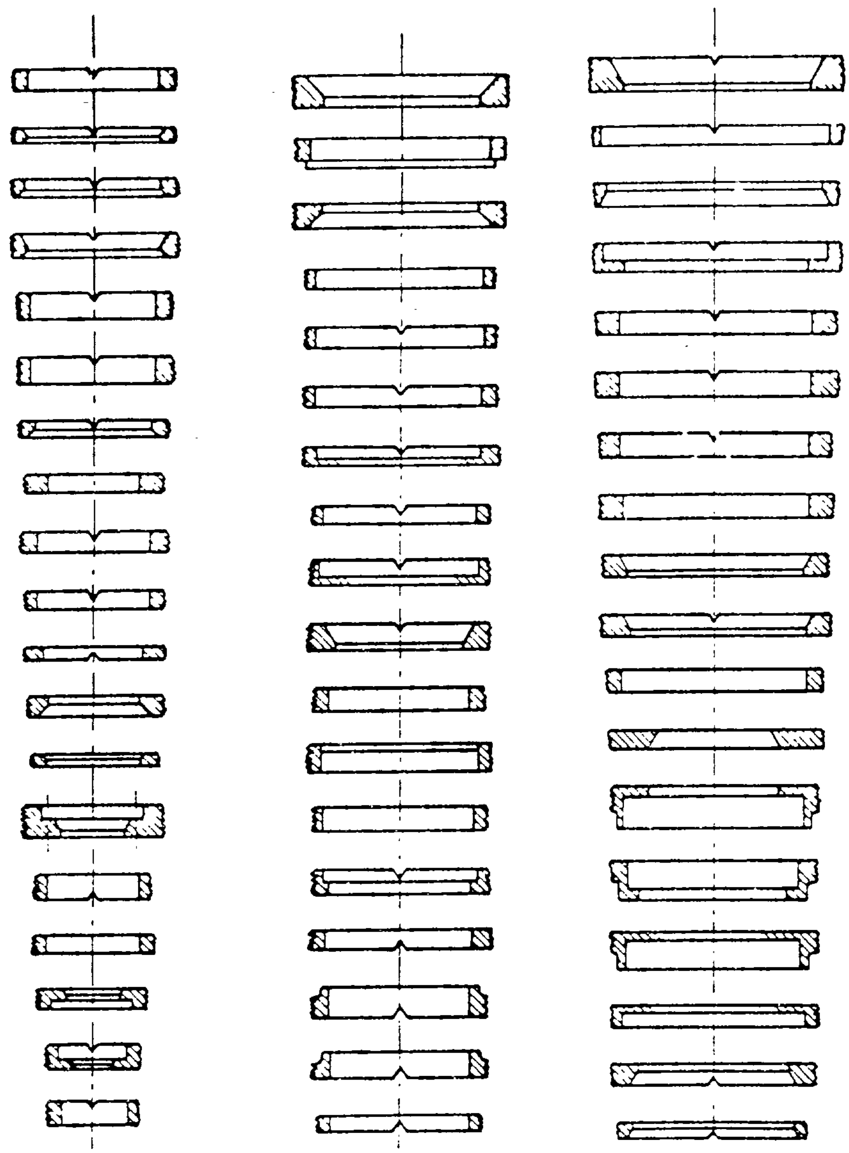


Figure 5: Group Technology; Design Family [2]

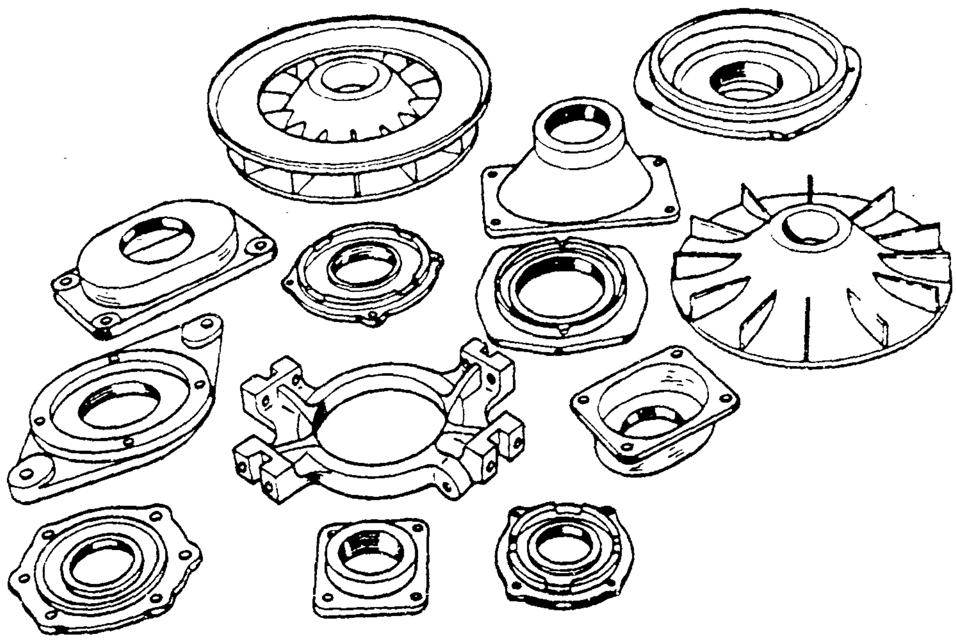


Figure 6: Group Technology; Manufacturing Family [2]

an appearance similar to a tree with many outstretched branches (figure 7). Unfortunately, such codes gain their descriptive power at the cost of complicated interpretation.

At the opposite end of the spectrum is the Chain method. In a chain code, each digit has a specific meaning, regardless of the previous entry's value (figure 8). While this technique provides a code that is easy to interpret, to get the necessary level of detail requires a large number of code digits.[2]

The most popular of the three techniques is a system that is essentially a marriage between the two previously described methods. The Hybrid scheme employs a hierarchical method in areas of the code that require a detailed description. Less critical characteristics are described using the chain technique (figure 9). This combination provides the user with a code giving sufficient detail in a compact form that is still easy to interpret.[7]

Application of Group Technology to DPCAM: Opitz System

In searching for a Group Technology coding system for use in DPCAM, two minimum requirements were established. First, the system must incorporate both design and manufacturing characteristics so it is useful as a retrieval tool for the RPC. Also it must lend itself well to integration into an user-interactive system with a graphical interface. A survey of current schemes pointed to the Opitz system, developed by H. Opitz of the University of Aachen, West Germany, as best suited for integration into DPCAM. (figure 10)

The Opitz system uses a hybrid technique to create a two part, nine-digit code describing both design and manufacturing attributes.[19,20] The first five digits reflect the form or geometry of the part while the latter four digits give

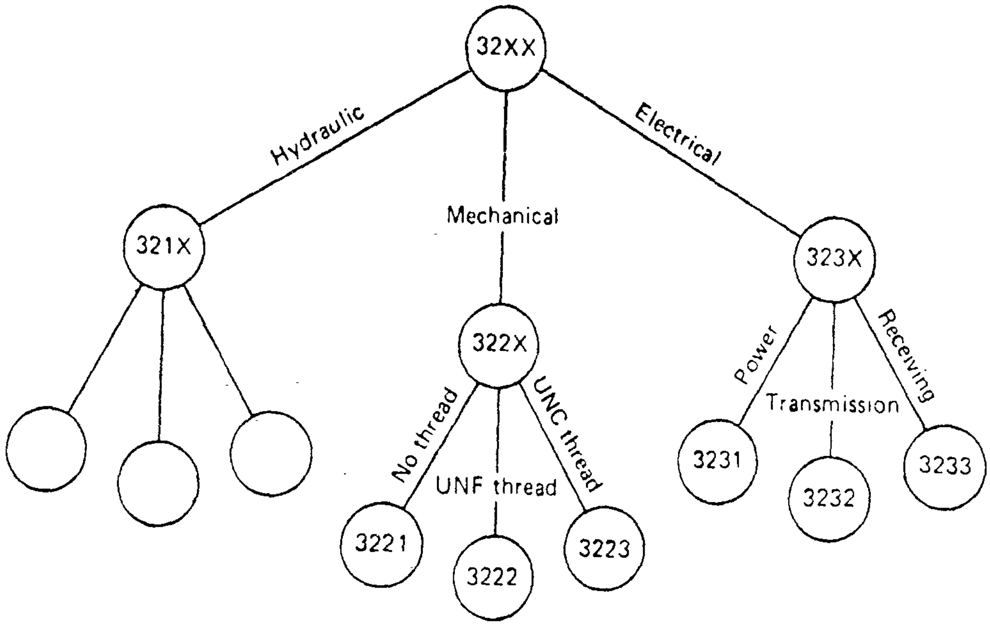


Figure 7: Hierarchical Code [2]

Digit position	1	2	3	4
Class of feature	External shape	Internal shape	Holes	
Possible value 1	Shape 1	Shape 1	Axial	
2	Shape 2	Shape 2	Cross	
3	Shape 3	Shape 3	Axial and cross	
4				

Figure 8: Chain Code [2]

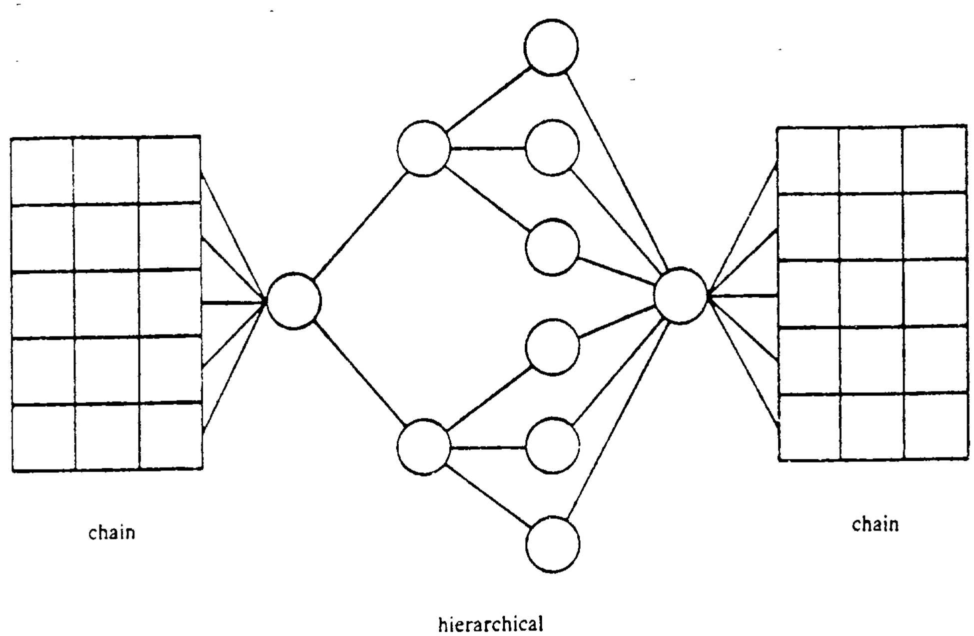


Figure 9: Hybrid Code [2]

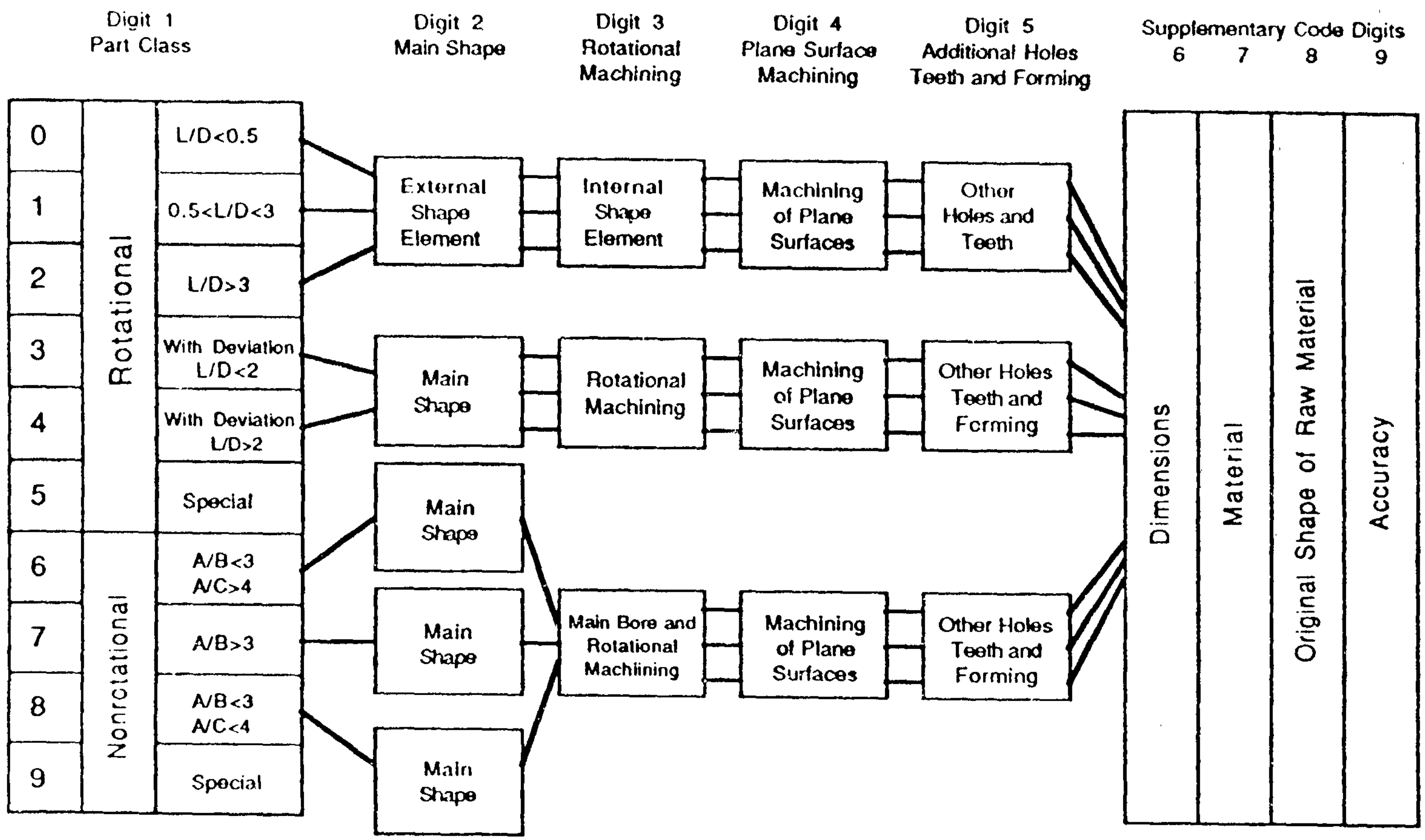


Figure 10: Opitz System [19]

information such as stock shape and material. An unanticipated advantage within the structure of the Opitz system was found in the layout of each coding digit. As each digit's value increases, a more complex configuration is indicated; a characteristic with important ramifications toward the development of the Part Evaluator (see chapter 4).[7]

The Opitz system is incorporated into DPCAM through the use of windowed graphical representations; each reflecting a single possible Opitz configuration. When the user begins to work with a new part, he is steered into the Opitz module. The code is arrived at by choosing from among the various sets of drawings, ones that best reflect the character of the part--it's internal and external shapes, any plane surface machining, etc.[1] Since DPCAM only uses the first section of the Opitz code, the output of the module is a five digit number.

To initiate Design Retrieval, a database of old parts needs to be created, which includes basic part information. In DPCAM, this Old-Part Database is automatically created and updated each time a new set of part data is entered for processing. It's entries include, part name, a short descriptive note and the Opitz code. The Opitz code provides the information on the part's configuration and manufacture needed to drive a search while the other entries include useful details on where more about the part might be found. To generate a list of parts that might provide useful information for the current application, a new item's Opitz code is compared to previous ones found in the Old-Part Database. Any matches are flagged as possible candidates for retrieval. This abbreviated list of old parts is then made available to the user for further investigation.

Conclusions

The Design Retrieval function of DPCAM applies Group Technology as the mechanism to retrieve previously created designs and estimates. The Group Technology code chosen is the Opitz system which uses a nine-digit code to describe both design and manufacturing characteristics. The Opitz code also has an ascending digit--increased complexity relationship that will be helpful in Part Evaluation. Applicable designs are retrieved by comparing a current part's Opitz code to ones of those previously created. A list of parts with matching codes is provided to the user for further investigation. (figure 11)

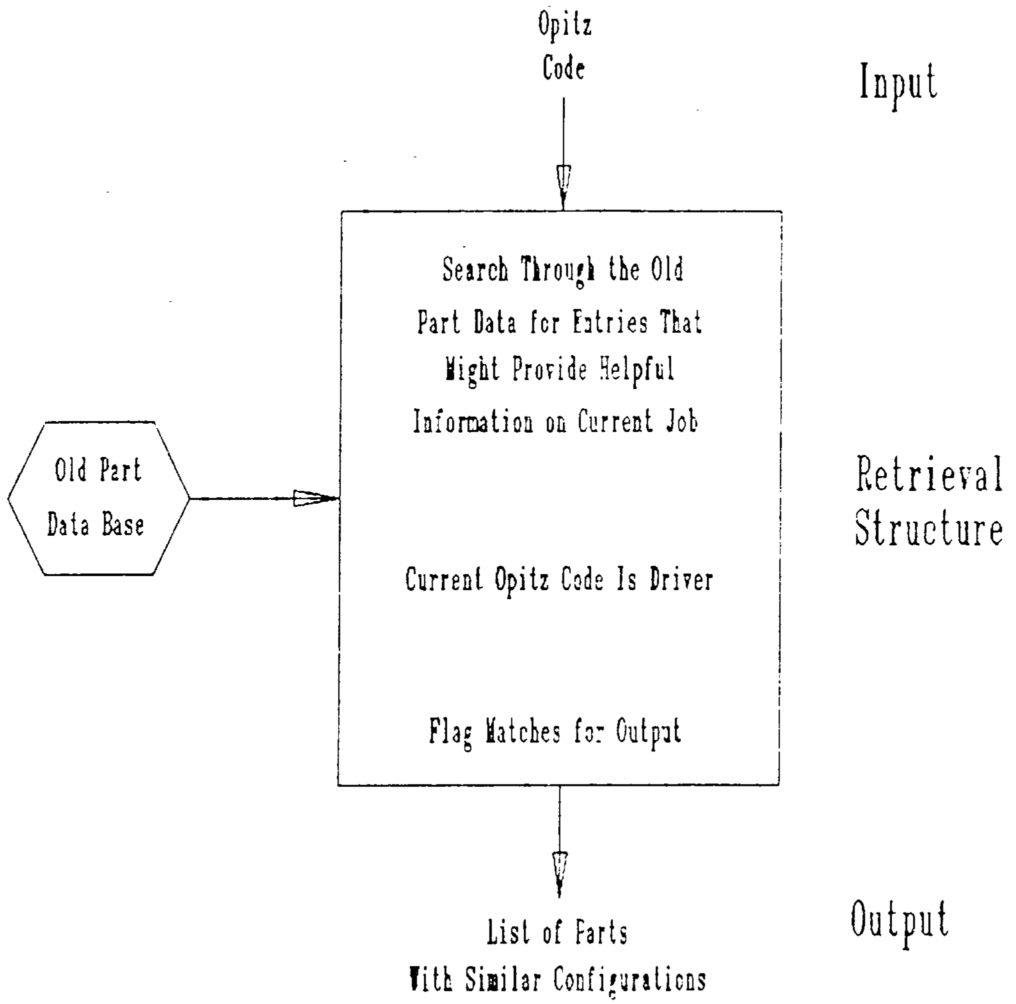


Figure 11: Diagram of the Retrieval Process

Chapter Four: Part Evaluation for a Machining Index

Before a machinist begins to remove metal from a piece of stock, he has already formulated each step of the machining process in his mind. The creation of this process plan is one of the key elements in the production of a part. It is also one of the elements incomplete within the Flexible Manufacturing Protocol. If it is not yet possible to automate the planning process, what aid can be given to the machinist now to more easily create better plans? One way is through the calculation of a Machining Index. Such an index, ranging from 0=easy to 10=hard, would reflect the complexity or "manufacturability" of a part.

Development of a Machining Index

In creating a process plan, a machinist is influenced by many factors: geometry, tolerances, materials, fixturing; the list goes on. The demands required to meet these criteria dictate the route the machinist takes through the manufacturing process. This is illustrated by examining a series of holes, each with it's own individual geometric requirements (figure 12). A simple bore with a flat bottom is more difficult than one in which a conical bottom is permitted. This is a by-product of the geometric configuration of the average drill bit,

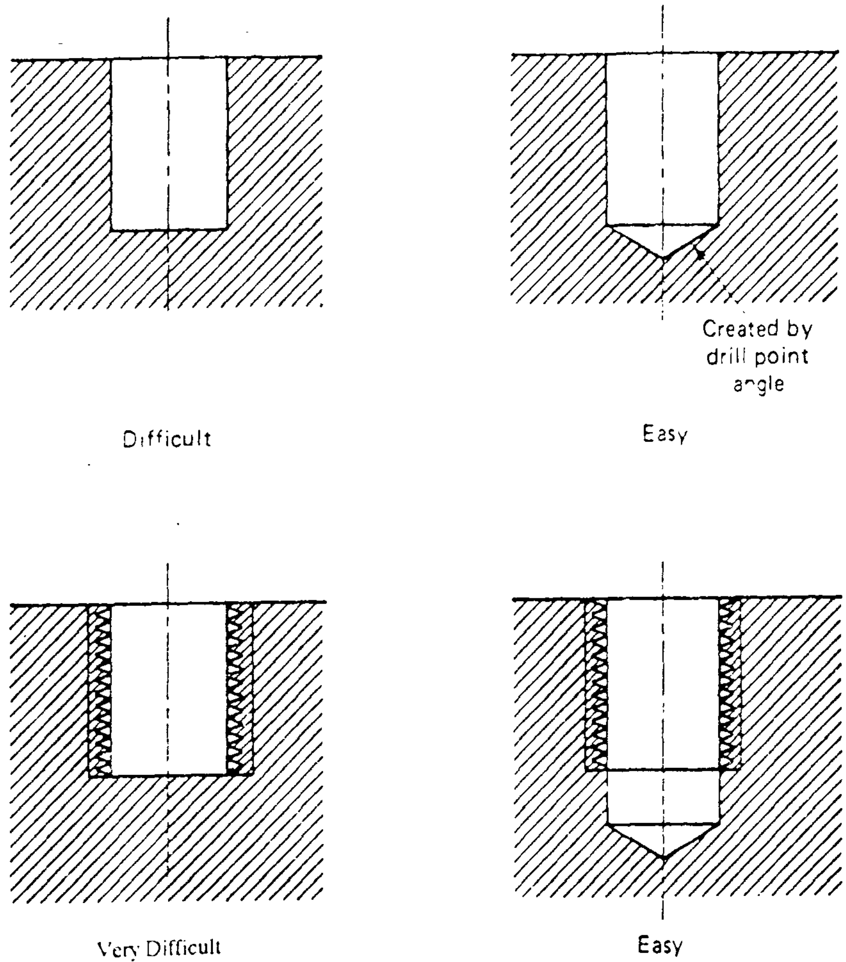


Figure 12: Illustration of Manufacturability [2]

which has a conical cutting surface. If threads are added to each hole, continuing them to the bottom of the flat-bottomed bore is virtually impossible. This is due to the lack of tool clearance at the bottom with this geometry. The same does not hold true for the conically-bottomed hole, since it's geometry allows for the needed tool clearance.[2] Thus, it is easy to see how design and manufacturing requirements can become so entangled, sometimes being such that manufacture is not attempted at all.[5] In formulating a Machining Index, it is proposed to take a set of machining factors, such as those mentioned above, and codify them into a global index. This index could then provide a yard stick to the machinist indicating the relative complexity of a particular undertaking. By applying this standardized Machining Index, a shop would achieve more uniform process decisions and cost estimates (see Chapter 5).

An examination of turned part manufacture indicates that the primary elements of the machining index are:

-Geometric Shape
-Tolerances
-Surface Finish
-Material

The combined evaluations of these factors should reveal the complexity of a part from the manufacturing point-of-view.[16,28]

Geometry

The geometry of a part is the major influence to a machinist in choosing the ordering of the machining processes. The more complex a part's geometry, the more critical it is that the features be cut in a particular order.[9] Thus it is important to have an accurate description of a part's form for the index. An unexpected advantage found in the Opitz system provides easy access to the geometric information needed. Since Opitz used geometric complexity as the basis for his technique, the needed information is already available in the Code's digit values.

For rotational parts, Opitz broke geometry into five parts:

Type
-without deviation
-with deviation

Rotational Shape
-Internal
-External

Plane Surface Machining

Gear Teeth, Holes and Forming

He then ordered the configurations in each category from least to most difficult. With these relationships as a starting point, a set of "Geometry to Machining Index" relationships are easily arrived at.

Tolerances

Tolerance requirements for a part dictate both process ordering and feed and speed restrictions to a machinist. As an example, a hole needs to be drilled through an angled face. If the tolerances required for the hole are high, then it must be drilled before the angled face is created. If the face is created first, then the hole made, the drill bit will most likely deform slightly when penetrating the angled surface. The result, an improperly placed hole.[9]

To accurately account for the complete effect tolerances impart to the index, they should be divided into two categories: Dimensional Tolerances and Positional Tolerances.[16] Dimensional Tolerances are those which pertain to the accuracy of each cut: Is the shaft truly "circular"?, Is a 3/4" bore actually 0.75" in diameter?, etc. Positional Tolerance reflect the needed accuracy in the area of feature placement. The placement of features is important in both their relationship to the work piece and each other.[6]

To incorporate this information into the Machining Index, a relation can be developed based on the order of magnitude of each type of tolerance.

Surface Finish

The quality of a part's surface finish is dependent on tool and speed and feed. Thus the quality of finish can be improved by varying these factors. Unfortunately, there is an upper limit to the finish possible for a particular process.[35] If a part needs a finish smoother then possible in the main machining process. an alternate process, like polishing, must be performed. (figure 13)

A relationship can be developed between surface finish and the Machining Index by examining the level of finish possible by the common machining methods, milling and boring. The information found in the *Machining Data Handbook* is in a form that lends itself well to this development.

Material

The type of material being cut can cause a machinist to modify every facet of the manufacturing process. This influence is a factor of a material's "machinability".[6] The machinabilty of various materials can be found in a

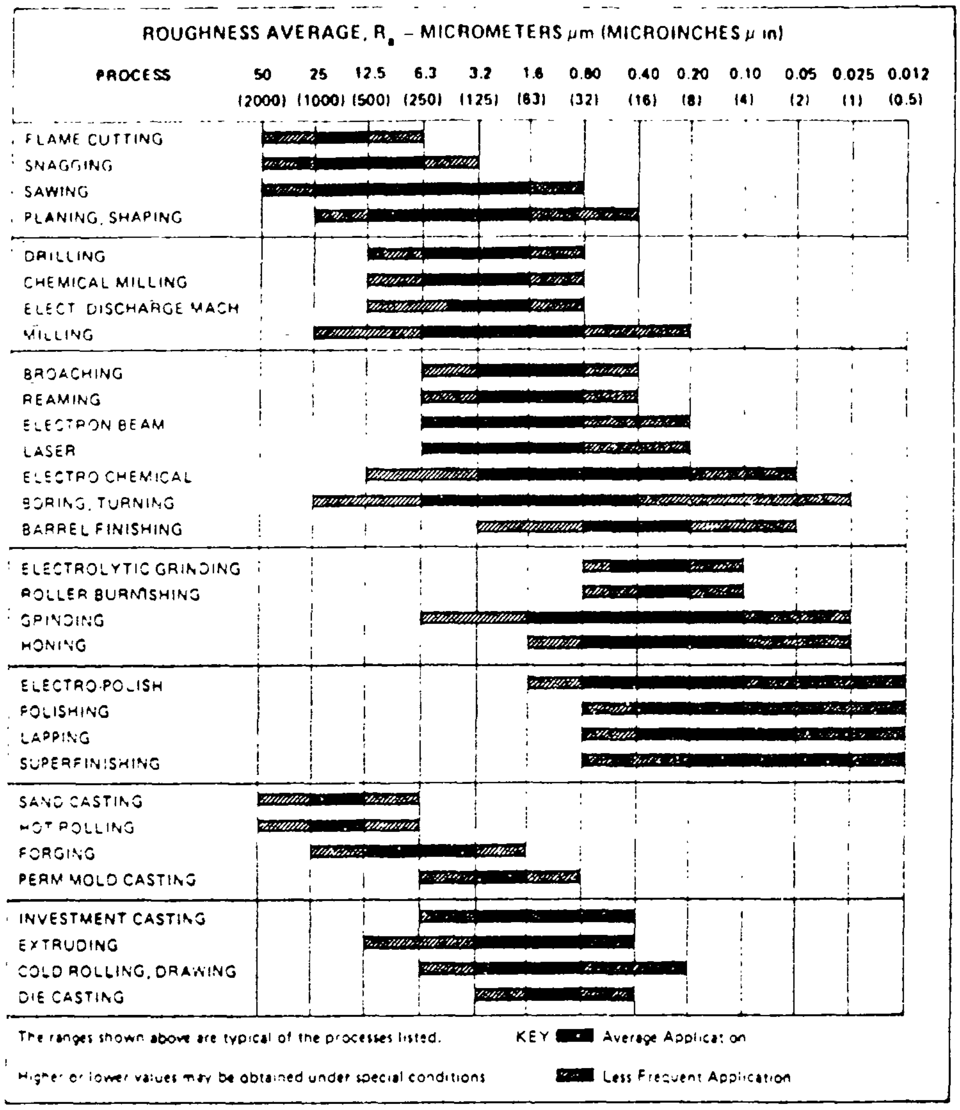


Figure 13: Surface Finish Data [6]

multitude of standard tables. For DPCAM, the information is obtained from the Machinability Ratings Tables of the Independent Research Committee on Cutting Fluids. (table 1) This data is then incorporated into the Machining Index.

Combining the Factors for a Machining Index

To determine a part's Machining Index, a user would enter the proper value into each of the previously described Index sub-elements (see Appendix B). The key to creating an accurate Machining Index is linking the individual elements in such a way that it reflects their proper contribution to the whole.[28] Work by H.G. Smart and D. Kochan indicates that part geometry and tolerances are the driving factors for determining the manufacturability of rotational parts. These facts aside, the only way to figure in the proper weightings is to compile an extensive study of your or another comparably equipped shop's manufacturing activities. This insures that the calculated Machining Index reflects a part's manufacturability for the shop of interest. Such a study would determine the relative importance of each of the sub-elements, based on the investigated facility's equipment and operational expertise. The initial weightings were obtained from a study of operations at the University of Maryland's Flexible Manufacturing Laboratory (table 2).

Exponential Weighting of Factors

This initial linear weighting and combination of sub-elements used in the Evaluator yields undesirable results. For parts with a low level of complexity, Class I, Ferrous, rating over 70%

Material specs.	C1110	C1117	C1120	C1137	C1016	B1112	A4023	A4119
Hardness (Bhn)	137–166	143–179	143–179	187–229	137–174	179–229	156–207	170–217
Rating	85	85	80	70	70	100	70	70

Class II, Ferrous, rating 50 to 65%

Material specs.	C1141	C1020	C1040	A3120	A4032	A5120	NE8024
Hardness (Bhn)	183–241	137–174	179–229	163–207	170–229	170–212	174–217
Rating	65	65	60	60	65	65	60

Class III, Ferrous, rating 40 to 50%, Class IV, Ferrous, rating under 40%

Material specs.	A1320	A2330	A6120	A2515	Stainless 18-8	H.S.S.
Hardness (Bhn)	170–229	179–229	179–217	179–229		
Rating	50	50	50	30	25	30

Class V, Nonferrous, rating over 100%; Class VI, Nonferrous, rating under 100%

Material specs.	Mg. Alloys	2S Al.	Yellow brass	Mng. bronze	Monel cast metal	Inconel
Rating	500–2000	300–1500	200	40	35	45

Table 1: Material Manufacturability [6]

Geometry	
1. External Shape	20%
2. Internal Shape	20%
3. Plane Surface Machining	10%
4. Auxiliary Holes, Gear Teeth	10%
Process Factors	
5. Dimensional Tolerances	15%
6. Positional Tolerances	10%
7. Surface Finish	5%
8. Material	10%
Total	100%

Table 2: Initial Weighting Distribution

the Machining Index calculated falls in line with expected values. However, when parts that are much more complicated or demanding are evaluated, the Machining Index calculated is deceivingly low. The perpetrator of this inaccuracy is the linear relationship between the sub-elements and the final Index. If all of the sub-elements have similar values, as occurs in simple components, the linear relationship is adequate. However, when one or two of the sub-elements have values far above the others, as with difficult components, the "easy" factors overwhelm the larger values when they are weighted and combined together. Thus the complexities of difficult parts are not properly reflected in the final Index.

The solution to the problem requires the development of a scheme that reflects the dominating nature of complex characteristics. In a "real world" problem, a single difficult factor becomes the focus of attention in considering possible planning alternatives. The more difficult the factor is, the greater it's influence becomes. This behavior can be mimicked in the Evaluator by making the sub-element values used in calculations an exponential reflection of the entered values. Since satisfactory Machining Index's occur with the current setup for simple parts, the linear behavior is preserved up to a sub-element value of "4". Above this, the sub-element index is boosted exponentially by the function $e^{.25x}$ (figure 14). This preprocessing is done for each sub-element and the results weighted together, as before, to compose the Machining Index.

Manual Calculation of a Machining Index

Sometimes the automatic creation of the Machining Index is not appropriate. An example is the case of rotational parts with deviations (ie.

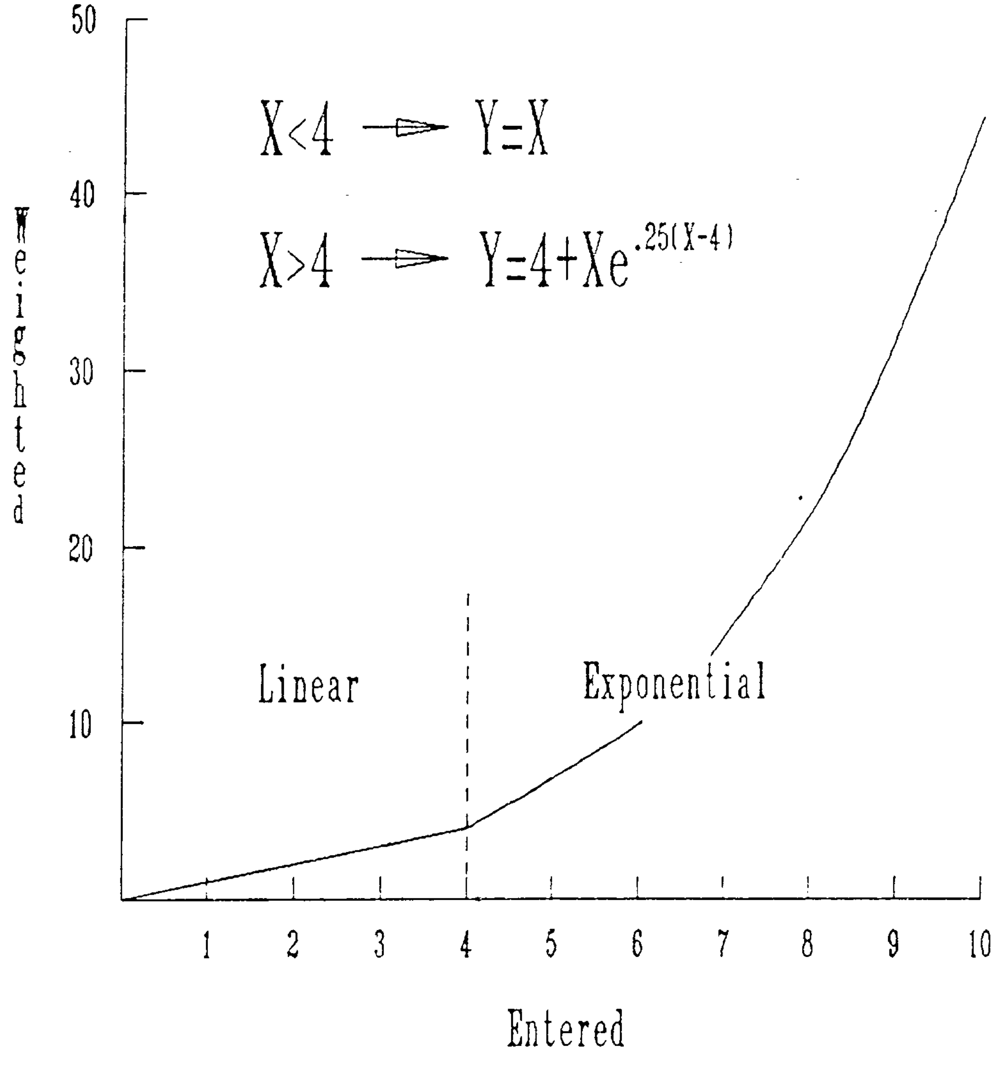


Figure 14: Exponential Weightings

camshafts, segmented rotational parts, etc). The widely varying process and machines used to create these types of parts makes it improbable that the computer could generate an accurate Index without a great deal of human assistance. For these cases as well as times when the user wants to create the Machining Index on his own, a manual Part Evaluator is included in DPCAM. This version, like the previously described automatic Evaluator, breaks the Index into sub-parts. It differs from the automatic version in that the user must input the sub-part values, along with the weightings, at his own discretion.

Conclusions

The Part Evaluator for axi-symmetric parts creates a Machining Index based on geometric, tolerance, surface finish and material information. These factors are combined in a way that best represents their actual relationships through the use of a set of hybrid weighting factor. These weighting combine both linear and exponential behavior. The results of this work is a Machining Index which indicates a part's complexity or "manufacturability". (figure 15)

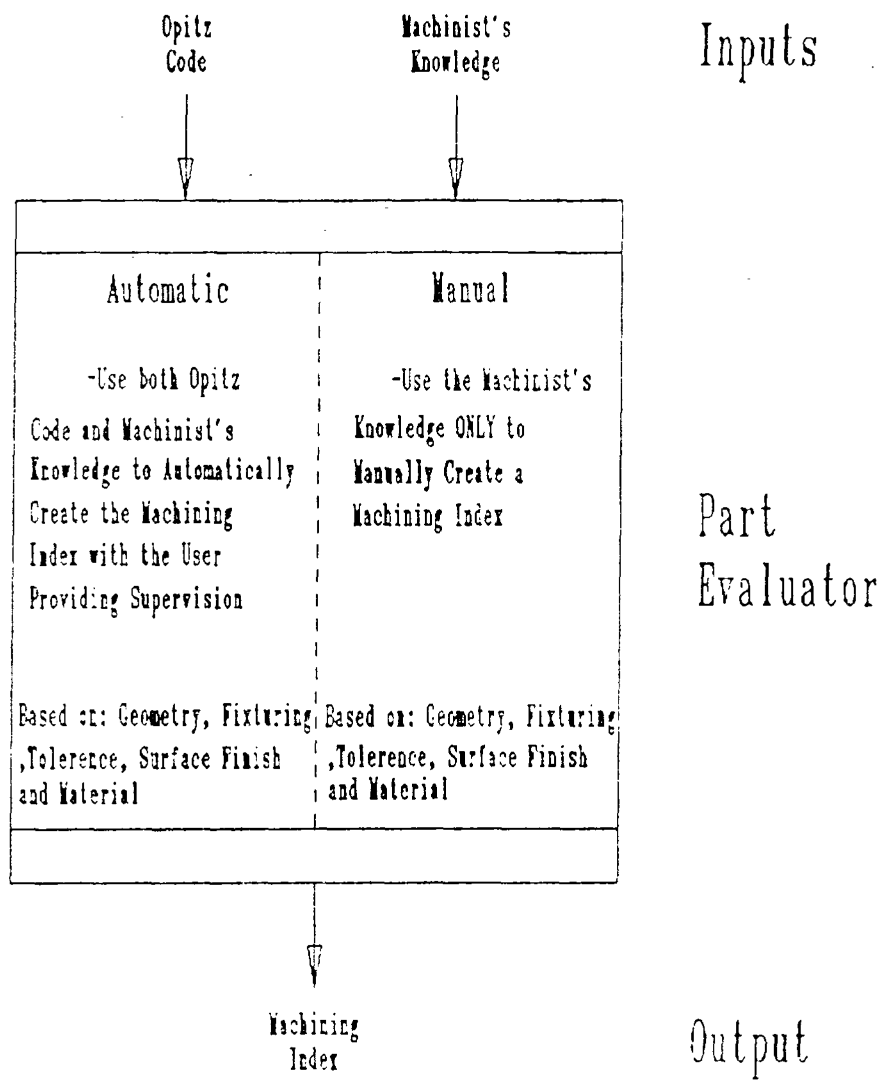


Figure 15: Diagram of the Evaluation Process

Chapter Five: Estimating

When a consumer makes any purchase, whether it be a bag of groceries or a new car, he shops around for a good deal. Companies that are successful and stay in business are those that offer the "best" prices; ones that give value to the user but also return an acceptable level of profit. Persons who contract a manufacturer for a particular item are also looking for a good deal. In order to get the job, a manufacturer must produce the best estimate of the item's total cost. The accuracy of this estimate should be sufficient to insure that not too low or too high a price is given; either of which could cause financial ruin. Thus it could be said that the success of a manufacturer rests on the accuracy of his estimates.

Unfortunately, the added complexities of todays marketplace in combination with the rapid turnover of technology on the shop floor, makes the calculation of accurate estimates difficult. One way to overcome these obstacles is to provide the estimator with newer techniques and equipment. Only in this way is greater efficiency and awareness possible.[21]

In considering the estimator for DPCAM, the decision is made to create a Computer-Aided Cost Estimator. It will not be a computer program that creates an estimate by itself, but more of an "advisor" to work with the machinist. By providing suggested values for different areas, the DPCAM output aims to be, if not more accurate, more consistent then a set of manually created estimates.

Elements of an Estimate-Estimator Structure

The first step in developing any estimate is the creation of a "work element structure". A work element structure serves as a framework for "hanging" on the individual values and times that make up the total estimate. Subdividing the estimate into a structure has advantages, such as allowing for individual area cost reports and the tracking of estimate accuracy as actual production proceeds.[30]

For the DPCAM Estimator, the structure needs to be general so that it can accurately report the wide range of items fabricated in a RPC environment. As such, the manufacturing process is divided into three major areas:

1. Preparation
2. Testing
3. Production

In turn, each of these must be subdivided further to provide enough detail for the creation of an accurate estimate.

Preparation includes the time needed to perform all setup and planning activities up to the point of the first cut. This time can be subdivided into four parts:

1. Preparation
    a) Examination and Adjustment of Blueprints
    b) Creation of a Process Plan

c) Initial Machine Setup

d) NC Code Generation

In addition, one of these elements, "Initial Machine Setup," is complicated enough to warrant further division into three subgroups:

c) Initial Machine Setup

i) Chuck/Fixture and Jig Selection

ii) Machine and Tool Selection

iii) Rough Cut/Stock

Testing time is the time needed to produce the first piece: the sample or "prototype" part. Not only must the time needed to cut the part be included but also time used for measurement and error correction. Testing can be subdivided into three sections:

2. Testing

a) Dry Run/Debug/Sample Machining

b) Sample Inspection

c) Modification and Correction

Production estimates include all of the cyclic actions typical of manufacture; positioning, machining, and unloading. To get a more accurate estimate, less frequent and one-time actions must be taken into account as well. Just as important are the costs associated with: preparation of the initial stock pieces, adjusting equipment to compensate for the wear of production, and the cleaning of facilities during and after production is complete.[30] With this in mind, the production phase is divided into six sections:

3. Production

a) Rough Cut/Stock

b) Positioning of Parts in Machine

c) Machine Operation

d) Quality Assurance of Random Samples

e) Adjustments for Wear

f) Clean Up

The combination of these three sets yields a general work element structure for DPCAM built around the needs of the RPC. (figure 16)

Role of a Computer in the Estimator

With the structure of the general RPC job set and broken down into individual elements, a method must be arrived upon that will "fill the branches" with the appropriate estimate values. In the case of DPCAM's Estimator the best approach is to put the computer in the role of an advisor to the machinist. It's function would be to provide the user with a "ballpark" value to be used as a starting point toward the final estimated time. In addition, the program can take advantage of a computer's math skills to keep track of all the individual categories as well as running totals of various areas of interest.

Generating Estimate Times

Many different ways are currently used to generate estimate time values for individual shop activities. Among the most popular are the use of standard handbooks and a machinist's "rule of thumb".[30]

Standard Estimating Handbooks, such as the American Machinist Cost Estimator Guide, provide tables of standard times for a wide range of shop activities. (table 3) To use them, all a machinist has to do is find a match for

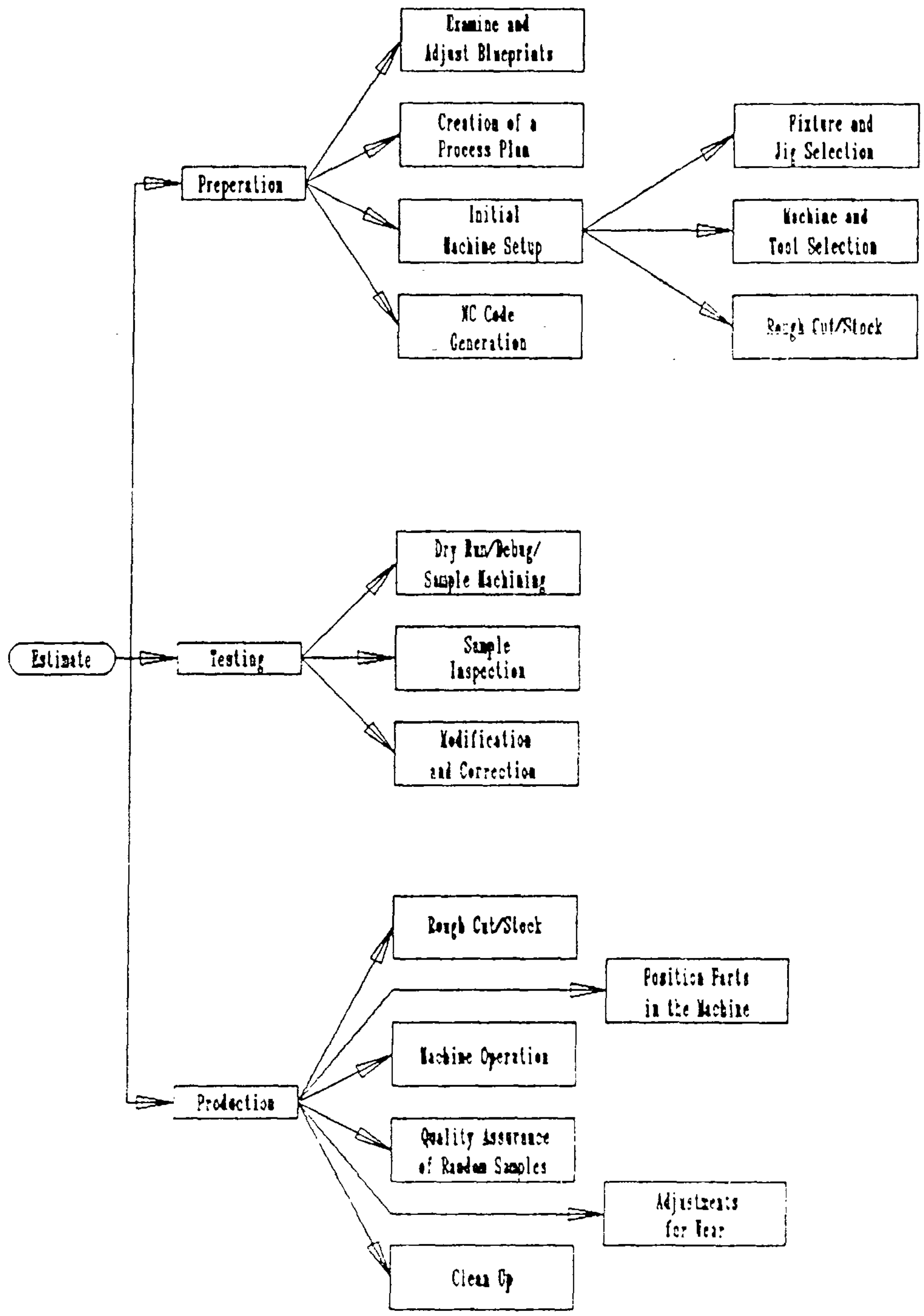


Figure 16: Work Element Tree for the RPC

A. Setup expressed in hours	
1. Punch in and out, study drawing	0.2
2. Turret lathe	
First tool	1.3
Each additional tool	0.3
Collet fixture	0.2
Chuck fixture	0.1
3. Milling Machine	
Vise	1.1
Angle plate	1.4
Shoulder-cut milling cutter	1.5
Slot-cut milling cutter	1.6
Tight tolerance	0.5
4. Drill press	
Jig or fixture	0.1
Vise	0.05
Number of numerically controlled turrets	
First turret	0.25
Additional turrets	0.07/each

B. Machining speeds and feeds

Material	Turning and Facing (fpm, ipr) High-Speed Steel Rough	High-Speed Steel Finish	Tungsten Carbide Rough	Tungsten Carbide Finish
Stainless steel	150,0.015	160,0.007	350,0.015	350,0.007
Medium carbon steel	190,0.015	125,0.007	325,0.020	400,0.007
Cast iron, gray	145,0.015	185,0.007	500,0.020	675,0.010

Material	HSS Slab Milling Rough	HSS Slab Milling Finish	HSS Slot 1 in. Rough	HSS Slot 1 in. Finish
Stainless steel	140,0.006	210,0.005	85,0.002	95,0.0015
Medium carbon steel	170,0.008	225,0.006	85,0.0025	95,0.002
Gray cast iron	200,0.012	250,0.010	85,0.004	95,0.003

Drill Diameter*	Stainless Steel	Medium-Carbon Steel	Cast Iron, Gray
$\frac{1}{4}$	0.55	0.20	0.20
$\frac{5}{16}$	0.61	0.23	0.23
$\frac{3}{8}$	0 65	0 25	0 25

Table 3: Example of Standard Time Tables [21]

each activity, and plug the given value into the corresponding location of his work element structure. Problems with this method occur when the machinist attempts to find exact matches to particular machining activities. The standards books can only cover so much material. As a result, they make many broad assumptions in their time calculations. In a setting like the RPC, where the newest machinery and techniques are used, the times stated in the books will most likely be a poor reflection of what actually occurs.

The other option available is to assemble a set of "rule of thumb" estimating data directly from the machinists in your RPC. This technique offers much improved accuracy over the "standard" method since information is based on the equipment and actions typically used in the shop of interest. The main drawback of "rule of thumb" information is that in order to assemble an accurate database, a historical study must be conducted on the operations of the particular shop.

Since a "rule of thumb" technique offers the potential for greater accuracy then the use of standard time tables, this is the estimating techniques incorporated into DPCAM. The required knowledge base is assembled through a historical study of shop estimates. The proper technique for this study would involve the organization of jobs into varying steps of difficulty. Using the RPC's work element structure as a base, the times for each level of difficulty can then be entered into the individual RPC estimate categories. The accuracy of the knowledge base can be improved by increasing the number of step divisions and the amount of data used in the averaging process. To allow for operations as soon as possible, an initial database is assembled through a short investigation of machining activities at the University of Maryland's RPC (see Appendix C).

Driving the Estimator

The final element needed within the estimator is a methodology to choose among the tables of assembled expert data for the appropriate time value. An ideal estimate "driver" would be one which describes the overall difficulty of a particular part's manufacture. Such a value exists in the form of the DPCAM-produced Machining Index. The Machining Index could be input into the Estimator to direct the program toward the appropriate time data. Incorporation of the Machining Index value into a searching structure gives the system the measure of a part's "manufacturability" needed to complete the estimate.

Conclusions

The goal of the Estimator module of DPCAM is to advise the user in the creation of an accurate and consistence estimate of work to be done in the RPC. Such information is crucial to the success and well being of any money making institution. The estimate is arrived at by breaking the job up into a set of general actions common to all RPC jobs. Time and cost outputs to the user are taken from a database of "rule of thumb" knowledge incorporated into the Estimator program. Decisions made by the Estimator are based on information regarding the part overall "manufacturability". The "manufacturability" is supplied by DPCAM's Machining Index. (figure 17)

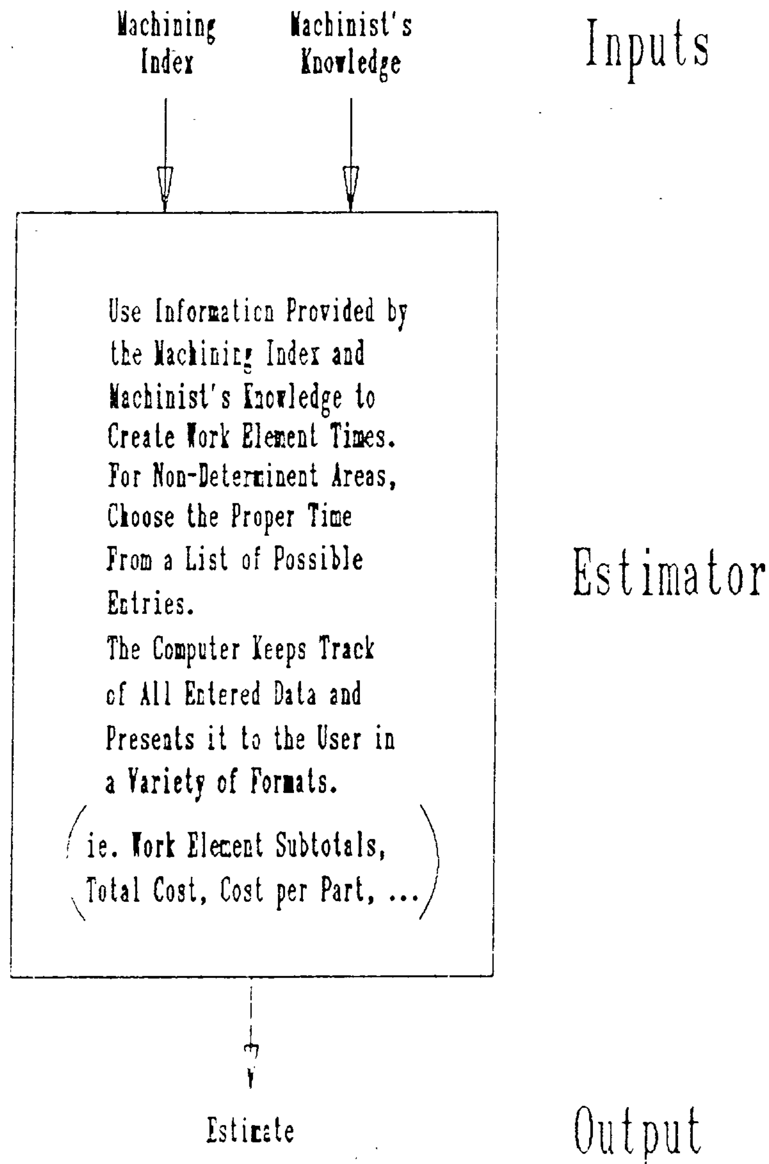


Figure 17: Diagram of the Estimation Process

Chapter Six: Code Generation for Axi-Symmetric Parts

When all is said and done, the common perception of the manufacturing process is the physical creation of a part. The cutting of metal is the "meat and potatoes" aspect of the entire process. Thus it comes as no surprise that most of the available aids to the machinist are in the area of metal cutting and forming. The amount of research and the volume of products available in this area, such as Computer Numerically Controlled (CNC) machine tools and programing aides for these tools, far outstrip advances made on other "less glamorous" topics.

In order for the RPC to achieve the desired level of flexibility and automation, CNC machines must be incorporated into the machine shop. To provide the user with assistance in this migration, DPCAM incorporates a NC Machine Code Generation module. To keep up with current technology, the module should provide a machinist an easy access to the power and flexibility of today's CNC machines. This is done by removing the need for fluency in the cryptic M+G language they communicate in.

History of NC Technology and How it Works

Lying at the heart of 80's manufacturing technology is the CNC machine. These machines have replaced normal machine tools in most levels of manufacturing due to the advantages, flexibility and repeatability they allow.

Today's NC tools got their start due to the imagination of a man named John T. Parsons. Parsons conceived of a method for instructing a machine to move on a path by using punched cards containing the necessary coordinate information. In 1948, following a demonstration of his work, the newly formed U.S. Air Force contracted MIT to do further work and the NC Machine was born.[25] The current version of NC technology is the Computer Numerically Controlled (CNC) Machine. The CNC machine differs from the NC machine through the incorporation of a digital computer as the controlling unit.

A CNC Machine can be divided into three major components (figure 18):

1) Program of Instructions

2) Control Unit

3) Machine Tool

The program of instruction is the set of steps that a user creates and feeds into a CNC Machine that dictates every move and action that the device will make.[8] The loosely standardized language that most CNC Machine communicates in is the set of M+G codes. Unfortunately, a line of M+G codes appears as a jumbled set of letters and numbers to the inexperienced user. To become a CNC machine operator means that the machinist has to spend many months or years becoming fluent in the machine's cryptic code. Since the

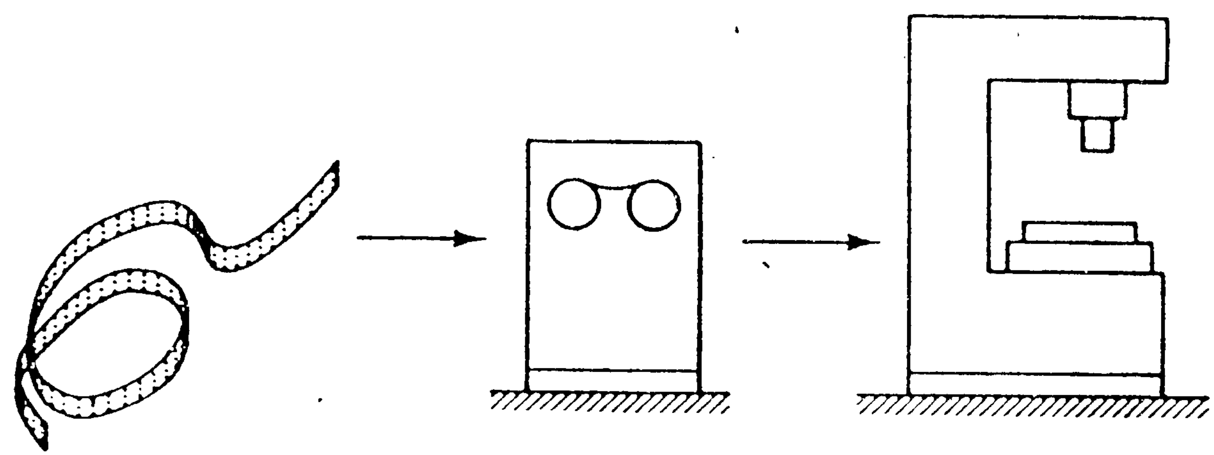


Figure 18: Diagram of an NC System [8]

development of numerical control, alternatives have been sought to this predicament.

Initially, "upper level" languages, such as APT and Compact II, were developed, that allowed a user to create a set of "English-like" instructions describing the tool paths. (table 4) This set of steps would then be run through a post-processor that converted them into the appropriate M+G codes, which were then fed into the machine.[25] The expansion of computer technology in the 1970's and 80's has made the use of interactive graphics code generation programs popular. In this method, the user enters the geometry of the desired part on the graphics screen of a computer, along with tool, speed and feed data. This input is then processed into the appropriate M+G codes for input into the machine.[24]

The Control Unit is the device which interprets the instruction set and directs the machine tool into the proper action.[8] Early Control Units used punched paper tape containing the M+G code as input to instruct the machine tool into action. These units had no bulk memory and required that the paper tape be run each time a part was made.

Today's controllers are powerful digital computers (thus Computer Numerical Control). This lets the user store many programs in memory and make changes in them at the controller console rather then creating an entirely new paper tape. In addition, computer control allows for easier machine control, in-process compensation and self-diagnosis of machine problems: all functions that save money and increase productivity.[8]

The Machine Tool is the component of the system which does the actual cutting. Early NC machines used lathes and mills as their tool base. As the

```
Column
1      6   8   10                                              72
PARTNO         EXAMPLE PART
               MACHIN/MILL, 1
               CLPRNT
               INTOL/.001
               OUTTOL/.001
               CUTTER/.500
P0         =   POINT/0, -1.0, 0
P1         =   POINT/6.0, 1.125, 0
P2         =   POINT/0, 0, 0
P3         =   POINT/6.0, 0, 0
P4         =   POINT/1.75, 4.5, 0
L1         =   LINE/P2, P3
C1         =   CIRCLE/CENTER, P1, RADIUS, 1.125
L2         =   LINE/P4, LEFT' TANTO, C1
L3         =   LINE/P2, P4
PL1        =   PLANE/P2, P3, P4
               SPINDL/573
               FEDRAT/2.29
               COOLNT/ON
               FROM/P0
               GO/TO, L1, TO, PL1, TO, L3
               GORGT/L1, TANTO, C1
               GOFWD/C1, PAST, L2
               GOFWD/L2, PAST, L3
               GOLFT/L3, PAST, L1
               RAPID
               GOTO/P0
               COOLNT/OFF
               FINI
```

Table 4: Example of the APT Language [8]

technique prospered, it was applied to virtually every type of manufacturing activity, from drilling to electrochemical machining.

The first NC machine tools only had positioning motors, which, placed on their primary axes, provided for accurate tool placement. Continued development on this basic concept have resulted in the ultimate expression of an NC machine tool, the Machining Center. Today's machining centers expound on the basic concept by adding additional axes, automated tool changing and automated part positioning. A powerful machining center is usually at the heart of a flexible manufacturing setup.[29]

Development of a Code Generator for Axi-Symmetric Parts

The Code Generation Module for axi-symmetric parts in DPCAM should allow a machinist to program a CNC lathe easily. Information supplied by a designer's blueprint and the machinist's knowledge on items such as feed rates and spindle speeds will be the only required input. (figure 19) The module can be broken into three sections:

-creation of a new Part File

-editing of an old Part File

-processing a Part File to M+G code.

Ideally the code generated by the module should model typical turning operation.

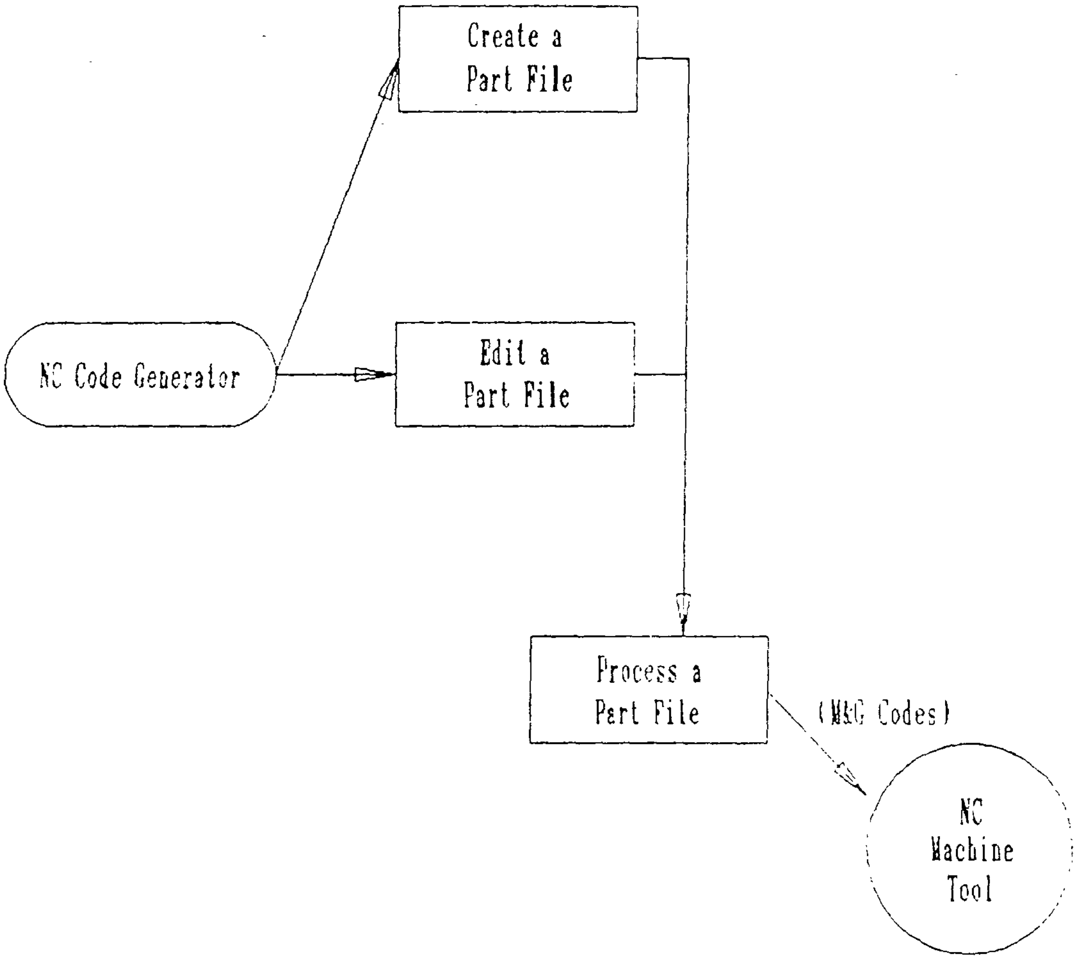


Figure 19: Layout of the DPCAM Code Generator

Turning Operations

Turning operations are divided into two categories, internal and external. (figure 20) [6] External cuts are the easier of the two types with little limitation on form or procedure. Among the contour types in external turning are facing, turning, threading and grooving/cutoff. Internal cuts present more of problem. Fixturing strategies for internal operations are limited to "chuck" types since access to the stock face is needed to achieve desired internal contours and overhangs. Among the contour types in internal turning are facing, turning, threading and drilling/boring.

Most turning operations are either facing or turning (external and internal). Facing and turning involves the creation of a desired contour along the part's surface. These contours are categorized as a continuous string of lines and arcs. The names "facing" and "turning" describe the direction in which the cutting takes place. Action taken along the longitudinal axis are "turning" ones while those along the diameter are "facing" cuts.

Both facing and turning contours are usually made in one of two ways, in a single pass or in multiple passes or "roughing". Single pass operations are performed just as the name indicates; in one pass. In roughing operations, the material is removed a little at a time until the desired contour is achieved. Roughing operations are usually followed by a single-pass finish cut.

Code Generator Structure

The goal of the Code Generator is to take blueprint part geometry and machinist process plan data as input. Output of the module should be M+G

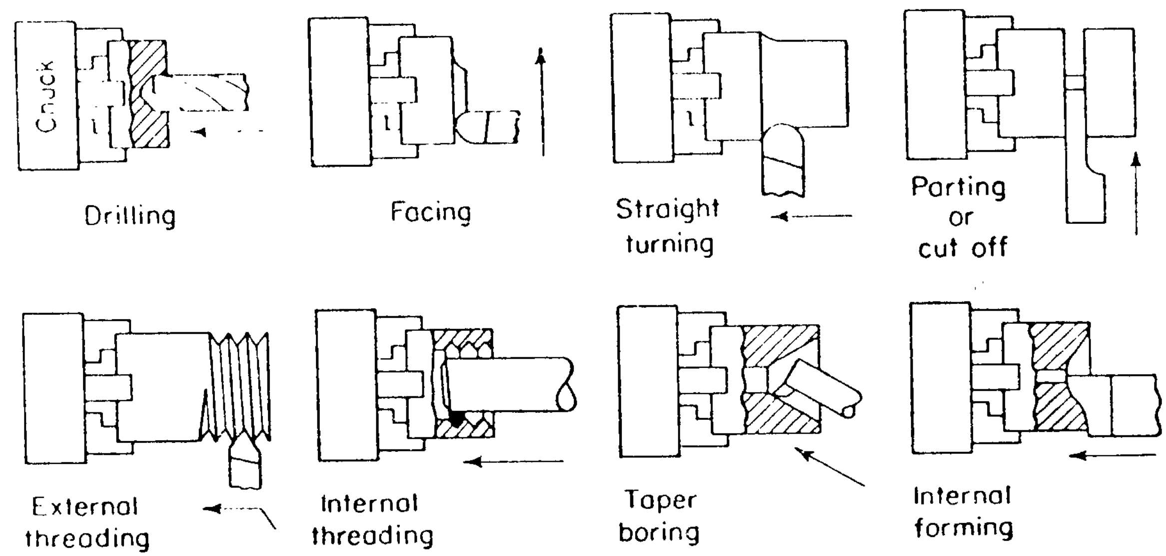


Figure 20: Internal and External Turned Cuts [6]

codes that provide ideal turning operations toward the desired shape. The first step toward this is the creation of an intermediate file. An intermediate file is used as a provision to create M+G codes for different turning centers. Most turning centers use M+G code, but incorporate slight deviations unique to their machine.[25] By creating an intermediate Part File containing all necessary machining parameters, slight formatting changes in an M+G processor will allow the module to make codes for different machines.

The intermediate Part File contains all the necessary information needed to machine a particular piece. (table 5) It is entered in such a way that process plan and machine parameter data are combined with the appropriate geometric information. This is achieved by having the user enter the information in ordered sets of operations. The first division involves grouping contours into sets of external and internal operations. Each of these sets is further divided into subsets of contours based on cutting order and cutting operation. The computer prompts the user to work his way through the hierarchy to the appropriate level and then requests all needed information to machine the feature. The shape of a desired contour is entered by the user as a continuous string of lines and arcs. (figure 21) Upon completion, the internal or external profile is punctuated by a return to the machine's home position. The saved Part File preserves this ordering so that proper editing and processing can then take place.

The editing functions created for the Code Generating module take advantage of the Part File's hierarchical structure. The user can add, delete or modify the Part File at any level. This makes functions ranging from the addition or deletion of an entire external or internal contour set to the modification of a single contour parameter, a simple operation.

```
rapid -7.000 8.250
tool_new 5.000
speed 3500.000
feed 1.000
cool_on
EXTERNAL
rapid -0.500 1.250
ROUGH
line -1.750 1.250
rough71_1 0.099 0.049
rough71_2 12.000 17.000 0.000 0.000
line -0.500 1.250
arc -1.000 0.750 1.000 1.000
line -1.250 0.750
line -1.450 0.250
line -1.500 0.250
line -1.750 0.250
ENDPASS1
finish 12.000 17.000
rapid -1.750 0.250
line -1.500 0.250
ENDROUGH
line -1.750 0.250
rapid -7.000 8.250
ENDEXT
cool_off
tool_old 5.000
end
```

Table 5: Typical DPCAM Part File

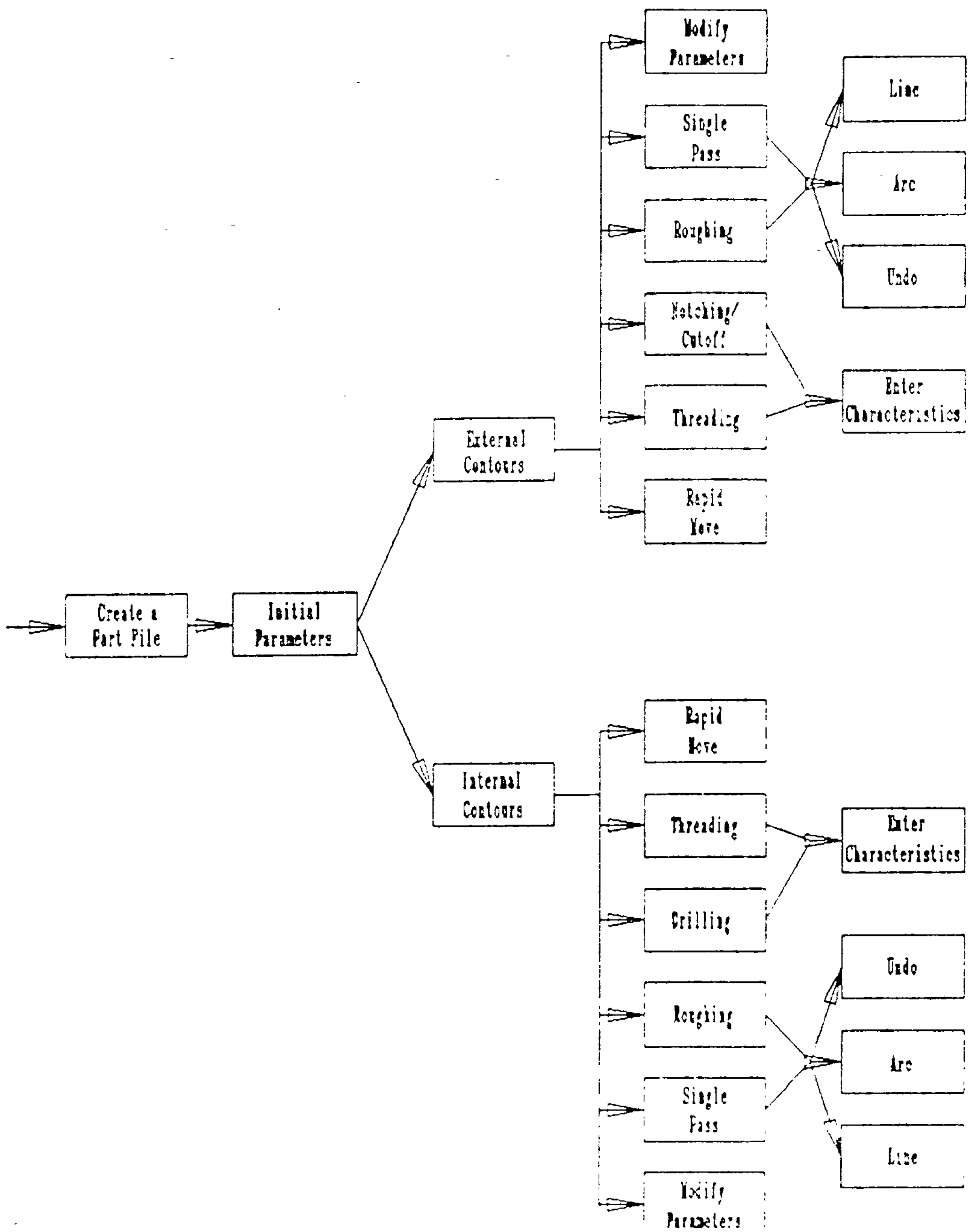


Figure 21: Layout of the *Create* Section

M+G codes are generated by running a completed Part File through the Code Generator's M+G processor. The processor functions as a translator, assembling the information supplied by the Part File into the M+G format the turning center being used expects. The processor included in DPCAM creates code formatted to satisfy a Fanuc OT class controller, the type currently available. (table 6) [34,36] The processor also includes the functions needed to download prepared M+G codes to the turning center's Control Unit.

Conclusions

The goal of DPCAM's Code Generation module is to allow the machinist easy access to the powerful CNC turning centers appearing on the shop floor. It does this by providing the user with a friendly graphical environment that prompts for basic geometric and process plan information. The module then converts this information into the M+G format the turning center expects. By removing the burden of part programming from the machinist's hands, CNC equipment becomes a more effective tool, open to a wider base of users not fluent in M+G coding techniques. (figure 22)

```
%O0001 ;
N1  G00 X-7.000 Z8.250 ;
N2  T0505 ;
N3  S3500   M3;
N4  F1.000 ;
N5  M08 ;
N6  ( EXTERNAL ) ;
N7  G00 X-0.500 Z1.250 ;
N8  ( ROUGH ) ;
N9  G01 X-1.750 Z1.250 ;
N10  G71 U0.099 R0.049 ;
N11  G71 P12 Q17 U0.000 W0.000 ;
N12  G01 X-0.500 ;
N13  G02 X-1.000 Z0.750 R1.000 ;
N14  G01 X-1.250 Z0.750 ;
N15  G01 X-1.450 Z0.250 ;
N16  G01 X-1.500 Z0.250 ;
N17  G01 X-1.750 Z0.250 ;
N18  ( ENDPASS1 ) ;
N19  G70 P12 Q17 ;
N20  G00 X-1.750 Z0.250 ;
N21  G01 X-1.500 Z0.250 ;
N22  ( ENDROUGH ) ;
N23  G01 X-1.750 Z0.250 ;
N24  G00 X-7.000 Z8.250 ;
N25  ( ENDEXT ) ;
N26  M09 ;
N27  T0500 ;
N28  M30 ;
%
\
```

Table 6: Typical DPCAM M+G Code

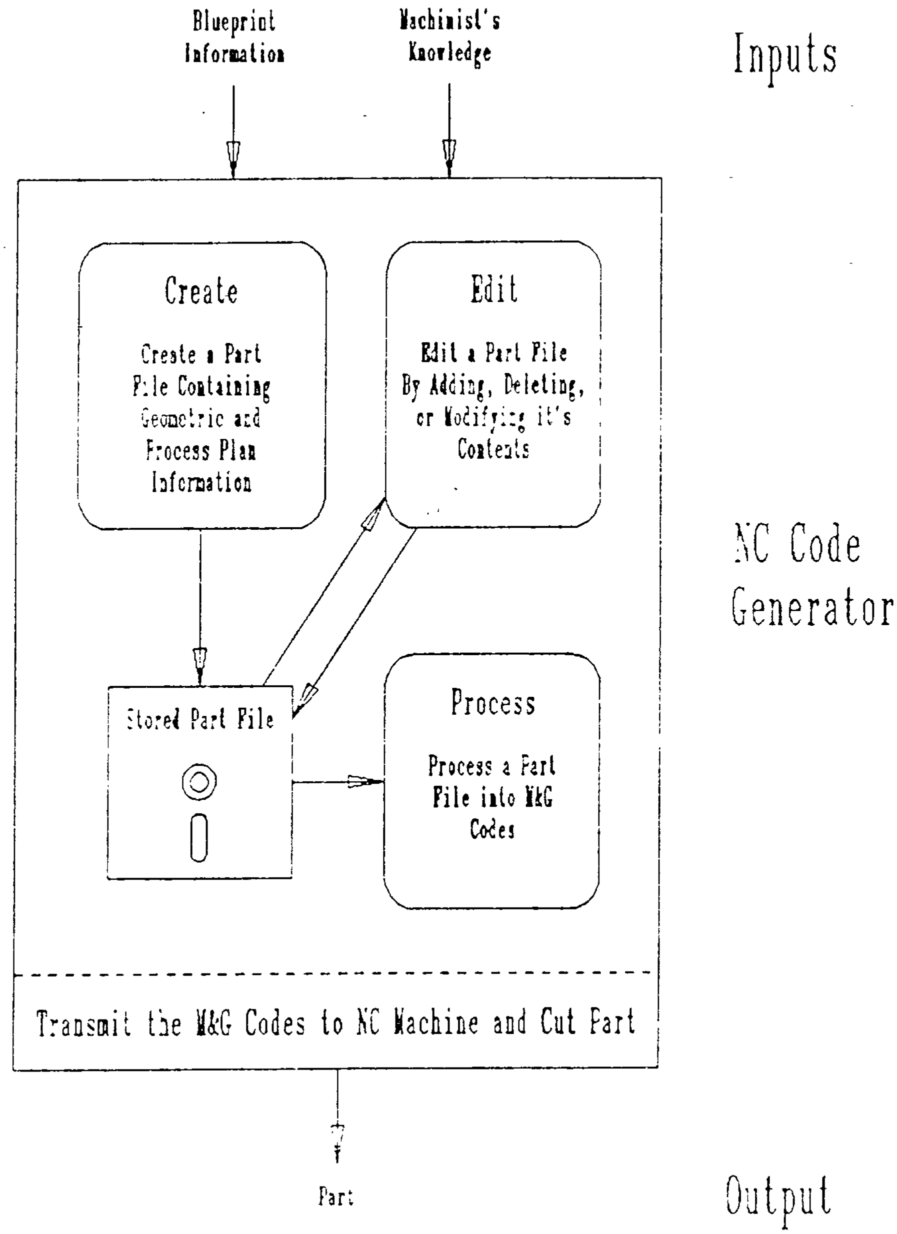


Figure 22: Diagram of the Code Generation Process

Chapter Seven: Development of the User Interface

Previous chapters of this thesis have concentrated on the underlying philosophy of the DPCAM concept and the strategies employed to create the "mechanisms" of the program. This effort could easily be rendered useless if as much time is not spent creating a methodology for communicating the results to the user. An effective user interface is as important as the computation power or function of a software package.[13]

For DPCAM, the best possible interface is one that can effectively deliver the wide variety of program output to the machinist. To achieve this one must consider his computer skills as well as his work environment, the shop floor. These requirements dictate the type of equipment as well as the software format.

Equipment Considerations

Two classes of computers are applicable to the level of power needed to implement the DPCAM strategy; "PC" or "Workstation" class computers.

Workstations, like the SUN 3/160, offer the benefits of high speed, large storage, multiple processing and high resolution graphics. Each of these elements are important when considering the problems programming DPCAM presents. Unfortunately, the advantages of a workstation are offset by a number of insurmountable problems. Workstations get their power at the cost of complex operating techniques. They are also, not in general, very ruggedly built. The final blow to the workstation is it's price tag; $20,000. and up. It is hard to justify developing a software package for the machine shop on an expensive piece of equipment that might not be able to function in that environment.

The other alternative is to make DPCAM "PC-Based". Ten years ago this would have been an impossible proposition. However, the 1980's have produced great strides in microcomputer technology. The latest generation of 80286 and 80386 "PC compatible" computers, outfitted with high resolution color graphics monitors, offer computational power and graphical abilities nearing those of the workstation. Another factor to the "PC's" advantage is their great proliferation throughout the marketplace. The result is an abundance of software and large base of support. Combining this with their low cost and the availability of hardened "shop floor" models makes the "PC" ideal as a hardware base for DPCAM.[32]

The language chosen for programming DPCAM is the "C" language. "C" offers the advantages of system portability and built-in graphics commands.[17] Borland's Turbo C v1.5 "C" compiler for IBM "PC" compatibles was used throughout the programming of DPCAM.[14,26]

Software Interface

The key to any software package is creating a successful dialog between the user and the program. The two possible methods for creating this interface are text or graphics.[4]

Text or Graphics??

Early attempts at computer integration concentrated on using "English-like" input and output as a means of communication.[8] Two examples of this are the APT NC Code generating scheme and the MICLASS group technology method. (table 7) Programs which employ this technique find themselves languishing on the shelf since the user is not supplied with an effective aid to help him visualize the problem.

Computer graphics are the cornerstone of most new CAD/CAM applications. For these types of programs, the graphics system is the part that a user depends on and interacts with the most. (figure 23) [13] It is only logical then that DPCAM employs graphics throughout its user interfaces.

Structure of the DPCAM Interface

Any application of DPCAM will involve the input and output of graphical, numerical and text type information. Depending on the particular module employed, one of three balances of information types are found. (table 8) In response to these needs, three different screen orientations are designed. (figure 24) Each of these screens use a series of viewpoints to segregate different areas of text and graphics. The result is the appropriate level of visual feedback for the particular module being used.

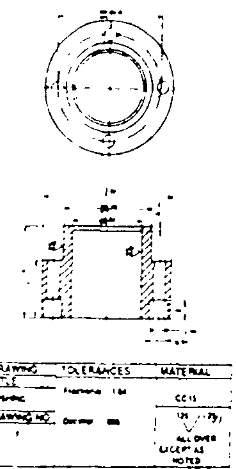
Two options are available to handle the input functions of DPCAM: either a menu-driven or a command-line-driven technique. In a menu-driven system, a list of options is presented to the user from which the desired choice is

```
RUN *MICLAS
   MICLAS  VERSION 2.0

ENTER THE CLASSIFICATION ROUTE (1 TO 9)  1
3 MAIN DIMENSIONS (WHEN ROT. PART D+L AND 0)  2.9375,2.0
  DEVIATION OF ROTATIONAL FORM  NO
  CONCENTRIC SPIRAL GROOVES  NO
TURNING ON OUTERCONTOUR (EXCEPT ENDFACES)  YES
  SPECIAL GROOVES OR CONE(S) OR PROFILE(S)  ON OUTERCONTOUR >NO
  ALL MACH. EXT. DIAM. AND ROT. FACES VISIBLE FROM ONE END
 (EXC. ENDFACES + GROOVE(S))  YES
TURNING ON INNERCONTOUR  YES
  INTERNAL SPECIAL GROOVES OR CONE(S) OR PROFILE(S) >NO
  ALL INT.DIA." + ROT.FACES VISIBLE FROM 1 END(EXC. GROOVES >YES
ALL DIA. + ROT.FACES VISIBLE FROM ONE END (EXCL. ENDFACES >YES
ECC. HOLING AND/OR FACING AND/OR SLOTTING >YES
  ON INNERFORM AND/OR FACES (INC. ENDFACES) >YES
  ON OUTERFORM >NO
ONLY ENCLOSED INTERNAL SLOTS >NO
ECC. MACHINING ONLY ONE SENSE >Y
  ONLY HOLES ON A BOLTCIRCLE (AT LEAST 3 HOLES) >YES
FORM-OR THREADING TOLERANCE >NO
DIAM. OR ROT. FACE ROUGHNESS LESS THAN 33 RU (MICRO-INCHES) >YES
  SMALLEST POSITIONING TOL. FIELD >.016
  SMALLEST LENGTHTOL. FIELD >.0313

CLASS.NR.= 1271 3231 3100 0000 0000 0000 0000 00
************************************************

DIGIT TO CHANGE >
CONTINUE (Y/N)>N
TTO  --  STOP
```



Drawing of Part

Table 7: Sample Text Screen (MICLASS) [2]

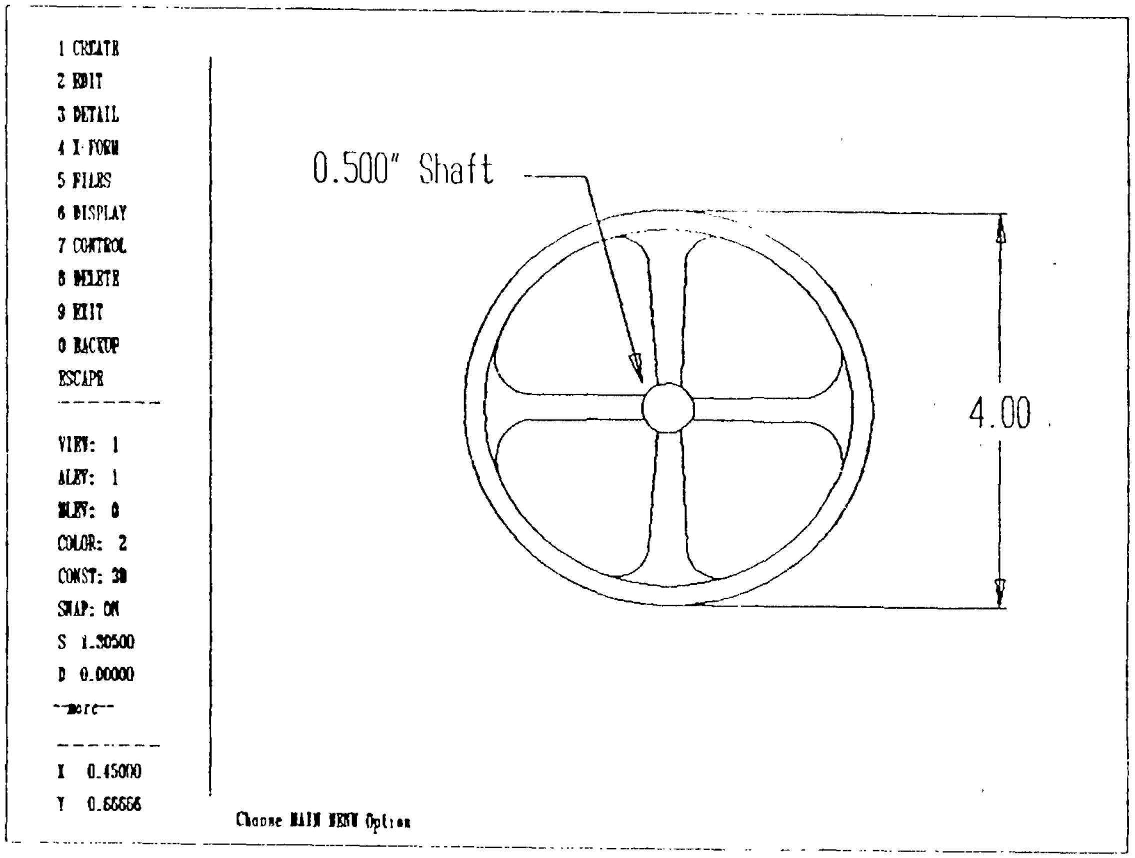


Figure 23: Sample Graphics Interface (CADKEY)

Module	Graphics	Text
Initialization	Avg.	Avg.
Opitz Module	High	Low
Design Retrival	Low	High
Part Evaluation	Avg.	Avg.
Estimation	Avg.	Avg.
Code Generation	High	Avg.
Equipment Managment	Avg.	Avg.

Table 8: DPCAM Screen Requirements

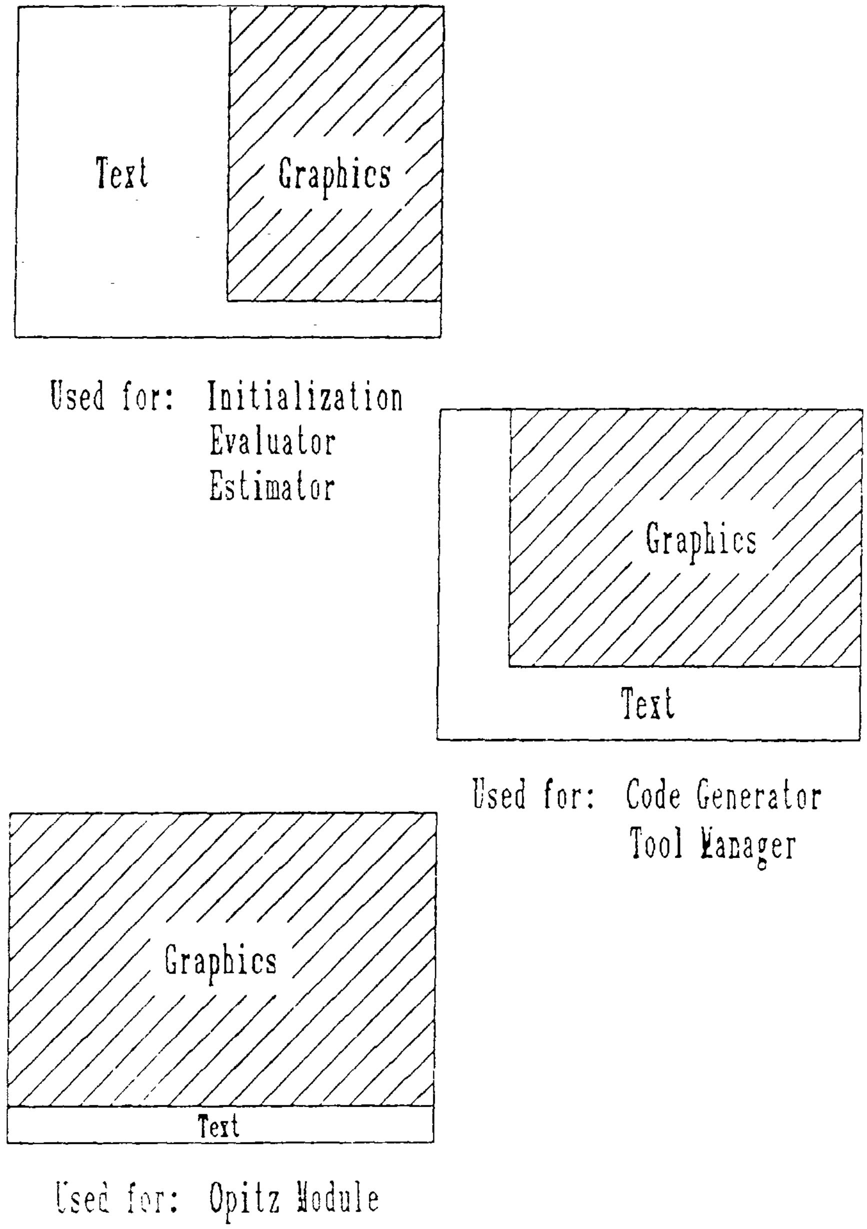


Figure 24: Three Screen Layouts for DPCAM

made. This is usually done by highlighting the choice with one of the computer's pointing devices; the "arrow-keys" for instance. Many times, menus are chained together to present the user with a larger range of choices. A menu system is generally considered the friendlier of the two ways but is slower since a single action may require multiple entries. [4,13]

A command-driven system requires the user to type in the desired action directly. This type of input is not very "friendly" but allows for very fast operation for an experienced operator. To make these systems easier to use, extensive help and error message facilities must be incorporated into the program. Command style systems are also the most effective ways for entering "real" data (numbers, names, etc.) into a system.[13]

Taking into account that the average user of DPCAM is a machinist, not an experienced computer operator, a network of menus is chosen for the majority of the input. When names and numerical data are needed, however, the command-line style is used.

DPCAM outputs a great amount of information to the screen. In order to help the user make sense of this mass of information, a technique is needed to direct the user's eyes to the correct area of the screen. The easiest way to attract a viewer's attention is through the use of color.[13] By using color, a programmer can integrate, emphasize or obscure particular outputs. Through the application of color, DPCAM allows the user to verify a menu choice, identify an error message, as well as a whole host of other features.

Conclusions

A key element in the development of DPCAM is the creation of a friendly interface. Without it, the program will go unused and the previous work wasted. After considering the abilities of the typical user and the probable work environment, it is decided to build the system around a "PC" style computer. The "C" language is chosen as the programming environment due to the advantages in graphical power and portability it offers. The requirements of the software require that three different screen configurations be used on the project. Each of these screens will employ numerous viewports, menus and color to achieve a maximum level of user friendliness.

Chapter Eight: An Example of DPCAM Operations

In its completed form, DPCAM is broken into two sections; initialization and application. (figure 25) At the initialization level, the user must input the basic information needed by the software; a part name and an Opitz code. A short note can also be attached containing additional information. Once the program has received these basic characteristics, it goes about its first task, Design Retrieval. The results of the search are displayed to the user as the program moves into the applications level.

The applications level greets the user with a menu of the functions available in DPCAM. After the user makes his choice, the appropriate module is started, which upon completion, returns him again to the menu. The user has the option to use as many applications as he desires, work on another part or quit the DPCAM system.

To give a more thorough description of each element's function, an example run through of the program is performed.

Example

It is desired to manufacture the item shown in figure 26 using the DPCAM package.

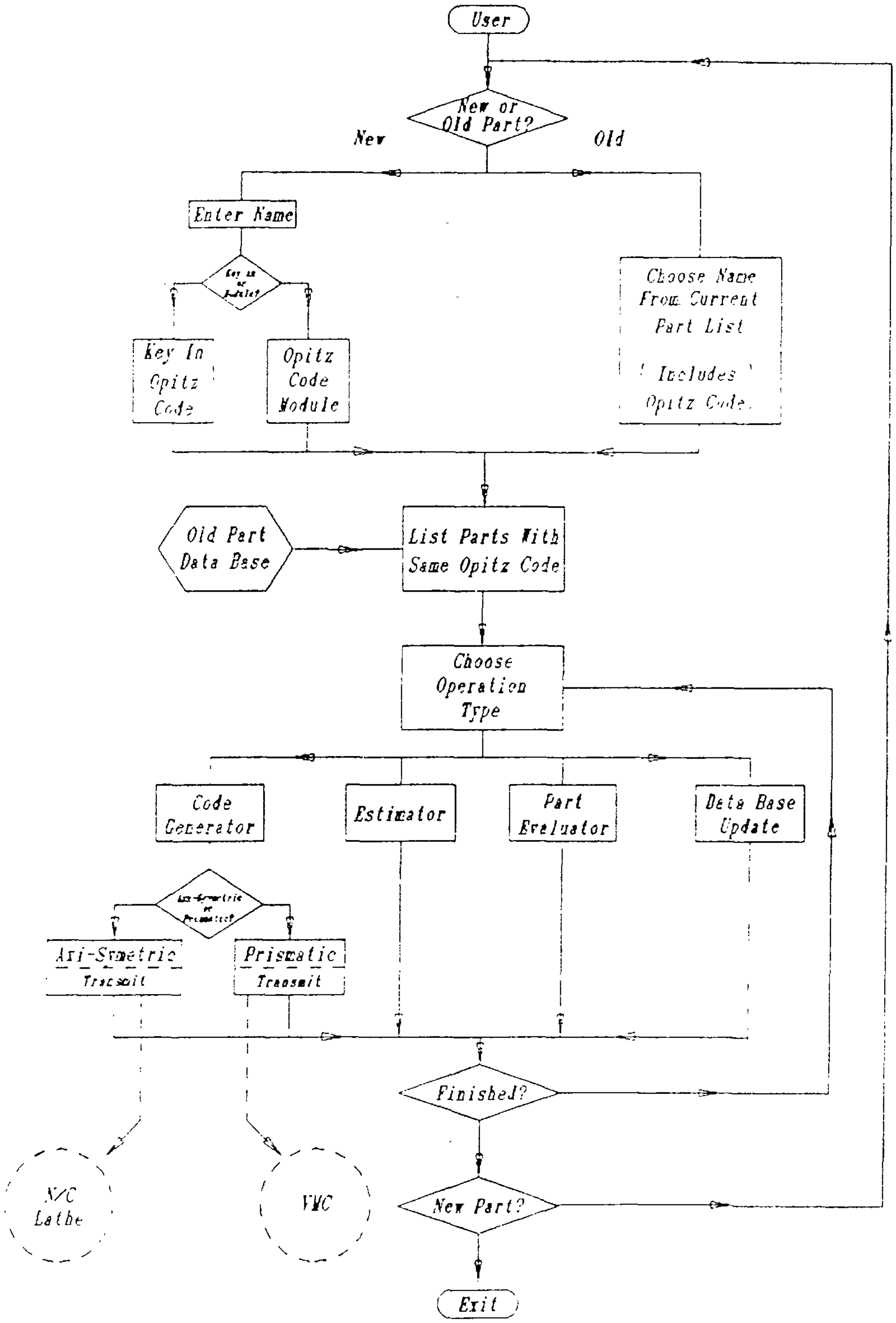


Figure 25: DPCAM System Structure

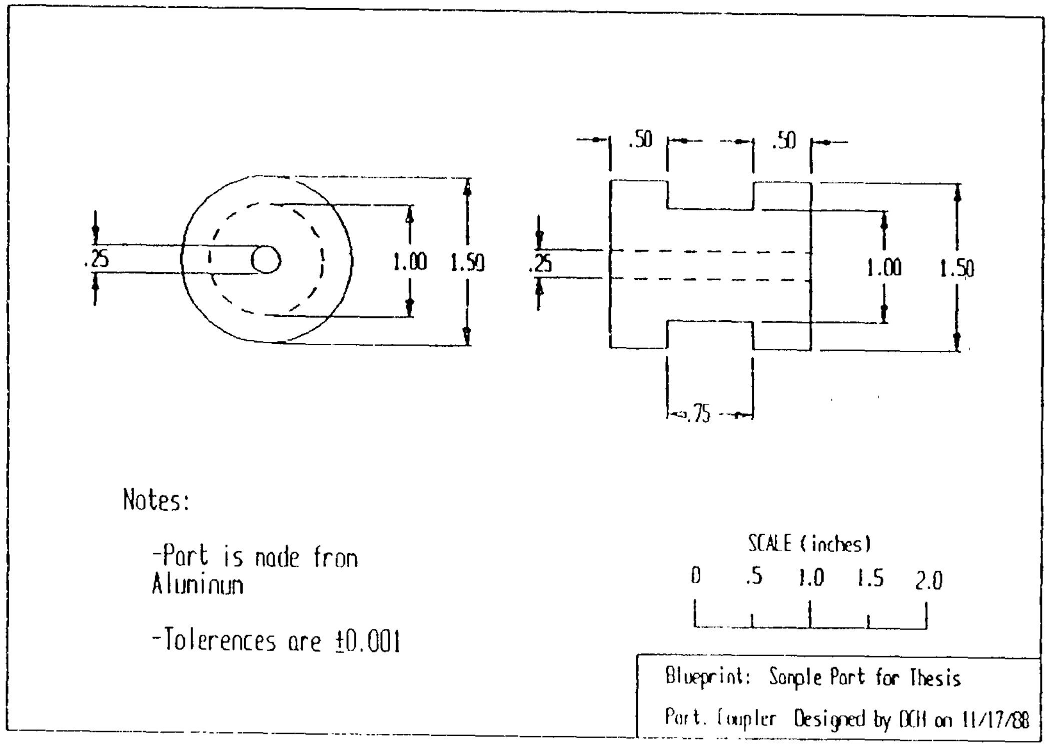


Figure 26: Blueprint of Example

To start the package, the command is "DPCAM" entered. An initial title screen appears and the computer prompts for the class of part to be worked on; "New" or "Old". (figure 27)

If no previous DPCAM work has been done on the current part, "New" is chosen. The machine then asks for the basic part characteristics; name, Opitz code and an additional memo. (figure 28) The entry of the Opitz code warrants special attention. If the user is familiar with the Opitz system, the option is given to key in the code directly. Otherwise, the Opitz generation module is used. This module uses a combination of questions and pictorial choices to direct the user toward composition of the correct Opitz code. (figure 29) When the data is completely entered and the user satisfied, this basic information is stored by the program. This allows for both future editing and Design Retrieval operations.

If previous DPCAM work has been done, "Old" is chosen. The machine then presents the user with a list of parts available for editing. Also listed is the status of the DPCAM work files for each part. Choosing the part of interest is a matter of scrolling through the list to the appropriate entry. (figure 30) The program uses the characteristics retrieved from the chosen parts stored data file as its driving data.

Once the proper part information is loaded into DPCAM, its initial operation, Design Retrieval, is performed. To do this, DPCAM opens a datafile containing all parts previously entered into the program. The utility searches this list, printing to the screen, names of parts with similar configuration. (figure 31) This decision is based on a comparison of Opitz values. The advantage of this operation is that the user might find a previous part applicable to the current use, and avoid duplication of past work.

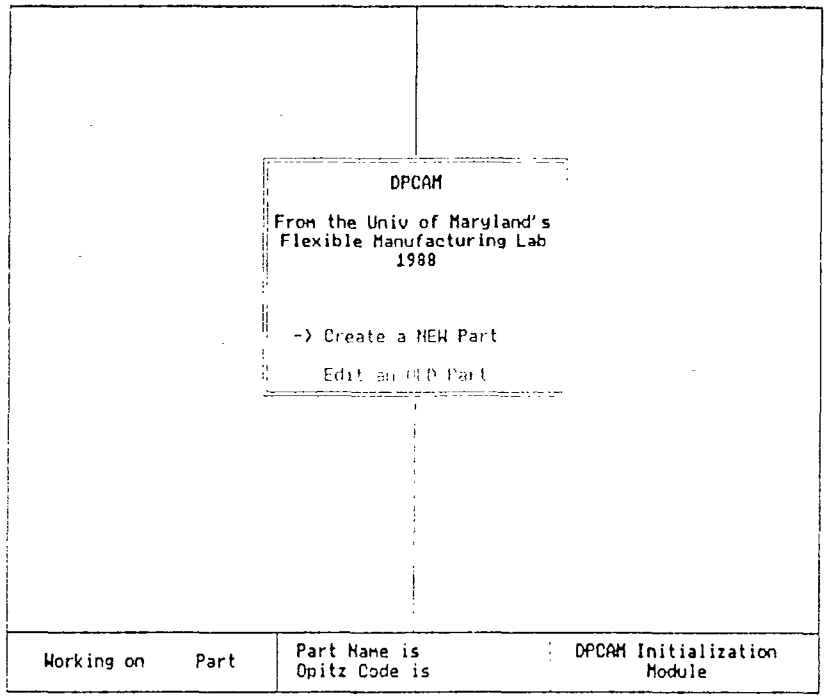


Figure 27: DPCAM Title Screen

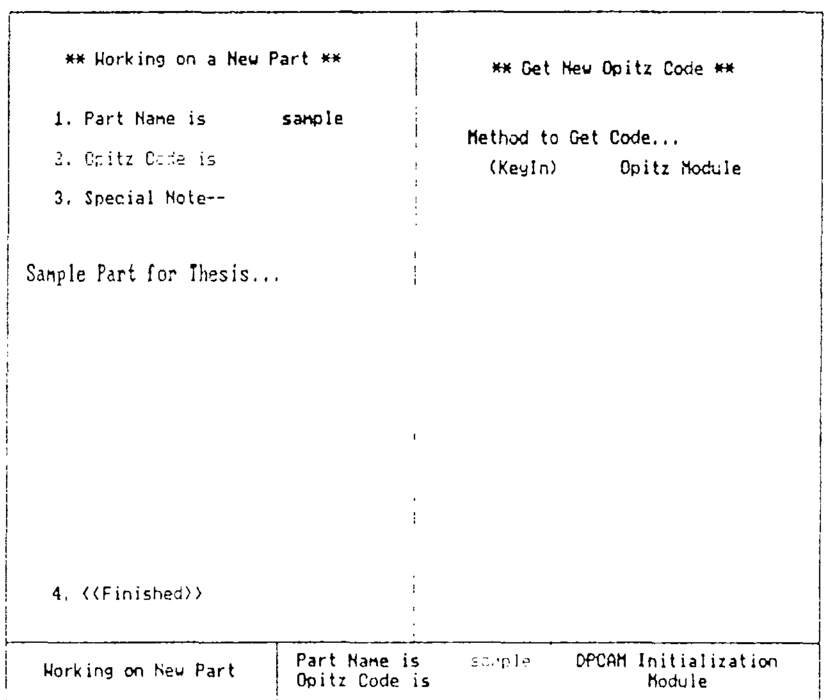


Figure 28: DPCAM *New* Part Option

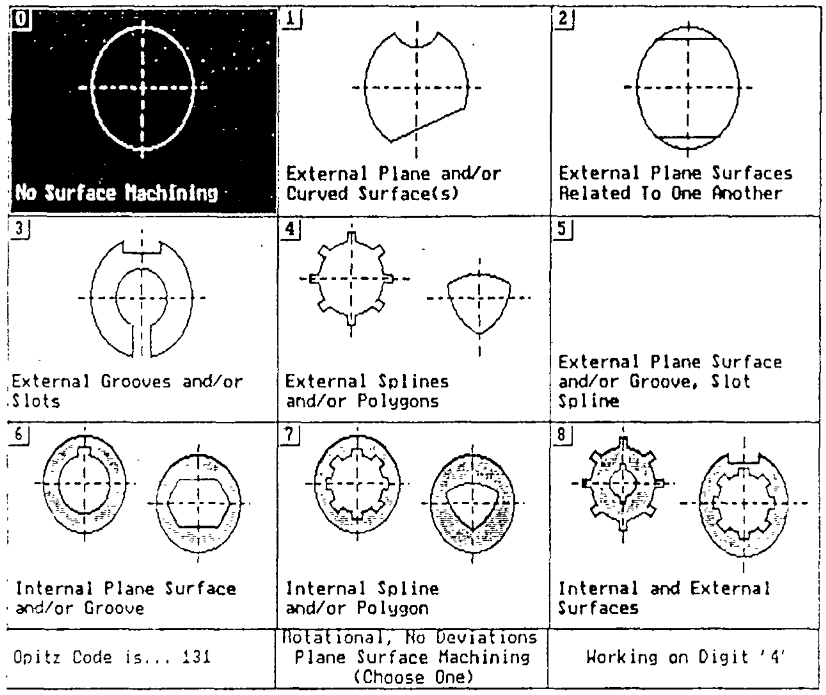


Figure 29: Opitz Module Operation

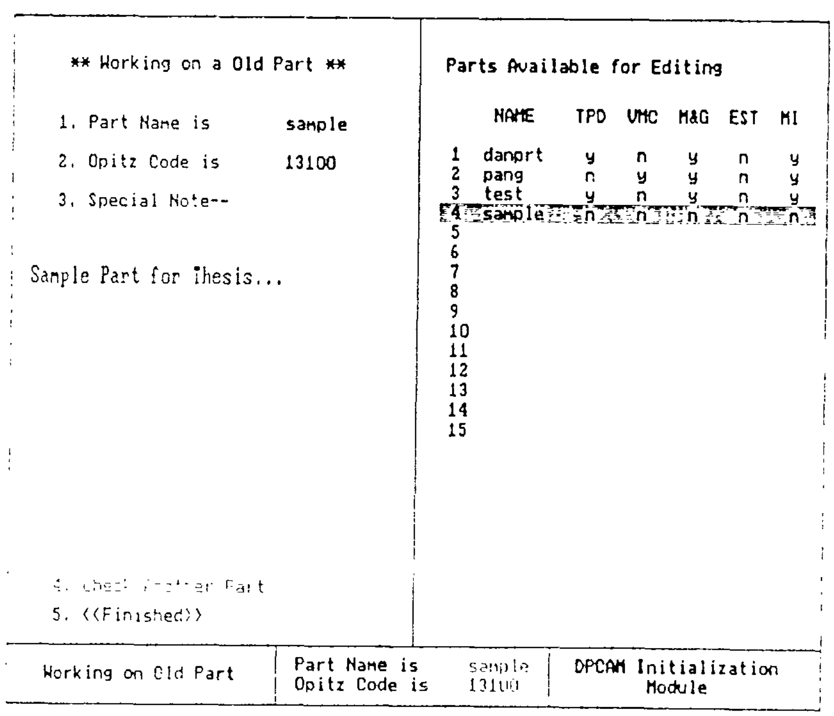


Figure 30: DPCAM *Old* Part Option

Next, DPCAM presents the user with the menu of possible applications. (figure 32) The menu structure presents them in their intended order of use. First in the list is Part Evaluation.

Upon entering the Part Evaluator, the user must make two choices; to "create a new" or "edit an old" evaluation (figure 33) and whether to use "automatic" or "manual" mode. (figure 34) The first choice is self-explanatory. If "new" is chosen, all initial values are set to zero. When "old" is chosen, the initial values are read from an already created Machining Index file for editing. In choosing between "Automatic" and "Manual" mode, the user is letting the computer know whether he wants assistance in evaluating or wants to work on his own. If an "Old" evaluation is being edited, the initial choice is automatically used again. In our case "new" and "automatic" are chosen. Operation of the module is, from then on, straight forward. The user highlights the item of interest and presses "Return". The computer responds with a window complete with a pictorial representation of the item. In addition, since automatic mode was chosen, a suggest index value and reasons for the choice are also displayed. (figure 35) A tally of the results is kept and when all areas have been examined, the final Machining Index displayed. To exit the module, the "Escape" key is pressed. Options available before returning to the applications menu are saving, printing or returning to the Evaluator. (figure 36) In our example, the Machining Index is saved for further use and the program returned to the applications menu.

The next module used is the Estimator. Again the user must let the computer know whether he wants to "start from scratch" or edit an existing estimate. In addition, since the estimator uses the Machining Index as the basis for it's driving mechanism, a warning is given to the user if this item is not

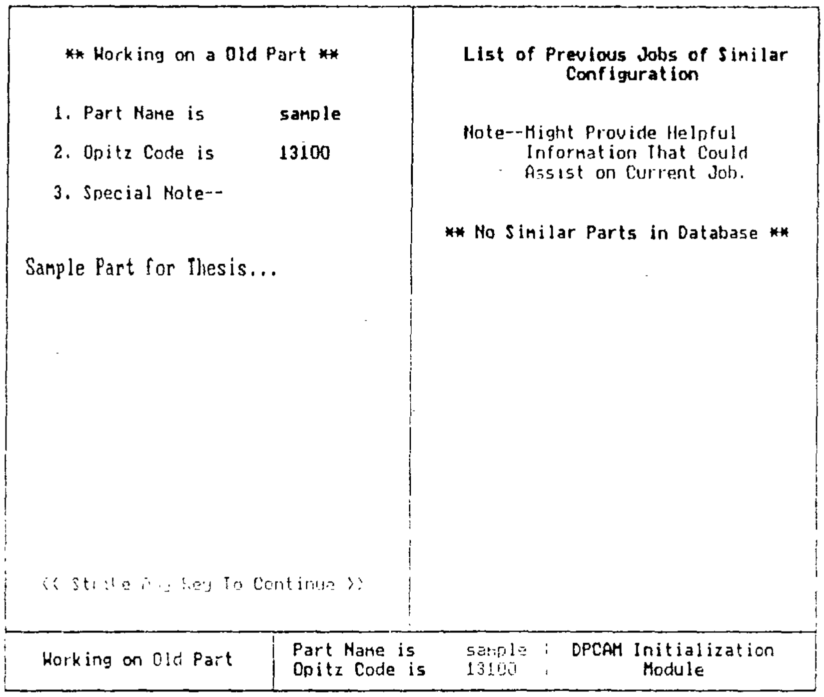


Figure 31: Design Retrieval Output

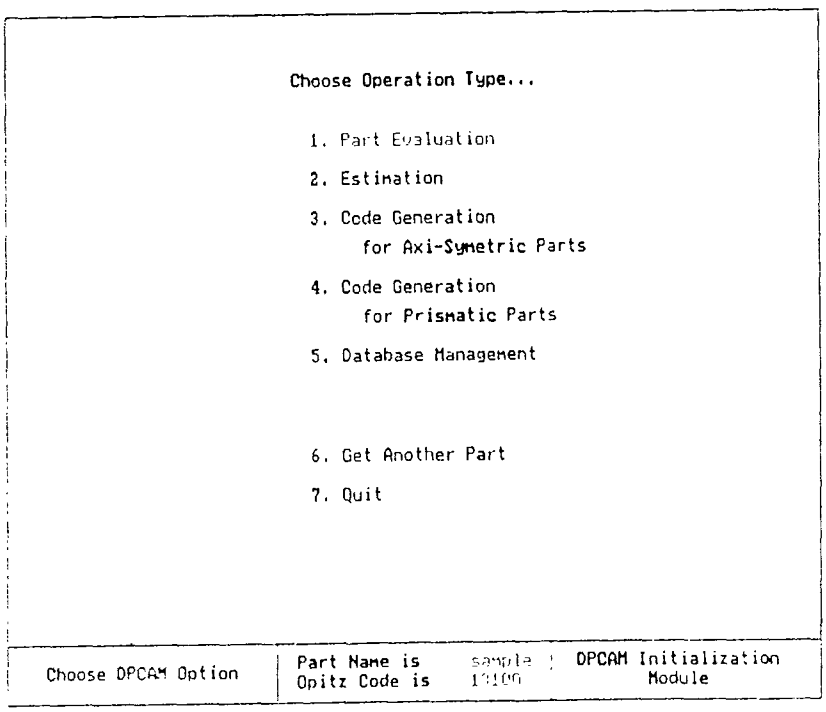


Figure 32: DPCAM Operations Menu

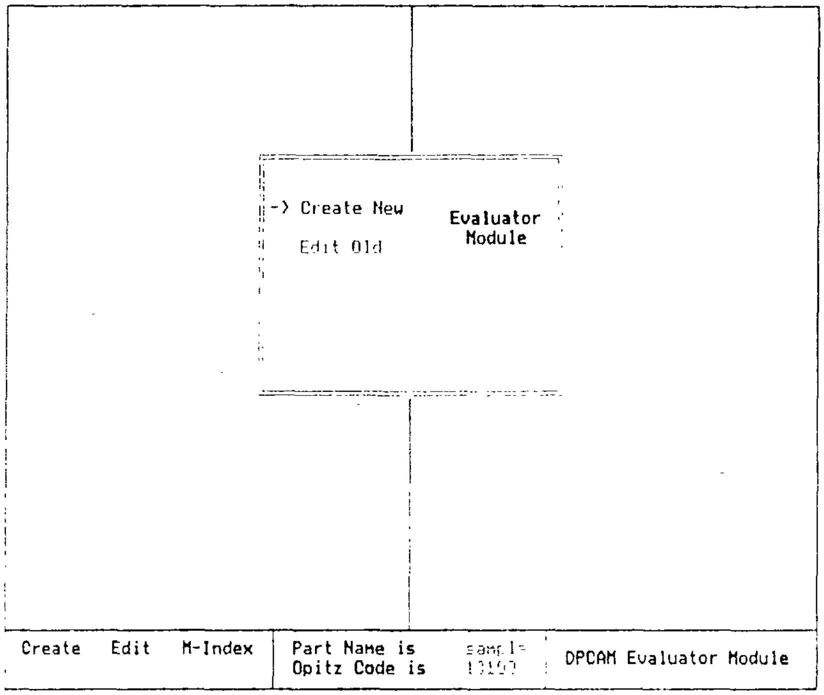


Figure 33: DPCAM Evaluator Title Screen

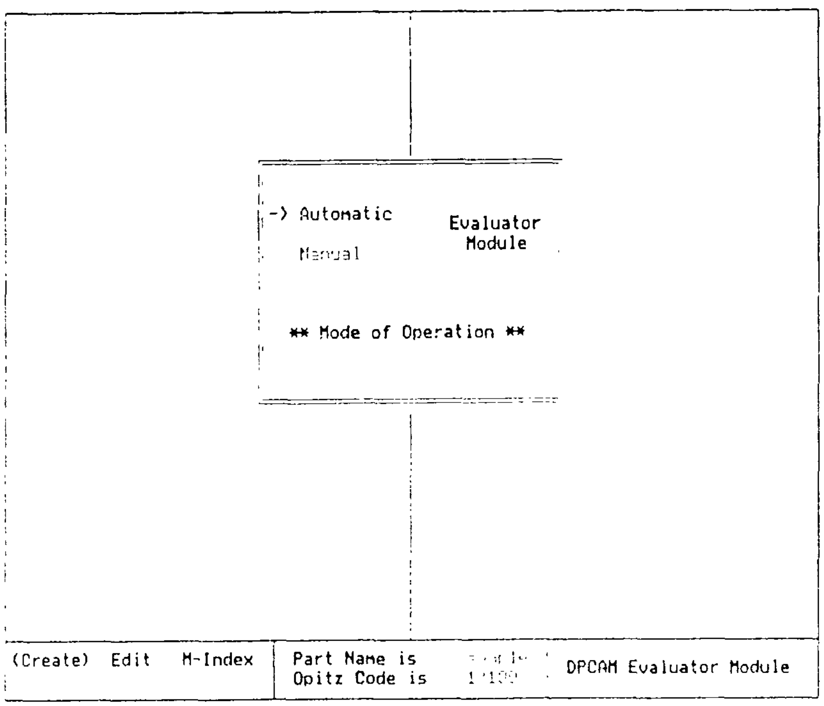


Figure 34: DPCAM Evaluator Options

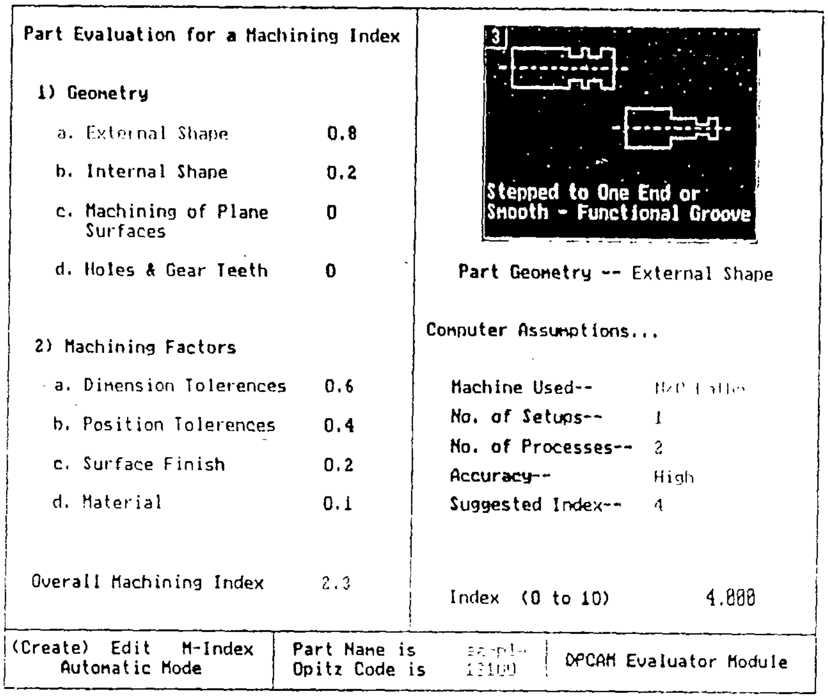


Figure 35: DPCAM Evaluator Operation

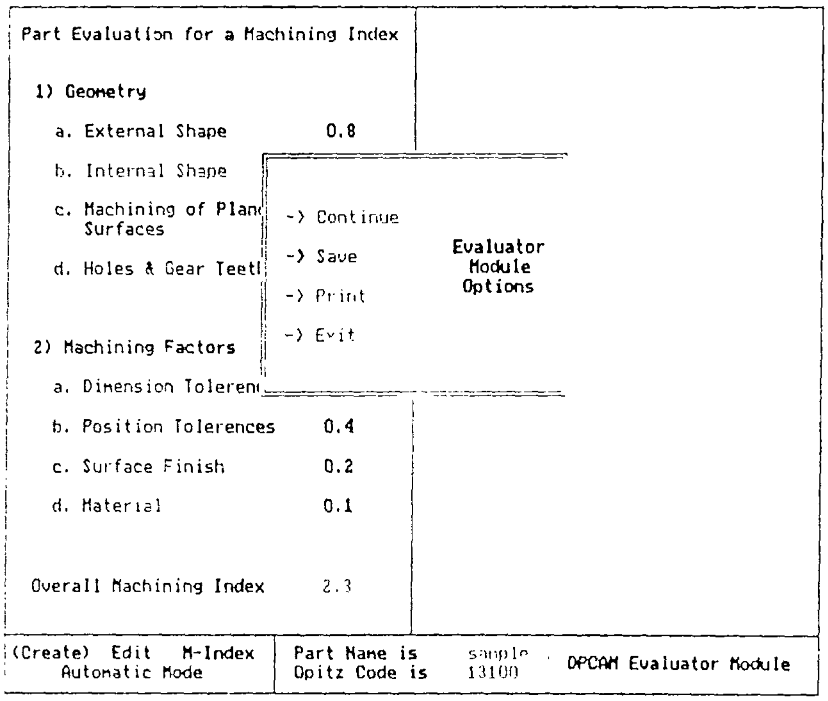


Figure 36: DPCAM Evaluator Exit Options

available. Operation of the Estimator is similar to that of the Evaluator. The user highlights the area of concern and hits "Return". The computer responds with a window containing the appropriate estimate information and its recommended value. (figure 37) The Estimator keeps a running tally not only of the total process time but also keeps track of times for each work element structure and actual machine operation. Once all of the areas have been evaluated and labor and material costs included, the final estimate is ready for printing or saving to disk. As in the Evaluator, the work is saved and the program returned to the applications menu.

If the Machining Index is such that the part is "manufacturable" and the contractor is satisfied with the estimate, machining can begin. To insure that the proper tools are available for the job, the Tool Database is accessed. This utility contains libraries of all the tools available in the shop. (figure 38) Once it is ascertained that the proper equipment is on hand, the Code Generator for Axi-Symmetric Parts is run.

The Code Generator presents the user with three choices: Create a new part, Edit an existing part or Process an existing part. (figure 39) Since our part is new, Create is chosen. The initial information entered into the computer are those items needed to formulate the machining parameters of the program. They include stock size and initial tool, feed and spindle speed values. (figure 40) Once these items are satisfactorily entered, one can go about creating the desired contours and machinable features of the part, Contours are constructed from two basic building blocks, arcs and lines. If a mistake is made, an "undo" function is provided that erases contour elements in the order they were entered. (figure 41) If a machinable feature is chosen, the computer prompts for the required information and outputs the result to the screen for

1. Preperation

a) Examination and Adjustment of Blueprints — 0.167

b) Creation of Process Plan — 0.167

c) Initial Machine Setup

-Fixture and Jig Selection — 0.500

-Machine and Tool Selection — 0.250

-Rough Cut/Stock — 0.100

d) NC Code Generation — 0.250

Preperation Time — 1.434

Total Time — 2.117

Initial Machine Setup
-Fixture and Jig Selection-

Available Fixtures and Jigs and Suggested Times...

for N/C Lathe...

Collet	---	0.50 hr.
3-Jaw Chuck	---	0.50 to 1.00 hr.
4-Jaw Chuck	---	0.50 to 1.00 hr.

for VMC...

Vise	---	0.333 hr.
Table & Clamp	---	0.50 to 2.00 hr.
Index Table	---	0.50 to 2.00 hr.
Angle Plate	---	0.50 to 2.00 hr.
Custom Fixture	---	0.50 to ??? hr.

Time 0.500

(Create) Edit Estimate | Part Name is sample | Opitz Code is 13100 | DPCAM Estimator Module

Figure 37: DPCAM Estimator Operation

/Tool/Turning/
/Library/

1. Show
2. Change
3. Add
4. Delete
5. Create
6. Print
7. Exit

TOOL LIBRARY

No.	Tool_Name	Dia/Wid.	Length	Others
101	BORING_BAR	0.49	1.31	SDT-50-3
102	BORING_BAR	0.615	1.5	SDT-62-3
103	BORING_BAR	0.74	1.87	SDT-75-3
104	BORING_BAR	1.25	14	BNV-TUR-100-32
105	BORING_BAR	1.5	14	BNV-TUR-125-32
201	GROOVE_CUTTER	0.125	2	GTB-062
202	GROOVE_CUTTER	0.125	14	GTB-180
203	GROOVE_CUTTER	0	6	SD-TMR-12-3
301	THREAD_TOOL	0.5	1.25	TR_60,GTB-046
302	THREAD_TOOL	0.625	2	TR_60,GTB-062
303	THREAD_TOOL	1.374	14	TR_60,GTB-180
304	THREAD_TOOL	0	6	TR_60
401	R_SURFACE_CUT.	1	4.5	80,CCLPR-12-4B
402	R_SURFACE_CUT.	1	4.5	80,CCLPR-12-4B
403	R_SURFACE_CUT.	1	4.5	80,MVLNR-12-3
404	R_SURFACE_CUT.	1	4.5	35,MCLNR-12-4
405	R_SURFACE_CUT.	1	4.5	35,MCLNR-12-4
406	R_SURFACE_CUT.	1	4.5	55,MDJNR-12-4
501	DRILL	1.187	3.56	VWDR-118-125
502	DRILL	0.75	2	VWDR-075-100

Use PgUp & PgDn to show tools, Esc to escape

Figure 38: DPCAM Equipment Manager Operation

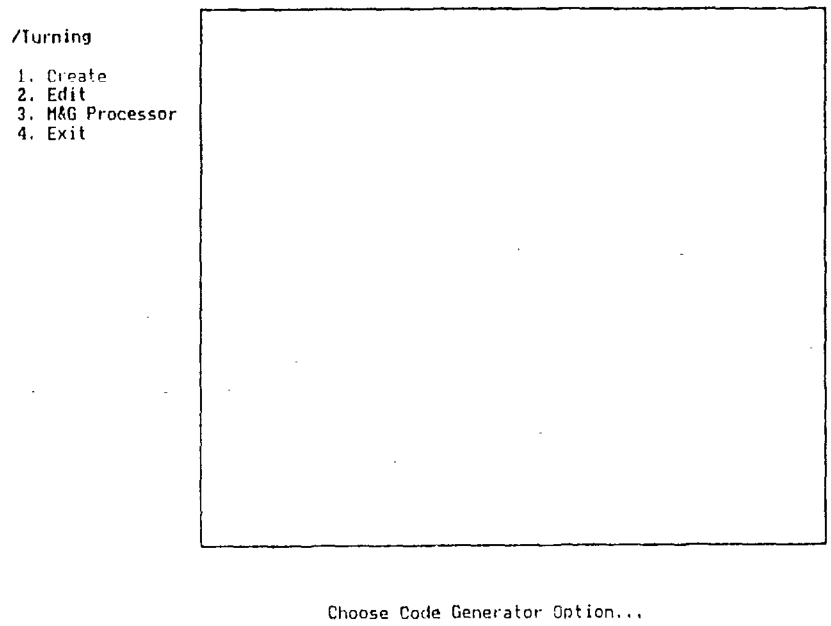


Figure 39: DPCAM Code Generator Options

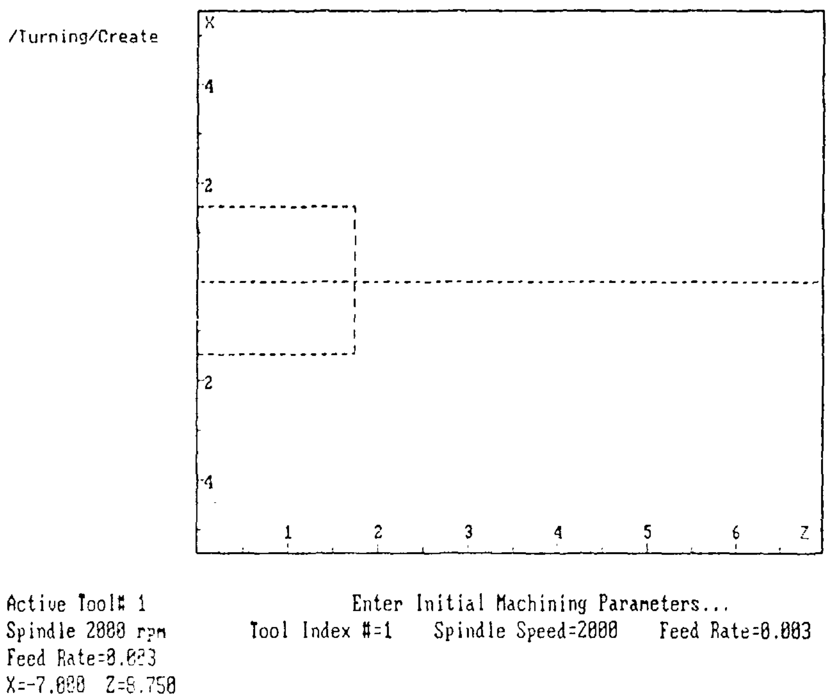


Figure 40: Code Parameter Initialization

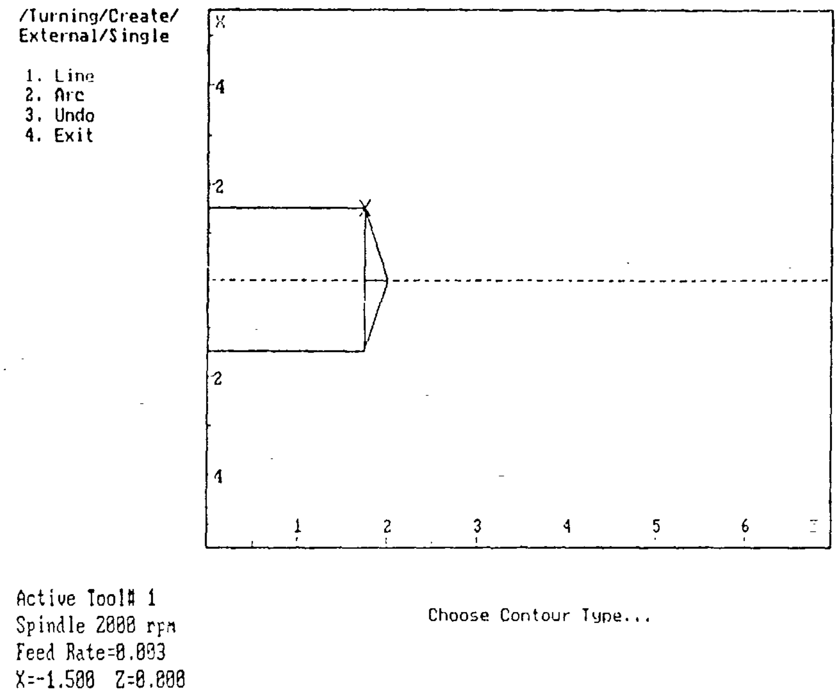


Figure 41: Contour Entry

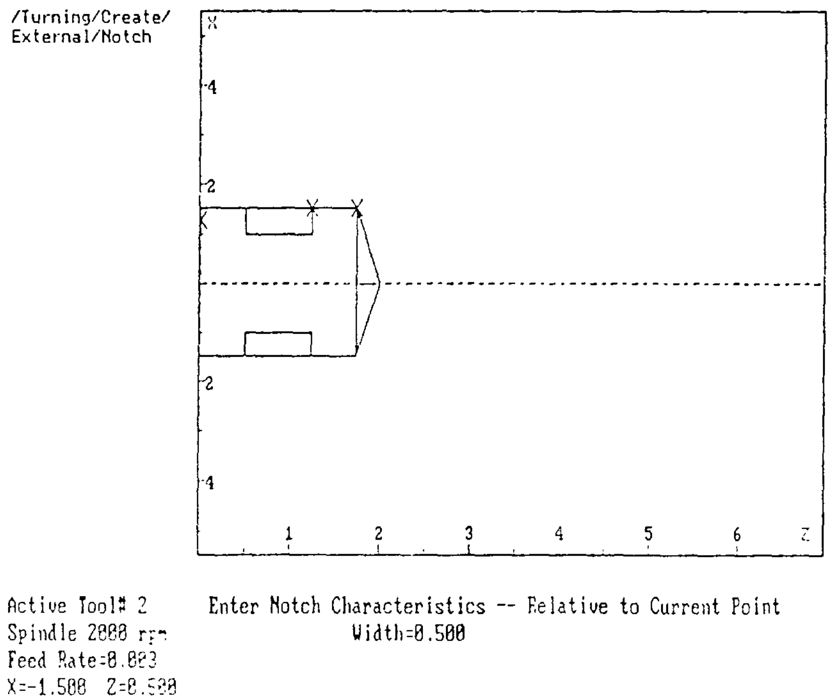


Figure 42: Feature Entry

acceptance or rejection. Examples of machinable features are bores, grooves and threads. (figure 42) Once the part is completed, the created utility is exited and the Part File saved. (figure 43)

If a similar Part File had existed, the Edit function could have been used to modify it to the desired configuration. This is done by stepping through the existing Part File and adding, deleting or modifying where necessary. (figure 44)

When a satisfactory Part File is finished, the Processor utility is used. The processor converts the Part File into M+G code properly formatted for a Fanuc OT controller; the one currently available on the shop floor. Options are provided for viewing, printing and saving the M+G code. (figure 45) When the time to machine arrives, the Processor's transmission function can sends the code into the Turning Center's controller for use. (figure 46)

All work needed from DPCAM now complete, the program is exited by choosing the "Quit" option from the applications menu.

To complete this example, the program is executed by the Turning Center and a finished part produced. (figure 47)

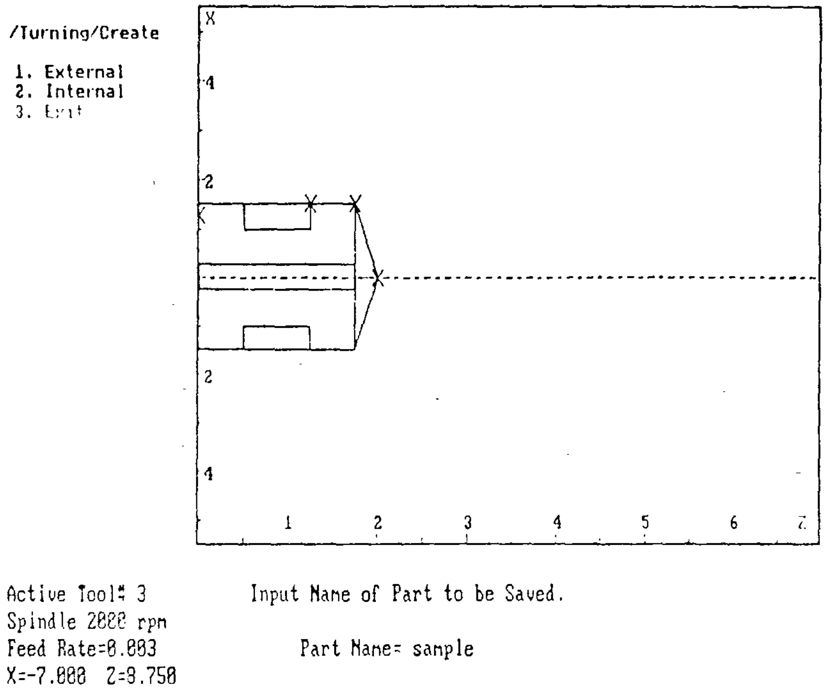


Figure 43: Saving a Part File

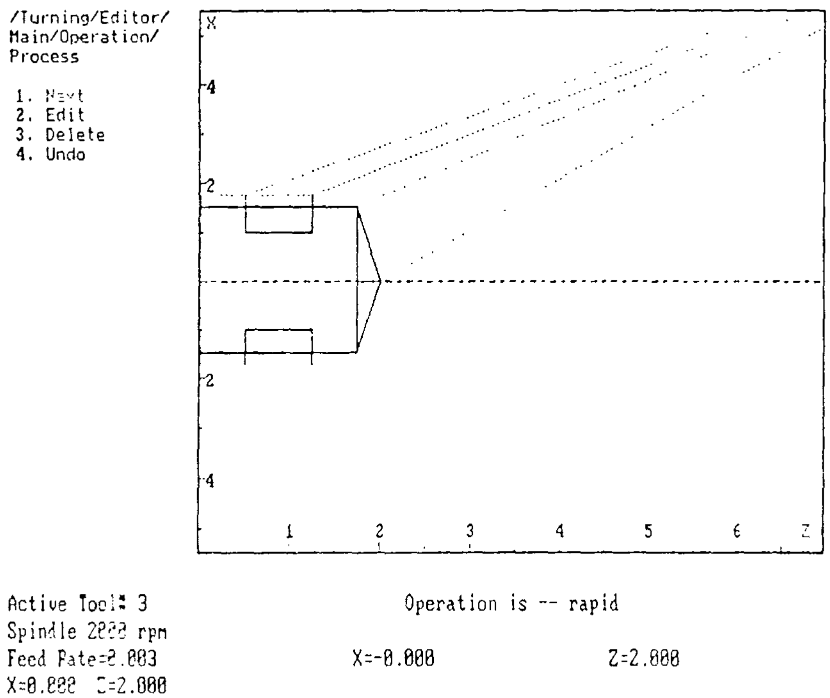


Figure 44: Editing Operations

/Turning/M&G/
Processor/View

1. PgUp
2. PgDn
3. Exit

```
N1  G00 X-7.000 Z8.750 ;
N2  T0101 ;
N3  S2000   M3;
N4  F0.003 ;
N5  M08 ;
N6  ( EXTERNAL ) ;
N7  G00 X-1.500 Z1.750 ;
N8  ( SINGLE ) ;
N9  G01 X0.000 Z1.750 ;
N10  G01 X0.000 Z2.000 ;
N11  G01 X-1.500 Z1.750 ;
N12  G01 X-1.500 Z0.000 ;
N13  ( ENDSINGLE ) ;
N14  ( RAPIDMOVE ) ;
N15  G01 X-1.750 Z0.000 ;
N16  G00 X-1.750 Z1.250 ;
N17  G01 X-1.500 Z1.250 ;
```

Choose Viewing Option...

Figure 45: Processor Operations

/Turning/M&G/
Transmit

Before Transmission Begins, Make Sure Cables are Connected
and NC Machine is Ready. Hit Any Key Begin...

Figure 46: Transmission Operations

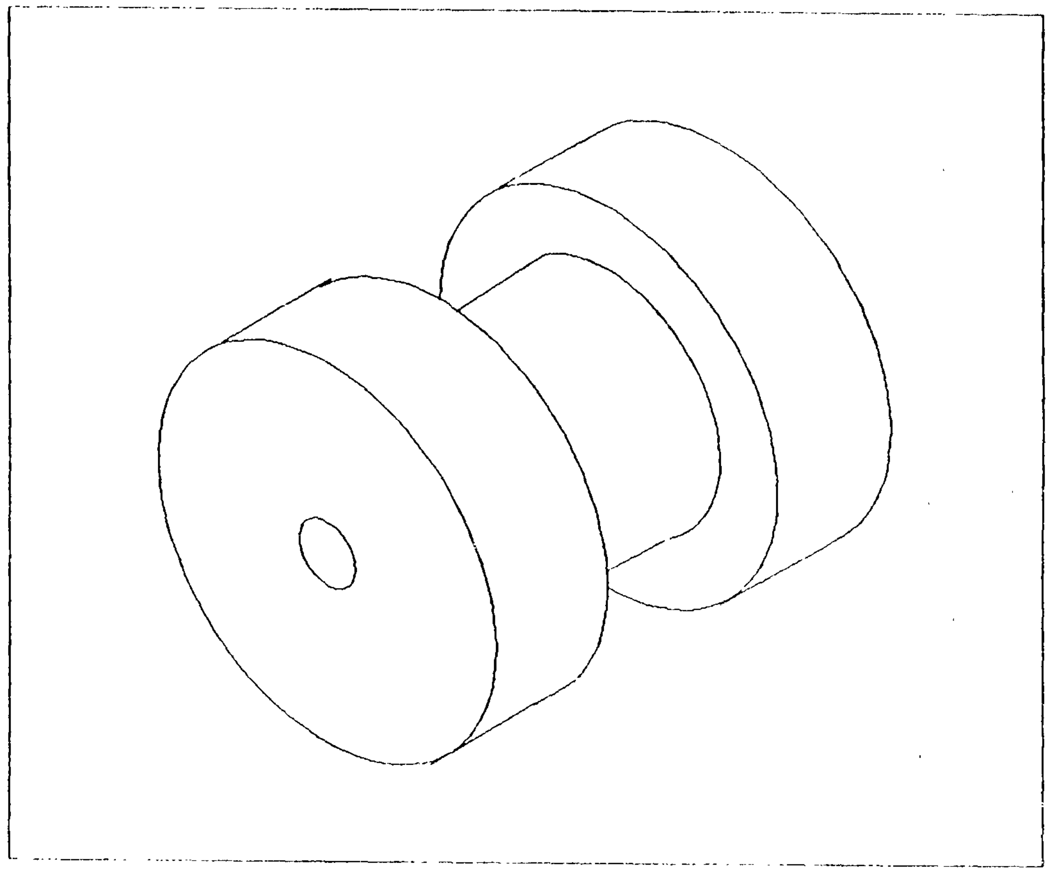


Figure 47: Finished Part

Chapter Nine: Suggestions for Future Work

DPCAM is currently implemented at the University of Maryland's Flexible Manufacturing Laboratory. This is a "test" version, however, and there is room for improvement.

Improved Knowledge Base

The Evaluator and the Estimator both rely on a knowledge base as the basis for their suggestions to the user. Thus, these modules can only be as accurate as the information that forms their evaluating foundations. The knowledge currently incorporated into DPCAM was compiled in short one-on-one discussions with machinists at the Flexible Manufacturing Laboratory's machine shop. While this data is sufficient for a test version, it is not accurate enough for everyday operations.

Improving the knowledge bases requires a historical study of manufacturing operations in the shop of interest. First the manufacturability characteristics of the shop should be determined; based on the available equipment and the employed machinist's level of experience. This manufacturability breakdown can then be subdivided into the desired number of steps and, in combination with the RPC's work element structure, used as the basis for an estimate time study. Once such a study is completed, the new data can be inserted into the Evaluator and the Estimator.

More Complete Automation

Once an accurate set of knowledge bases are in place and confidence is built up in the automatic systems included in DPCAM, the human supervisory role in some modules could be phased out. This especially comes into play in the Evaluator and Estimator modules where final values could be arrived at with much less intervention.

Revised Code Generation/Equipment Management

An "easier to use" Code Generation module could be developed that would provide a more automatic generation process. This would result by removing some of the burden in the areas of process, speed and feed selection through the incorporation of a set of "machining rules". With a these modifications made to the system, the user would simply need to enter blueprint information in the desired cutting order to create the necessary M+G codes.

A more powerful Equipment Manager, with provisions for tool-life predictions could be integrated with the Code Generator. This would allow for automatic tool selection as well as warnings when a tool is about to fail or needs to be sharpened.

Chapter Ten: Conclusions

This research has shown that a considerable gap exists between the goal of Rapid Automated Prototyping and the currently available commercial CAM systems. The purpose of this work was to provide an intermediate step toward total automation for small-scale manufacturing. This concept has matured into a software package called DPCAM. The system strived to provide the user with a tool closer to the true definition of CAM then commercially available systems. This was achieved by applying state-of-the-art techniques developed for the total automation problem to the "intermediate" solution. The modules incorporated into DPCAM include:

-Design Retrieval

-Part Evaluation

-Estimation

-NC Code Generation

-Equipment Management

These options were tied together by a user-friendly graphical interface. The importance of this feature can not be overemphasized, since a "hard to use" program will not be applied to it's fullest potential. Programming was done in the "C" language on a "PC" compatible computer equipped with an Enhanced Graphics Adapter.

Among the advanced features incorporated in the development of DPCAM are:

-Group Technology including a Graphical-based Opitz Code Generation Module

-Concise Measurement of a Part's Manufacturability in the form of a Machining Index

-Estimation Utility suited for a wide range of jobs based on Engineering Principles

-Equipment Management System that allows for Tool Management for N/C Machine Tool Magazines as well as on the Shop Floor

While these items are unique to DPCAM, individual CAM packages can be found which approach these capabilities. The feature which makes DPCAM a viable intermediate "solution" above existing systems is the fact that all of the individual operations are integrated into a single software package. An advantage offered by this single-system operation is the removal of chance data loss associated with shuffling from system to system. In addition, through DPCAM, a user has access to a complete set of CAM activities with out having to learn how to operate multiple software packages.

A number of sample parts have been produced, using a test version of DPCAM at the University of Maryland's Flexible Manufacturing Laboratory. Depending on the part manufactured, a time savings of up to 50% has occurred. This savings is a result of the partial automation of the small-scale manufacturing process that DPCAM encourages.

In order to move DPCAM out of the lab and onto the commercial shop floor, additional work must first be done in the Estimator and Evaluator. To reach the level of accuracy needed for the operational environment, the module's knowledge bases need to be enhanced. This can be achieved by an extensive study of the particular shop's manufacturing activities. With this new data in place, DPCAM will be able to fulfill it's goal as a viable intermediate step toward the total automation of the Rapid Prototyping Center.

Appendix A: Opitz Code for Rotational Parts

Contains information concerning parts in Opitz families "0", "1", "2", "3", and "4." Drawn from A Classification System to Describe Workpieces, Part 1 by H. Opitz.

GEOMETRICAL CODE

1st Digit

	Component Class	
0	Rotational Components	$\frac{L}{D}$ 0 5
1		0 5 · $\frac{L}{D}$ 3
2		$\frac{L}{D}$ · 3

2nd Digit

	External Shape, external shape elements	
0		Smooth, no shape elements
1	Stepped to one End or smooth	no shape elements
2		with screwthread
3		with functional groove
4	Stepped to both Ends (Multiple increases)	no shape elements
5		with screwthread
6		with functional groove
7		functional taper
8		Operating thread
9		Others (·> 10 funcuonal diameters)

3rd Digit

	Internal Shape, internal shape elements	
0		Without through bore blind hole
1	Smooth or Stepped to one End	no shape elements
2		with screwthread
3		with functional groove
4	Stepped to both Ends (Multiple increases)	no shape elements
5		with screwthread
6		with functional groove
7		functional taper
8		Operating thread
9		Others (> 10 functional diameters)

4th Digit

	Plane Surface Machining
0	No surface machining
1	External plane surface and/or surface curved in one direction
2	External plane surfaces related to one another by graduation around a circle
3	External groove and/or slot
4	External spline and/or Polygon
5	External plane surface and/or slot and or groove, spline
6	Internal plane surface and/or groove
7	Internal Spline and/or Polygon
8	External and Internal splines and/or slot and/or groove
9	others

5th Digit

	Auxiliary Hole(s) and Gear Teeth	
0	no gear teeth	No auxiliary hole(s)
1		axial hole(s) not related by a drilling pattern
2		axial holes related by a drilling pattern
3		radial hole(s) not related by a drilling pattern
4		holes axial and · or radial and or in other directions, not related
5		holes axial, and or radial and in other directions related by drilling pattern
6	with gear teeth	spur gear teeth
7		bevel gear teeth
8		other gear teeth
9		others

GEOMETRICAL CODE

Digit	1st Digit: Component Class	2nd Digit: Overall Shape	3rd Digit: Rotational Machining	4th Digit: Plane Surface Machining	5th Digit: Auxiliary Hole(s), Gear Teeth, Forming
0	Rotational Components	Around one Axis No Segments: Hexagonal bar	No rotational machining	No Surface machining	No auxiliary holes, gear teeth and forming
1	Rotational Components	Around one Axis No Segments: Square or other regular Polygonal section	External Shape: machined	External plane surface and/or surface curved in one direction	No forming, No gear teeth: axial hole(s) not related by drilling pattern
2	Rotational Components	Around one Axis No Segments: Symmetrical cross-section producing no unbalance	External Shape: with screwthread(s)	External plane surfaces related to one another by graduation around a circle	No forming, No gear teeth: holes axial and/or radial and/or in other directions, not related
3	Rotational Components: $\frac{L}{D}$ 2 with deviation	Around one Axis No Segments: Cross-sections other than 0 to 2	Internal Shape: smooth	External groove and/or slot	No forming, No gear teeth; related by a drilling pattern: axial holes
4	Rotational Components: $\frac{L}{D}$ 2 with deviation	Segments after rotational machining	Internal Shape: Stepped towards one or both ends (Multiple increases)	External spline and/or Polygon	No forming, No gear teeth; related by a drilling pattern: holes axial and/or radial and/or in other directions
5		Segments before rotational machining	Internal Shape: with screwthreads	External plane surface and/or slot and/or groove, spline	Forming, No Gear Teeth: Formed, no auxiliary holes
6		Around more than one axis: Rotational Components with curved axis	Ext. and Int. Shape: machined	Internal plane surface and/or groove	Forming, No Gear Teeth: Formed, with auxiliary holes
7		Around more than one axis: Rotational Components with two or more parallel axes	Ext. and Int. Shape: screwthread(s)	Internal Spline and/or Polygon	Gear teeth, no auxiliary holes
8		Around more than one axis: Rotational Components with intersecting axes	External Shape Elements	External and Internal Spline and/or slot and/or groove	Gear teeth, with auxiliary hole(s)
9		Others	Other Shape Elements	Other	Other

Appendix B: Part Evaluation Knowledge Base for Rotational Parts

Contains information used to automatically arrive at the Machining Index. Values represent the global difficulty of each factor given the equipment and capabilities of a machine shop, on a scale from 1 to 10. The sub-element index values and justifications stated here resulted from discussions with Dion Slater-El, machinist at the University of Maryland's Flexible Manufacturing Laboratory which took place in Fall, 1988. Includes:

- -Geometry (as broken down in the Opitz System)
  - --External Shape
  - --Internal Shape
  - --Plane Surface Machining
  - --Aux. Holes & Gear Teeth
- -Process Factors
  - --Dimensional Tolerances
  - --Positional Tolerances
  - --Surface Finish
  - --Material

Study Procedures

Geometric Information:

In order to determine the proper sub-index values for the various geometric configurations, three items are examined. These three items include: Number of Setups, Number of Processes and the Inherent Accuracy of the Configuration. The sub-index values are influenced mainly by the Accuracy factor; an example of this is that a spline is more "accurate" than a "key way". The number of setups and processes are used to rate features that have similar accuracy requirements.

Machining Factors:

The sub-index values assigned to the Machining Factors are a function of the type and quality of equipment available as well as on the level of machinist experience. Values are arrived at by studying the equipment, employees and class of work done by a shop.

Geometry--External Shape

Opitz Digit	Setups	Processes	Accuracy	Index
0	1	1	--	0
1	1	1	--	1
2	1	2	Med	2
3	1	2	High	4
4	2	1	--	3
5	2	2	Med	4
6	2	2	High	5
7	-	-	V.High	7
8	-	-	Ext High	8

Geometry--Internal Shape

Opitz Digit	Setups	Processes	Accuracy	Index
0	0	0	--	0
1	1	1	Low	1
2	1	2	Med	2
3	1	2	High	4
4	2	1	Low	3
5	2	2	Med	4
6	2	2	High	5
7	-	-	V.High	7
8	-	-	Ext High	8

Geometry--Plane Surface Machining

Opitz Digit	Setups	Processes	Accuracy	Index
0	0	0	--	0
1	1	1	Low	2
2	2	2	Med	3
3	1	2	Low	2
4	>3	-	V.High	7
5	>3	-	V.High	7
6	1	>1	High	5
7	1	-	V.High	7
8	-	-	Ext High	8

Geometry--Auxiliary Holes & Gear Teeth

Opitz Digit	Setups	Processes	Accuracy	Index
0	0	0	--	0
1	1	1	Low	1
2	1	1	High	3
3	>1	1	Low	2
4	>2	2	Low	3
5	2	2	Med	5
6	-	-	--	?
7	-	-	--	?
8	-	-	--	?

Tolerances--Dimensional Accuracy

Geometric Tolerance	Evaluation Factor
>0.1	1
0.01-0.1	3
0.002-0.01	5
0.0005-0.002	7
<0.0005	9

Tolerances--Positional Accuracy

Positional Tolerance	Evaluation Factor
>0.1	1
0.01-0.1	3
0.005-0.01	5
0.001-0.005	7
<0.001	9

Surface Finish

Roughness (micro in.)	Evaluation Factor
>1000	1
250-1000	3
68-250	5
16-68	7
<16	9

Material

Level of Difficulty	Evaluation Factor
Easy (Aluminum)	1
Average (Bronze)	3
Difficult (Med Steel)	5
Very Difficult (Ti Steel)	7
Ext. Difficult (Ni Alloy)	9

Appendix C: Part Estimation Knowledge Base for Rotational Parts

Contains knowledge used to determine process time in Estimation module. This information is determined by dividing up the work base of the machine shop into varying steps of difficulty (the more steps, the better). Using these steps and the RPC work element structure as a framework, the appropriate time average for each of the RPC tasks are entered. This knowledge then becomes the foundation for all DPCAM estimates, with the proper time determined using the DPCAM-created Machining Index as the search driver. The information presented is taken from discussions with Dion Slater-El, machinist in the University of Maryland's Flexible Manufacturing Laboratory which occurred in Fall, 1988.

Note: All Times are in Hours.

Item	Easy	Med	Hard	Extreme
1. Preparation				
a. Blueprints	.167	.333	1.0	>2.0
b. Process Plan	.167	.50	1.0	>2.0
c. Initial Setup				
i. Fixturing	(look at breakdown)			
ii. Tools	.25	.50	1.0	>2.0
iii. Rough Cut	(look at breakdown)			
d. Code Generation				
-by hand	.167	.50	1.0	>2.0
-by computer	.25	.333	.50	>1.0
2. Testing				
a. DryRun/Debug				
-by hand	.25	.50	1.0	>2.0
-by computer	.083	.25	.50	>1.0
b. Inspection	.25	.50	1.0	>2.0
c. Initial Setup	.083	.25	.5	>1.0
3. Production				
a. Rough Cut	(look at breakdown)			
b. Positioning	.017	.083	.25	>1.0
c. Machining	.033	.10	.25	>.50
d. Inspection	.017	.05	.117	>.25

e. Adjustments	.05	.10	.25	>.50
f. Cleanup	(at users discretion)			

Fixturing

Collet	-	.50
3-Jaw Chuck	-	.50 to 1.0
4-Jaw Chuck	-	.50 to 1.0

Rough Cut

First Piece:

"Off the Shelf"	-	.167
Minor Prep	-	.50
Major Prep	-	1.0
Complex Prep	-	>2.0

Additional Pieces:

"Off the Shelf"	-	.017
Minor Prep	-	.083
Major Prep	-	.25
Complex Prep	-	>1.0

Appendix D: Technical Information on Fanuc OT-Model A Control Unit

Contains information regarding the operation of the Fanuc OT-Model A controller found on the Flexible Manufacturing Laboratory's Nakamura-Tome Slant Jr. Turning Center. Included is the total list of M+G codes for the controller and the data transmission parameters.

M+G Code for a Fanuc OT-Model A Control Unit

Code	Function
G00	Positioning, Rapid
G01	Linear Interpolation
G02	Circular Interpolation, CW
G03	Circular Interpolation, CCW
G04	Dwell
G10	Data Setting
G20	Inch Data Input
G21	Metric Data Input
G22	Stored Stroke Limitation, ON
G23	Stored Stroke Limitation, OFF
G25	Spindle Speed Fluctuation Detect, OFF
G26	Spindle Speed Fluctuation Detect, ON
G27	Reference Point Return Check
G28	Return to Reference Point
G30	2nd Reference Point Return
G31	Skip Cutting
G32	Thread Cutting
G34	Variable-Lead Thread Cutting
G36	Automatic Tool Compensation, X
G37	Automatic Tool Compensation, Z
G40	Tool Nose Radius Compensation Cancel
G41	Tool Nose Radius Compensation, Left
G42	Tool Nose Radius Compensation, Right

G50	Coordinate System Setting, Maximum Spindle Speed Setting
G65	Custom Macro Calling
G68	Mirror Image for Double Turrets, ON
G69	Mirror Image for Double Turrets, OFF
G70	Finishing Cycle
G71	Stock Removal in Turning
G72	Stock Removal in Facing
G73	Pattern Repeating
G74	Peck Drilling on Z Axis
G75	Grooving on X Axis
G76	Multiple Threading Cycle
G80 thru G89	CANNED Cycles for Drilling
G90	Outer/Internal Diameter Cutting Cycle
G92	Thread Cutting Cycle
G94	Endface Turning Cycle
G96	Constant Surface Speed Control, ON
G97	Constant Surface Speed Control. CANCEL
G98	per Minute Feed
G99	per Revolution Feed
M00	Program Stop
M01	Optional Stop
M02	End or Program
M03	Normal Spindle Rotation
M04	Reverse Spindle Rotation

M05	Stoppage of Spindle Rotation
M08	Coolant Motor, ON
M09	Coolant Motor, OFF
M10	Chuck Closed
M11	Chuck Open
M23	Chamfering, ON
M24	Chamfering, OFF
M25	Error Detect, ON
M26	Error Detect, OFF
M30	Reset and Restart of Memory Operation
M41	Spindle, LOW Range Command
M42	Spindle, HIGH Range Command
M51 thru M69	(Reserved)
M98	Sub-Program Call
M99	Return to Main Program

Transmission Parameters for Fanuc OT-Model A Control Unit

Using RS232 Transmission Protocol-

Baud: 4800 Databits: 7

Parity: Even Stopbits: 2

Appendix E: DPCAM Output

Contains a sample of the printouts available form DPCAM. Includes output from the Evaluator, Estimator and Code Generator for Axi-Symmetric Parts.

Part Evaluation for sample Opitz Code 13100

1) Geometry

a. External Shape	0.800
b. Internal Shape	0.200
c. Machining of Plane Surfaces	0.000
d. Holes & Gear Teeth	0.000

2) Machining Factors

a. Dimension Tolerences	0.600
b. Position Tolerences	0.400
c. Surface Finish	0.200
d. Material	0.100

Overall Machining Index	2.300

Estimate for sample Opitz Code 13100

1. Preperation

a) Examination and Adjustment of Blueprints	0.167 hr.
b) Creation of Process Plan	0.167 hr.
c) Initial Machine Setup	
-Fixture and Jig Selection	0.500 hr.
-Machine and Tool Selection	0.250 hr.
-Rough Cut/Stock	0.100 hr.
d) NC Code Generation	0.250 hr.

Total Preperation Time 1.434 hr.

2. Testing

a) Dry Run/Debug/Sample Machining	0.083 hr.
b) Sample Inspection/Quality Assurance	0.250 hr.
c) Modification and Correction	0.100 hr.

Total Testing Time 0.433 hr.

3. Production

a) Rough Cut/Stock	0.000 hr.
b) Positioning of Parts in Machine	0.000 hr.
b) Machining Operations	0.000 hr.
c) Quality Assurance of Random Samples	0.000 hr.
# of Samples per 100--	0 per 100
Total Number of Parts in Production Run	0 parts
d) Adjustments for Wear	0.000 hr.
# of Adjustments per Run--	0 per run
f) Clean Up	0.250 hr.

Total Production Time 0.250 hr.

4. Cost Factors

a) Hourly Labor Rate	$25.00 per hour
b) Machine Labor Rate	$25.00 per hour
c) Special Costs	$0.00
d) Material Costs	$1.00 per part

Estimate Overview

Total Time	2.117 hr.
Machine Time	0.083 hr.
Total Cost	$55.00
Cost per Part	$55.00

M&G Codes for sample   Opitz Code 13100

```
N1  G00 X-7.000 Z8.750 ;
N2  T0101 ;
N3  S2000   M3;
N4  F0.003 ;
N5  M08 ;
N6  ( EXTERNAL ) ;
N7  G00 X-1.500 Z1.750 ;
N8  ( SINGLE ) ;
N9  G01 X0.000 Z1.750 ;
N10  G01 X0.000 Z2.000 ;
N11  G01 X-1.500 Z1.750 ;
N12  G01 X-1.500 Z0.000 ;
N13  ( ENDSINGLE ) ;
N14  ( RAPIDMOVE ) ;
N15  G01 X-1.750 Z0.000 ;
N16  G00 X-1.750 Z1.250 ;
N17  G01 X-1.500 Z1.250 ;
N18  ( ENDRAPID ) ;
N19  ( NOTCH ) ;
N20  G01 X-1.750 Z1.250 ;
N21  ( TOOLCHANGE ) ;
N22  G00 X-7.000 Z8.750 ;
N23  T0100 ;
N24  T0202 ;
N25  G00 X-1.750 Z1.250 ;
N26  ( ENDCHANGE ) ;
N27  G75 R0.050 ;
N28  G75 X-1.000 Z0.500 P1500 Q1000 R0.000 ;
N29  G01 X-1.500 Z0.500 ;
N30  ( ENDNOTCH ) ;
N31  G01 X-1.750 Z0.500 ;
N32  G00 X-7.000 Z8.750 ;
N33  ( ENDEXT ) ;
N34  ( INTERNAL ) ;
N35  G00 X0.000 Z2.000 ;
N36  ( DRILL ) ;
N37  G01 X0.000 Z2.000 ;
N38  ( TOOLCHANGE ) ;
N39  G00 X-7.000 Z8.750 ;
N40  T0200 ;
N41  T0303 ;
N42  G00 X0.000 Z2.000 ;
N43  ( ENDCHANGE ) ;
N44  G74 R0.250 ;
N45  G74 X0.000 Z0.000 P0 Q5000 R0.000 ;
N46  G00 X0.000 Z2.000 ;
N47  ( ENDDRILL ) ;
N48  G01 X0.000 Z2.000 ;
N49  G00 X0.000 Z2.000 ;
N50  G00 X-7.000 Z8.750 ;
N51  ( ENDINT ) ;
N52  M09 ;
N53  T0300 ;
N54  M30 ;
```

References

1. Richard E. Billo, "Enhanced Group Technology Modeling with Database Abstractions," *Journal of Manufacturing Systems*, Vol. 7, No. 2, pp 95-106, 1988.

2. Tien-Chien Chang, *An Introduction to Automated Process Planning Systems*, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1985.

3. Sujen Chen, *Structure of a Flexible Manufacturing Protocol for Design and Control of a Vertical Machining Center*, Master's Thesis, University of Maryland, College Park, Maryland, 1988.

4. Dimitris N. Chorafas, *Engineering Productivity Through CAD/CAM*, Butterworths, London, England, 1987.

5. Mike Courtright, *Manufacturability Analysis Software for a Flexible Manufacturing Cell*, Master's Thesis (in preparation), University of Maryland, College Park, Maryland, 1988.

6. Lawrence E. Doyle, *Manufacturing Process and Materials for Engineers, 3rd. Edition*, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1985.

7. C.C. Gallagher, *Group Technology Production Method in Manufacture*, Ellis Horwood Limited, Chichester, West Sussex, 1986.

8. Mikell P. Groover, *CAD/CAM: Computer-Aided Design and Manufacturing*, Prentice Hall, Inc., Englewood CLiffs, New Jersey, 1984.

9. Caroline Hayes, *Planning in the Machining Domain: Using Goal Interactions to Guide Search*, Master's Thesis, The Robotics Insitute, Carnegie Mellon University, Pittsburg, PA, 1987.

10. John Herzog, "Take Full Advantage of Computer-Aided Cost Estimating", *Modern Machine Shop*, Vol. 61, No. 4., pp 92-98, September 1988.

11. Edward G. Hoffman, "Microcomputer CAD/CAM for All Shops, Part ," *Modern Machine Shop*, Vol. 60, No. 5, pp 82-90, October 1987.

12. Edward G. Hoffman, "Microcomputer CAD/CAM for All Shops, Part II." *Modern Machine Shop*, Vol. 60, No. 6, pp 102-107, November 1987.

13. Stuart W. Hubbard, *CAD/CAM Applications for Business*, Oryx Press, Phoenix, Arizona. 1985.

14. Stan Kelly-Booth, *Mastering TURBO C*, Sybex, Inc., Alameda, California, 1988.

15. J.A. Kirk, "The Use of IGES in Rapid and Automated Design Prototyping," *Advances in Design Automation-1988*, ASME, New York, New York, pp 27-32, 1988.

16. D. Kochan, *CAM: Developments in Computer Integrated Manufacturing*, Springer-Verlag, New York, New York, 1985.

17. Stephen G. Kochan, *Programming in C*, Hayden Books, Indianapolis, Indiana, 1983.

18. B. Malakooti. "A Methodology for the Automation of Medium or Small Manufacturing Companies," *Computers in Industry*, Vol. 7, No. 4 ,pp 333-341, Aug 1986.

19. H. Opitz, *A Classification System to Describe Workpieces, Part 1*, Pergamon Press Ltd., Headington Hill Hall, Oxford, 1970.

20. H. Opitz, *A Classification System to Describe Workpieces, Part 2*, Pergamon Press Ltd., Headington Hill Hall, Oxford, 1970.

21. Phillip F. Ostwald, *Cost Estimating, 2nd. Edition*, Prentice Hall, Inc., Englewood Cliffs, New Jersey, 1984.

22. Dachen Pang, *Integrated Numerical Control Manufacturing for Prismatic Parts, Master's Thesis* (in preperation), University of Maryland, College Park, Maryland, 1988.

23. Chan S. Park, "Computer-Assisted Estimating of Nonconventional Manufacturing Costs," *Computers in Mechanical Engineering*, Vol. 6, No. 1, pp 16-25, July/August 1987.

24. Werner P. Rieben, "Future Trends in CNC," *The Carbide Tool Journal*, Vol. 17, No. 1, pp 17-21, January/Febuary 1985.

25. Michael Sava, *Computer Numerical Control*, Reston Publishing Company, Inc., Reston, Virginia, 1983.

26. Herbert Schildt, *Advanced TURBO C*, Osborne McGraw-Hill, Berkeley, California, 1987.

27. Howard B. Singer, "The Real Economics of CAD," *Modern Machine Shop*. Vol. 61, No. 4, pp 78-88, September 1988.

28. H.G. Smart, "Least Cost Estimating with Group Technology," *Journal of Manufacturing Systems*, Vol. 1, No. 1, pp 99-110, 1983.

29. Michael H. Smith, "The New Role of CNC in Computer Integrated Manufacturing," *The Carbide Tool Journal*, Vol. 17, No. 4, pp 10-15, July/August 1985.

30. Rodney D. Stewart, *Cost Estimators's Refernece Manual*, John Wiley & Sons, New York, New York, 1987.

31. J. Sumida, Lecture Notes History 225, University of Maryland, College Park, Maryland, 1988.

32. Leland Teschler, "How PCs are Shaping Cell Control," *Machine Design*, Vol. 60, No. 4, pp 86-92, Febuary 25, 1988.

33. Rajiv Uppal, *Flexible Manufacturing Protocol Driver for a Vertical CNC Machining Center*, Master's Thesis, University of Maryland, College Park, Maryland, 1987.

34. *Fanuc OT-Model A Operator's Manual*, Fanuc Ltd., 1985.

35. *Machining Data Handbook, 3rd Edition, Volume Two*, Machinability Data Center, Cinicinnati, Ohio, pp 18.11-18.13, 1988.

36. *Slant Jr. Turning Center Instruction Operation Manual*, Nakamura-Tome Precision Industry Co., Ltd., Tsurugi Ishikawa, Japan, 1986.