# AUTOMATED COMPENSATION OF PART ERRORS DETERMINED BY IN-PROCESS GAUGING

Kenneth W. Yee
Herbert T. Bandy
Jack Boudreaux
Neil D. Wilkin

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Manufacturing Engineering Laboratory
Automated Production Technology Division
Gaithersburg, MD 20899

NIST

AUTOMATED COMPENSATION OF PART ERRORS DETERMINED BY IN-PROCESS GAUGING

Kenneth W. Yee
Herbert T. Bandy
Jack Boudreaux
Neil D. Wilkin

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Manufacturing Engineering Laboratory
Automated Production Technology Division
Gaithersburg, MD 20899

June 1992

1. INTRODUCTION

The Quality in Automation (QIA) program of the Manufacturing Engineering Laboratory of the National Institute of Standards and Technology is a program to develop a quality-control/quality-assurance system using deterministic manufacturing principles. The premise is that most errors in the manufacturing process are repeatable and predictable and, therefore, can be compensated. The program uses a quality-control architecture with multiple feed-back loops (figure 1) to control the process for small-batch manufacturing.
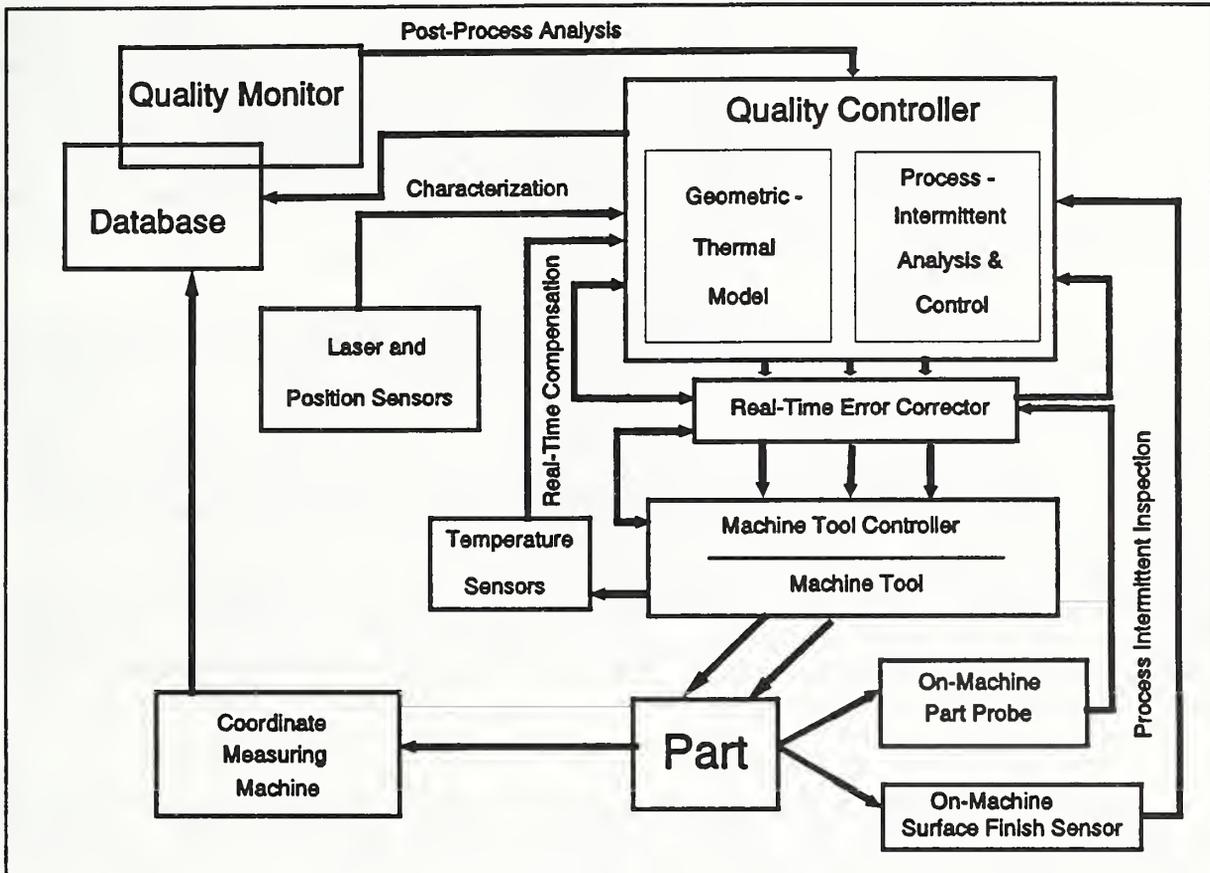

Figure 1. Quality in Automation Architecture

As previously described [1,2,3], the function of the **Real-Time** control loop is to monitor the machine tool and the cutting process and to modify the tool path, or cutting parameters to achieve higher dimensional and form accuracy and surface quality of finished parts. Real-time error compensation is achieved by implementing tool-path modification, using a Real-Time Error Corrector (RTEC), using errors predicted by kinematic and geometric-thermal (G-T) models of the machine-tool errors. The

1

models are derived from the structure of the machine and **Pre-Process Characterization** machine measurements. The function of the **Process-Intermittent** control loop is to determine workpiece dimension and form errors not compensated by the real-time loop in a semifinish cut, and compensate for them in the finish cut. One proposed method is to use fast probing (an additional function of the RTEC) with a touch-trigger probe for in-process (on-machine) gauging, and modification of the tool-dimension offsets or NC-part program (EIA RS-274D) coordinates for implementing compensation. We will describe the use of the RTEC as an alternate means of compensation in this loop which requires no change to the NC program.

The **Post-Process** control loop is used primarily to verify and optimize the other two loops. Finished parts are measured, after a batch is finished, on a coordinate measuring machine (CMM) to determine residual errors in part features. Systematic errors in specific features will be used to modify the algorithms used in the process-intermittent loop, or, in the long term, to modify the parameters in the real-time G-T model. Alternatively, the pre-process characterization may need to be revised.

An NC probing program is required for each inspection (gauging) session (in the process intermittent loop or after the finish cut) for each part processed through the QIA system. A software procedure was developed to facilitate the generation of the required probing program. A commercially available Computer Aided Design (CAD) system with a representation of the part is first used to select the probing points and develop a file of probe-path end-point coordinates. Then a program, described later, generates the NC probing program. After a semifinish cut, coordinates generated by the part probing are used by the CAD system to determine the deviations from the nominal dimensions. These "errors" are passed to a program which determines the corrections to be inserted by the RTEC during the finish cut.

2. ERROR COMPENSATION USING THE REAL-TIME ERROR CORRECTOR

THE RTEC

The design of the RTEC has been previously described [2,4]. It was originally developed to implement compensation of G-T errors in the real-time loop and fast probing for the process-intermittent loop. Fast-probing is on-machine probing using a touch-trigger probe at feed rates 10- to 20-times faster than commonly used. The RTEC is a *black box* that is inserted between the position feedback element of the axes of a machine tool and the machine-tool controller (MTC) as shown in figure 2. The device independently counts the
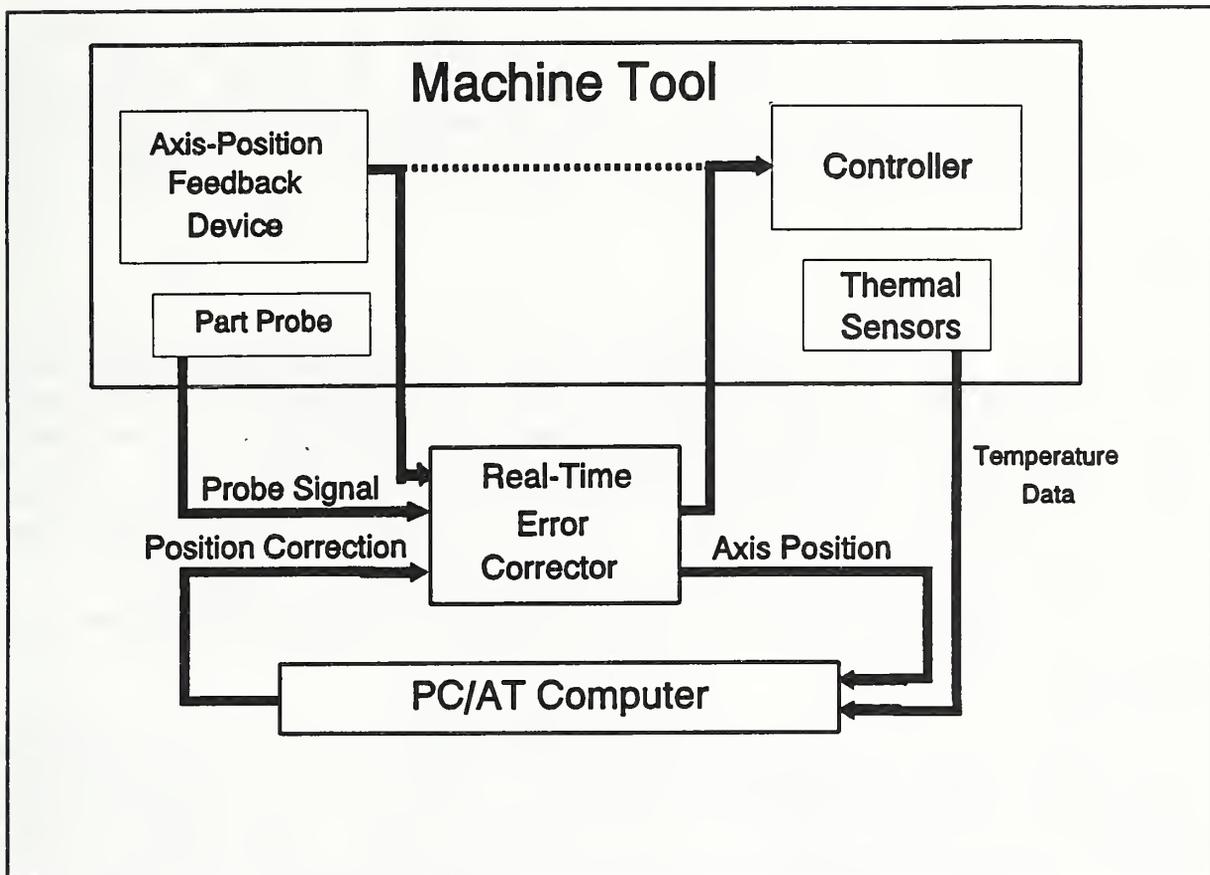
Figure 2. Real-Time Error Corrector and Machine Tool

unaltered signals from the feedback element (the MTC counts the corrected signal) to produce the machine position and sends this position data to a PC-compatible computer, called the Quality Controller, which records the data. There is a quadrature decoder and microcomputer for each axis. The RTEC alters the tool-path by modifying the square-wave signals from "encoder-type" feedback devices before they go to the MTC. Pulses are added which alter the position count in the MTC, causing the machine axis to go to a slightly different position to compensate for the predicted errors. The correction counts to be added are determined by the Quality Controller using the G-T model and the current machine temperatures and axes positions. For the machine tool in use, a two-axis Monarch Metalist[1] turning center with a GE-2000T CNC controller, the position is altered by 4 μm for each correction count.

In fast-probing, the RTEC captures the axis positions in response to the probe trip and sends them to the PC. At the 2.5 m/min feed rate used, the position must be captured within 25 μs to avoid introducing a position error greater than the position resolution of this machine, which is 1 μm. Most MTCs are not designed to respond to probe trips rapidly enough to allow fast probing.

ON-MACHINE GAUGING

In general, on-machine gauging, using a touch-trigger probe or other probe, can only detect part errors caused by process-induced errors, not machine geometric errors. Most geometric errors will be reproduced during the probing operation and will not be "seen" (i.e., the same incorrect path is traced by the tool and the measuring probe). Therefore, only process-induced errors, caused by tool dimension, tool wear, tool and/or part deflection, etc., can normally be detected. The part is measured after a semifinish cut which uses the same cutting parameters (speed, feed, and depth of cut) as will be used in the finish cut to "standardize" process-dependent effects such as cutting-force induced tool or part deflection.

In the QIA system, real-time G-T compensation is not implemented during probing since imperfections in the real-time compensation loop would then be hidden. Also, the RTEC microcomputer is not capable of doing both error compensation and fast probing simultaneously. However, compensation for the errors predicted by the G-T model, using temperatures recorded with the run, can be applied to the probing data off-line.

The part geometry and dimensions are measured using fast probing. In fast probing, the machine moves the probe across the part boundary at 2.5 m/s. The MTC "canned" probing program (e.g., a "G51" command for our MTC) is not used. An NC program is written to move the probe in a manner that ensures the probe trips during the constant velocity portion of the motion. Otherwise, a timing variation would be introduced which would cause a position variation due to the delay between probe trip and position capture. The stand-off and over-travel distances are selected depending on the machine acceleration and deceleration capabilities. Our machine requires a stand-off of 5 mm and overtravel of 2.5 mm. This gives about 2 mm of constant velocity travel. The probing locations are selected using a commercial computer aided design (CAD) package that runs on the Quality Controller. The number and location of the measurement points are selected and entered. The CAD package then determines the starting and ending points, in part program coordinates, for each probing location. These end-point positions for the probing moves are then used by the program that automatically generates the NC-part-program which moves the probe. These operations will subsequently be described further.

4

A PART-FORM ERROR

We will now describe the compensation of an error in part form detected by on-machine fast probing. In turning a hemispherical nose on a part, any error in the circular shape of the tool nose directly translates into an error in the shape of the part (see figure 3a.). While cutting the hemisphere, the tool nose "rolls around" the part. Every point on a 90° portion of the tool nose cuts the part. The cut is commonly made by executing a single NC instruction, a "G02" circular interpolation instruction. If the tool nose deviates from the ideal circular shape, such as with a flat spot shown in figure 3b., the part will have a corresponding "bump" on the circular cross section of the hemisphere. To demonstrate this phenomenon, we ground a flat spot on a tool nose, removing about 25 μm from the tool to simulate a worn tool. This should result in a "bump" on the nose of the part of the same magnitude.

THE DETERMINATION OF PART-FORM ERRORS FROM PROBING DATA

The probing data, in machine coordinates, is fitted to the nominal geometry, which is a circle for this part, using a least-squares routine written in C. The deviation at each probing location from this circle is then used to determine the residual errors. Figure 4 shows the outline of a part, with a 38.1-mm radius hemispherical nose, generated by the CAD package. A magnified plot of the residual errors (deviation from the circle fit) after a semifinish cut of the hemisphere, cut with the "worn" tool, is shown in figure 5. The nominal dimension of this semifinish cut is 0.254 mm in radius larger than the finish part. A "bump" is clearly visible just above the centerline. The height of the "bump" is the sum of the largest positive error and the negative error at the axis. The "bump," which is about 23 μm high, is essentially the same as the depth of the flat on the tool nose.

To determine part-form errors, more probing points must be used than are required to measure just dimensions. In this example, 20 points were probed on the 90° quadrant of the part-nose circle. As can be seen from figure 5, about 6 points fall on the "bump." This number is near the minimum that allows a reasonable estimate of the shape of this "bump." Fast probing is essential to implementing part-form compensation without wasting excessive machine time.

COMPENSATING FOR THE PART-FORM ERROR

Next, the residual errors of figure 5 are converted to a table of error magnitudes (in micrometers) versus angle from the part axis (in radians). This table is used by the control program [3] running on the Quality Controller which interfaces with the RTEC.

a.

Tool
Nose

Part
Nose

b.

Worn
Tool
Nose

Bump
on
Part
Nose
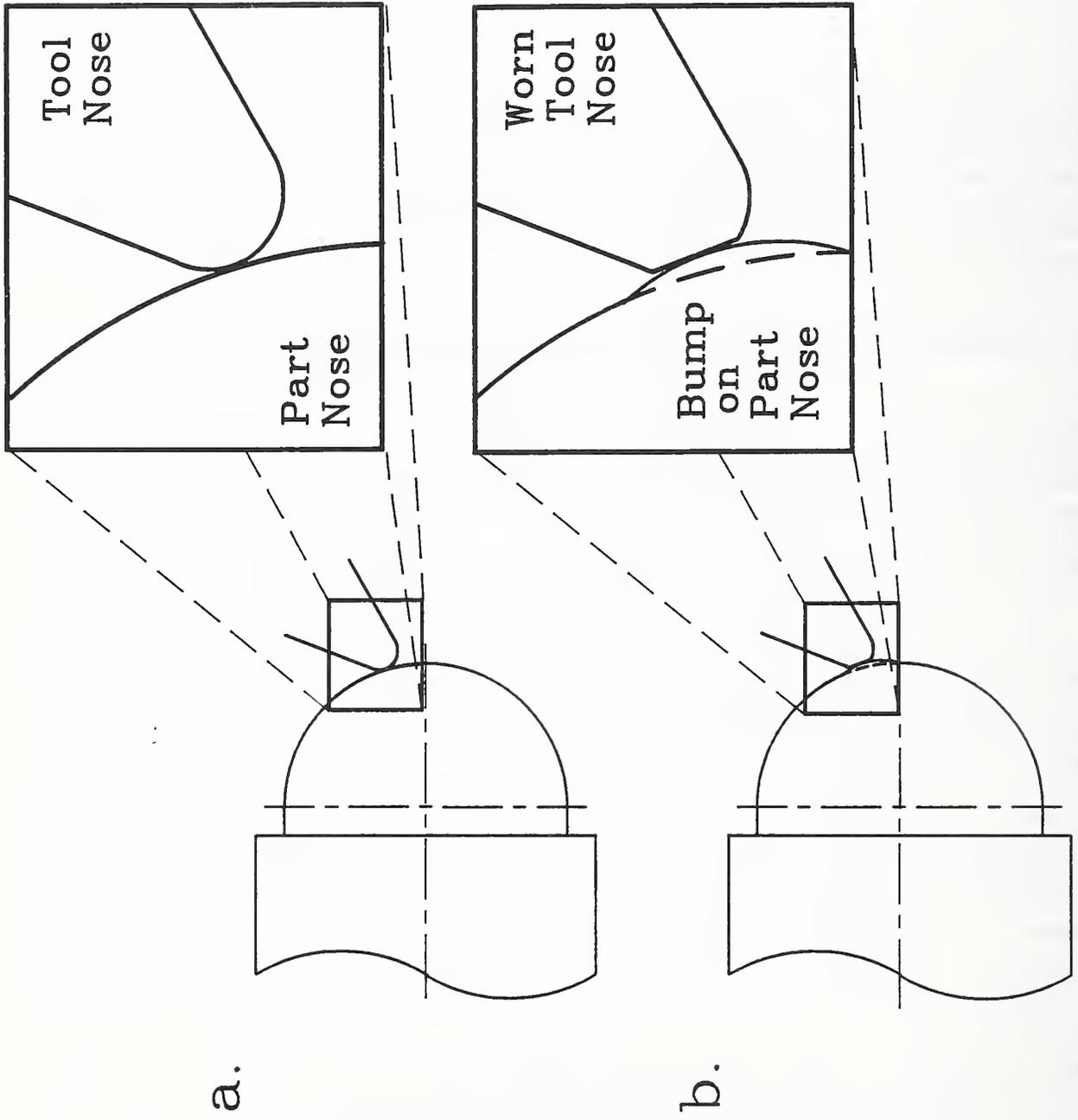
Figure 3. Part-Form Error from a Worn Tool Nose

QUALITY IN AUTOMATION

Process Intermittent Gauging

Error Plotting

Error Bars X250

25.4un X250 ━━━
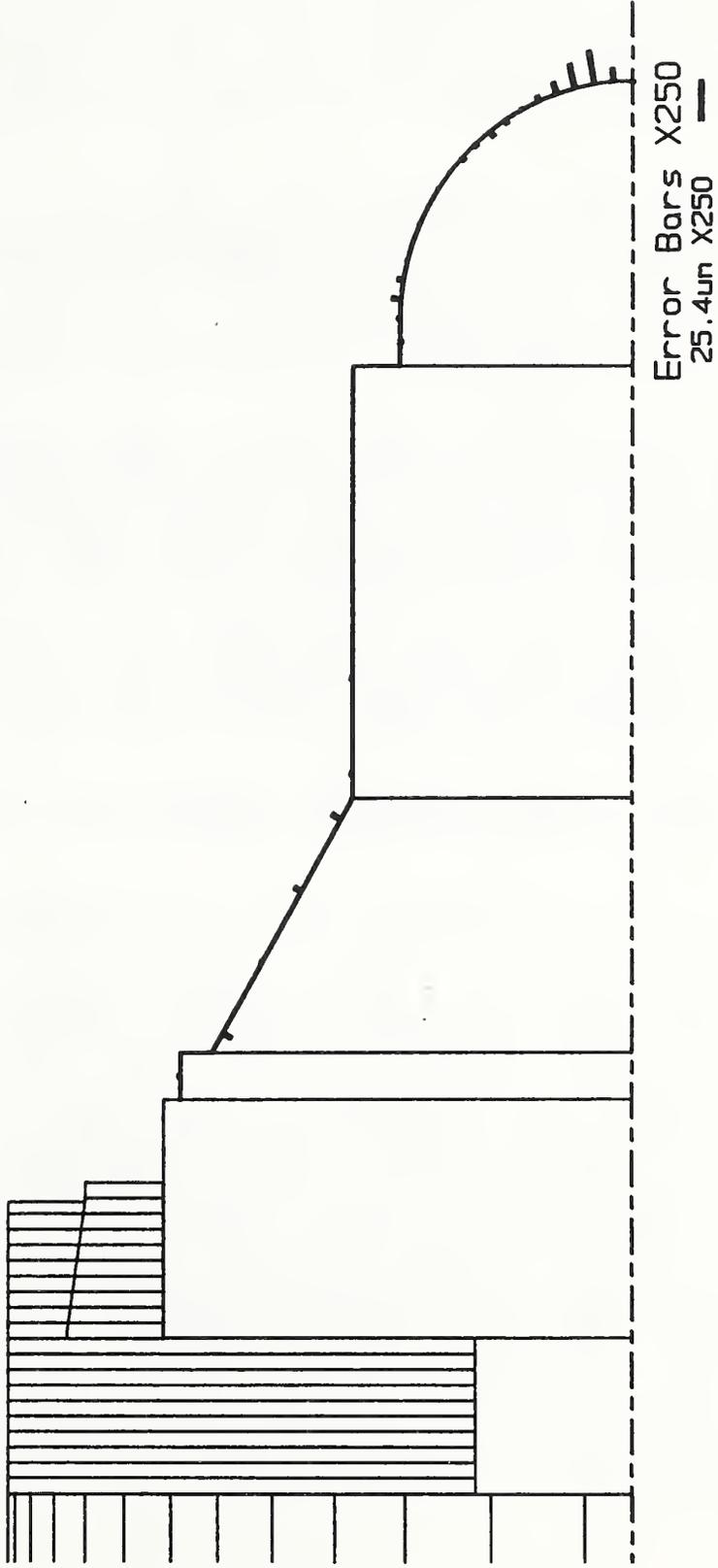
Figure 4. CAD Program Display of Part

7
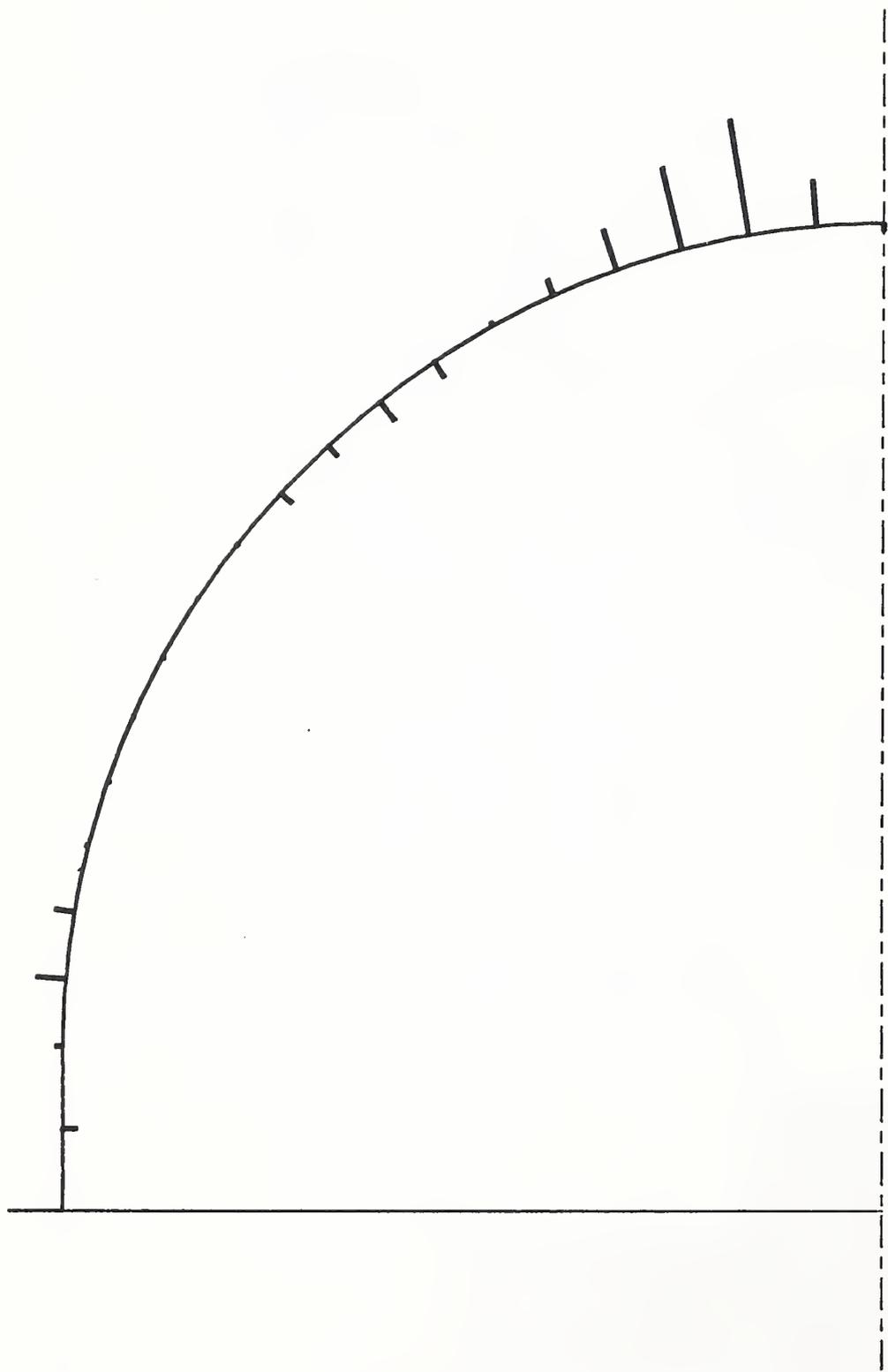
Error Bars X250
25.4um X250

Figure 5. Error Plot of Uncompensated Part

8

This program converts these residual errors into RTEC correction counts for each machine axis as a function of machine position. Remember, that for this machine, one correction count is 4 μm in axis position. Therefore, since the largest error bar in figure 5 is 21.4 μm, 5 correction counts are needed to compensate for this error. During the finish cut, the PC continuously reads the axes' positions from the RTEC and sends the correction count commands to the RTEC to be inserted in real time.

After the finish cut, the part is again measured with fast probing. The errors are determined and then displayed by the CAD package. Figure 6 shows the residual errors on the nose displayed by the CAD package. If the compensation system works as well as possible, the components of the residual errors in each machine axis are less than the 4 μm correction resolution. The longest error bar in figure 6 is 4 μm. The SD of the circle fit is 2.0 μm.

Alternatively, compensation using modification of the NC part program would require replacing the single "G02" instruction line with ninety "G01" linear-interpolation lines which approximate this 38-mm radius within about 1.5 μm. The coordinates of the points in the region of the bump would then be offset from nominal to compensate for the error. Using the RTEC is much simpler.

The following sections will describe in more detail the use of the CAD system, the generation of the NC probing program, and the determination of corrections to be inserted by the RTEC.

3. PROBING POINT AND ERROR DETERMINATION USING A CAD PACKAGE

PROBING POINT GENERATION USING A CAD PACKAGE

We use a commercial CAD package, CADKEY [5], which runs on the Quality Controller PC. Probing path generation with the CAD package starts with a CAD profile drawing of the finished part similar to figure 4. This profile contains all surfaces (part features) on which the probing points are to be located. Each surface is represented by an entity within the CAD package. These entities are arcs (parts of spheres), lines (cylinders, tapers, and steps), and conics. Probing points and the resulting probing paths are defined interactively using a program called PARTMK.CDP. This CADKEY CAD Language (CADL) program is used to identify the part features as separate entities and sorts them according to probing order. Once the features are sorted, the CAD operator can interactively locate probing points on each surface. These probing points can be located with a mouse or specified from actual dimensions. The CADL program then creates a file of these probing points, which consists of the "start" and "finish" points of each probing move. The probing paths are defined as the sequence of
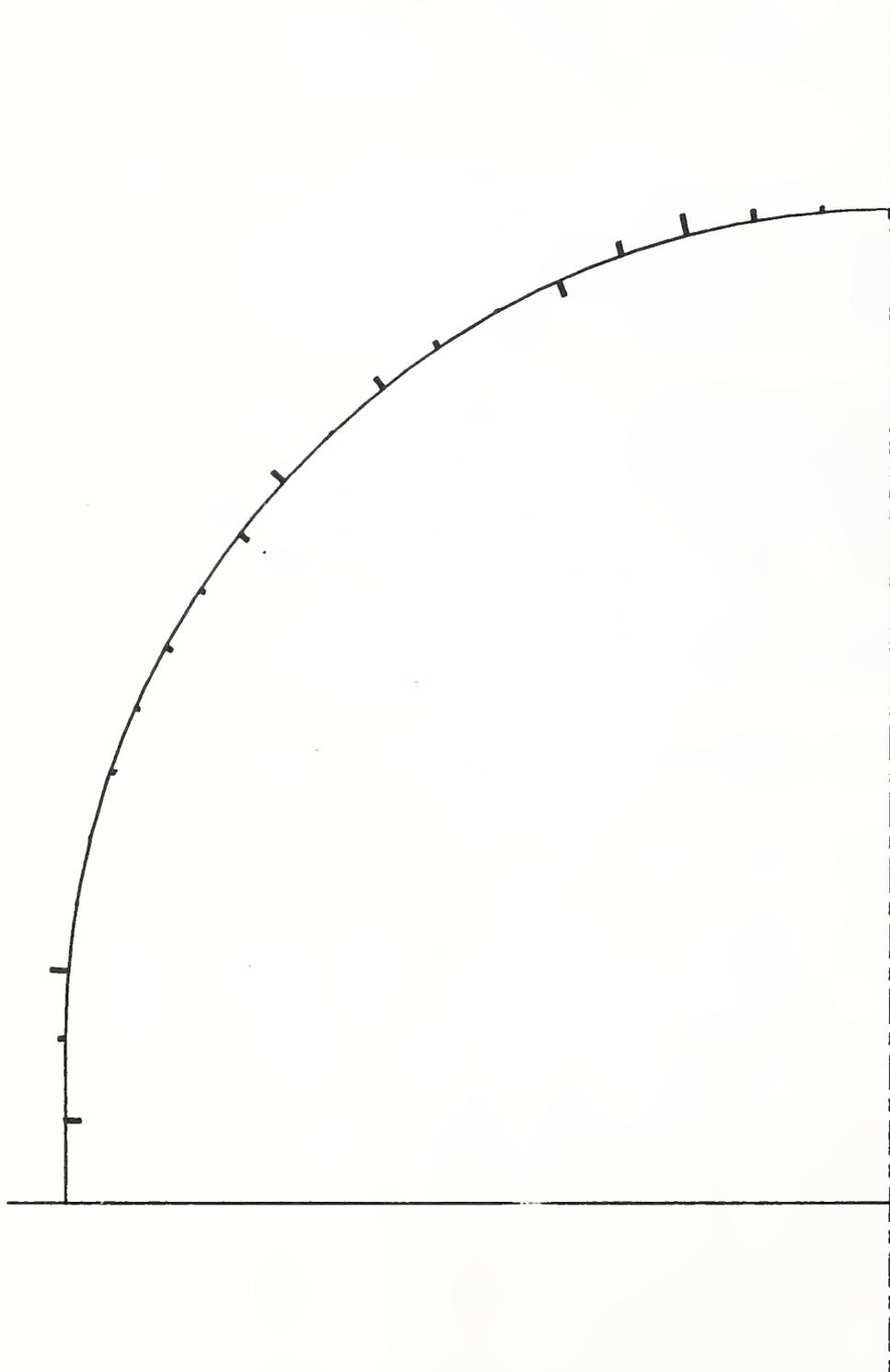
Error Bars X250

25.4um X250

Figure 6. Error Plot of Compensated Part

10

moves between the points which are passed in the file to the program to generate the actual probing program, NCMAKE. A file of the expected trip-point coordinates is passed to the ERROR correction program. These files passed differ in that the file to NCMAKE includes additional points which are interactively generated to ensure bypassing surface features which could damage the probe, i.e., avoiding crashes. Using the ERROR file and the actual probing data (machine coordinates at probe trip), part errors can then be calculated.

PART ERROR CALCULATIONS

A program, ERROR.EXE, in the C programming language, was written to calculate part errors by comparing the expected probing-point coordinates with the fast-probing data. In this report, the emphasis is placed on the compensation of the errors in the hemispherical feature of the part even though error correction could be performed on all part features. Determination of the hemispherical errors is performed by the least squares (LS) fit of a circle to the fast-probing data. Determination of part shape errors from probing data is nontrivial since the only data available are the two-dimension X and Z machine coordinates at each probe trip. One needs to know both the radius and center of the semicircle (ideally, the hemisphere) in order to accurately determine part shape.

The error calculation program implements this LS fit in a two-step process. Two input files are needed, the fast-probing data (rtecdat.out) and the expected coordinates at the probing points (probpath.cdl) generated within CADKEY from the part outline. These expected points were calculated during the probing-path generation. The complete set of fast-probing data is first corrected for effects of the probe-tip ball radius and the direction of approach. Next, the data from the hemispherical section is LS fit to a circle and the errors at each probing point determined. A standard deviation (SD) calculation is performed on the resulting best-fit circle and the data. For the data in figure 5, the SD with all the data points included is 5.6 μm. Any point with a "large" SD value will bias the LS circle fit to the data, moving both the center of the circle and surface shape away from the true values. With the removal of the largest of the calculated error data points (whose SD exceeds a 0.5 SD threshold), always keeping the two end points, a second LS circle fit is performed and a new error data set calculated for all the data. Performing a second SD calculation on the fit using this altered data correction set results in a better fit with significantly lower SD values, e.g., 1.7 μm for the data in figure 5. This second LS circle fit results in a center closer to the true location and a better estimate of the surface shape. This has been proven by using input data sets for a circle and one with known shape errors in place of the probing data as a test. Using this second error data set, two

11

files are written, error.cor for use by the Quality Controller
program and whisker.cdl used to display the surface-normal errors
on the part outline with the CAD package.

CAD ERROR DISPLAY

Determination of the part errors has now been performed in the
ERROR program and written to a file. This file contains the error
magnitude and angle information for each probed point on the
hemisphere. The CAD package is used to display these errors on the
part outline. The errors are drawn normal to the surface at each
measurement point. A magnification of length of the errors has
been included which will display the error magnitudes at a size
which can easily be observed on the CAD screen. The expected error
at any point, after RTEC corrections have been performed, should be
no greater that 4 micrometers (about 0.16 mils). By applying a 250
times magnification to the length, a 4 micrometers error is
displayed as a 1 millimeter (about 40 mils) bar relative to the
part dimensions (i.e., nose radius) displayed.

This CAD error display program requires only one input file. The
program sets up the CAD display of the part outline and displays
the error bars as "whiskers" (figure 4). The hemispherical section
of the part may be magnified on the screen along with the error
whiskers for that feature (figure 5).

4. AUTOMATIC GENERATION OF THE NC PROBING PROGRAM

The CAD procedure to develop the file of probe-path end-point
coordinates has been described above. The program called NCMAKE,
described here, reads that file, prompts the user for certain NC
program parameters, generates the NC probing program, and writes it
to a file. NCMAKE, written in the C programming language, runs on
the Quality Controller PC.

OPERATING PROCEDURE AND FEATURES

A file containing all the coordinates defining the probe-path end
points is prepared in advance using the CAD package. When NCMAKE
is run, a display of the current values for all the NC program
parameters appears on the screen, each identified by name, and
referenced with a letter of the alphabet. For example, M may refer
to a feed rate whose current value is 2.5 (m/min) [100 in/min]; and
R could refer to a note, "Touch face of part then MDI W0.75..." A
prompt informs the user that the value of a parameter may be
changed by first striking the key for the letter corresponding to
the parameter. To change this note, the user would strike the R
key and see a message explaining the parameter (in this case,

stating that it is the first line of general comments and may not exceed 66 characters). The choice is then to either leave the note unchanged or key in a replacement.

When satisfied that the values of the parameters are all appropriate, the user may proceed to have the program produce the NC program. Striking the Y key will initiate the generation of the NC program which includes comments to explain each move of the probe. This is ideal if the file is to be visually inspected. Alternatively, if <Enter> is struck, the program will have minimal comments, and will therefore comprise a significantly smaller file which will conserve disk space and memory in the MTC.

The first twenty lines of the NC program are immediately displayed. More of the program will be displayed, twenty lines at a time, each time <Enter> is struck. The program will be down-loaded to the MTC if the D key is struck. If it is not desired to down-load the program, the user may terminate the program at this point (or any point) by striking <Esc>.

The NC program is saved to a filename identified by the part number and segment number (explained in the next section). The date and time the program was produced are included in comments.

REQUIREMENTS OF NC PROBING CODE

Various needs and developments led to the effort to automate the generation of NC probing code. NC programs for a part to be processed by the QIA system must be divided into several machining and inspection segments to accommodate the need for process-intermittent dimensional inspection and subsequent compensation of the final cut. Each inspection session has many aspects which are common from part to part, allowing the NC code to have repeated characteristics. Probe paths are determined by methods which are similar for nearly every part, yet writing probing code is tedious. A CAD representation of each part is usually available, and may be used to develop lists of coordinates defining probe-path end points. It was decided to use such a list as the basis for automatically generating probing code.

QIA procedures include a process-intermittent loop for refining tool-path coordinates in an NC program. A workpiece first undergoes rough and semifinish cuts, and then is inspected with the probe. The RTEC may be used to compensate for the dimension or form errors detected or, alternatively, the tool-path coordinates for the finish cut may be adjusted. After the finish cut, the part is inspected again. The alternating machining and probing sessions require the segmented NC programs. NCMAKE produces the probing segments such that they can be executed as independent NC programs. The NC program identification label, required in the first block of NC code, is assigned as six alphanumeric characters. The first two

13

characters are the two-digit part number. The next two are the two-digit part-version number. The fifth character is the letter "s," indicating that the following character is the one-digit segment number. The first machining segment is numbered 1, and therefore probing segments are always even-numbered. The filename for an NC program segment consists of the letters "pt" followed by the NC program identification label, and with the file extension "dnc". For example, for the first probing segment (and therefore the second segment overall) for the fourth version of Part 63, the filename would be "pt6304s2.dnc".

Considering various requirements of process-intermittent inspection, it is clear that most segments of NC probing programs should have certain types of notes and comments in common. NCMAKE provides for certain "standard" kinds of comments (e.g.,tool numbers and lengths)--some of which were selected to demonstrate the program's capability for the prototype discussed in this report--which will be included in every file of NC program produced. For example, a comment states whether the touch-trigger probe has an extension. This type of note may be easily changed or deleted, and is included to demonstrate the degree of flexibility and control a user may have over the code produced by the program.

The routine nature of probe moves during inspection allows a simple design for the file of probe-move end-point coordinates required by the program. The file, named pathfin.cdl (figure 7), is made up of repeated sequences of coordinates to approach (start), touch, and retract from (finish) each successive gauging point. The probing-path definition program PARTMK.CDP, described in section 3, was developed to write the probing-path end-point coordinates to pathfin.cdl. NCMAKE uses the coordinates directly.

```
POINT start 5.148643, 1.793852, 0.000000
POINT touch 1, 5.123869, 1.494877, 0.000000
POINT finish 5.148643, 1.793852, 0.000000
POINT start 5.000000, 1.800000, 0.000000
POINT touch 2, 5.000000, 1.500000, 0.000000
POINT finish 5.000000, 1.800000, 0.000000
POINT start 4.850000, 1.800000, 0.000000
POINT touch 3, 4.850000, 1.500000, 0.000000
POINT finish 4.850000, 1.800000, 0.000000
POINT bypass 4.854984, 2.092784, 0.000000
POINT start 4.550000, 2.100000, 0.000000
POINT touch 4, 4.550000, 1.800000, 0.000000
POINT finish 4.550000, 2.100000, 0.000000
POINT start 4.243750, 2.100000, 0.000000
POINT touch 5, 4.243750, 1.800000, 0.000000
POINT finish 4.243750, 2.100000, 0.000000
```

Figure 7: Example "pathfin.cdl" File of Probe-Path Coordinates: similar to probing points where the hemisphere ends and the cylinder starts in figure 4.

An exception to this pattern is the case of a "bypass" point, a point in space to be traversed by the probe after retracting from one gauging point, before proceeding to the approach point for the next. A bypass point is sometimes necessary to provide sufficient clearance where the probe would otherwise collide with the part. Comments are included in each record in the file, indicating whether the values in that record are coordinates of the approach, touch, retract, or bypass point. The gauging point number is also given for each touch point.

NCMAKE is designed to create code which moves the machine first to the user-defined probing start location, and then moves only along the Z axis to the Z coordinate of the approach point for Gauging Point 1. Next, it moves in only the X direction to the approach point. Moving one axis at a time in this order ensures that the preliminary moves of the probe allow maximum clearance from the workpiece. The probe will touch Point 1, retract, approach Point 2, etc., according to the coordinates in pathfin.cdl, as explained above.

The program uses a file named template.prb as a template for the NC probing-program segment. The file (figure 8) is copied character-by-character, except that variable symbols are replaced by corresponding numerical values or text strings. A variable symbol consists of a special character followed immediately by an integer. There are two special characters used: $ and @. If the special character is $, then the integer is a number referring to one of the NC-program parameters discussed earlier. The only exception is $21, which is replaced by the complete set of probe-move blocks comprising the main body of the NC program segment. If the special character is @, then the text to be substituted does not originate from the parameter list, but from within the main program. This approach is necessary, for example, when the substitution will be derived from a calculation which is based on a parameter value. For example in line N0180, the $4 will be replaced by the tool number entered as a parameter, and the @4 will be replaced by a tool offset number (e.g., 00 for no offset) from the main program.

15

```
N0010   (ID,PROG,$1a1$2,$3 SEG$2: GAUGING a2)
N0020   ! a3
N0030   ! Tool $4:  Touch-trigger probe$5 in $6 boring bar holder.
N0040   !    Length: Z=$7  X=$8    Offset $20: Z=$9  X=$10
N0050   ! Touch face of part, then MDI W$11 for PROBING START
N0060   ! P$15 =$13 PROBING START: Machine Z coordinate to which
N0070   !    probe moves from the "standard" initial machine coordinates
N0080   !    [X0, Z10].
N0090   ! Probe tip diameter = $14
N0100   ! Feedrate = $12
N0110   ! $16
N0120   ! $17
N0130   ! $18
N0140   G70                          ! INCH MODE
N0150   G94 T00                      ! LINEAR FEED, CLEAR TOOL
N0160   G00 Z(P$15)                  ! RAPID TO PROBING START
N0170   G92 X30.4 Z$19               ! PRESET REF COORDS
N0180   T0$4a4                       ! INDEX TURRET TO PROBE
N0190   M14                          ! TURN ON PROBE
N0200   M15
$21
Na5   G00 X30.4 Z$19 T00             ! BACK TO REF COORDS, CLEAR TOOL
Na6   G92 X0 Z(P$15)                 ! BACK TO PROBING START
Na7   G00 Z10                        ! BACK TO INITIAL MACHINE POSITION
Na8   M02                            ! END OF SEGMENT
N9999   (END, PROG)
```

Figure 8: The Template File for the NC Program, "template.prb"

The current version of NCMAKE has been tailored to the requirements
of the prototype QIA turning center. However, the program has been
structured so that a few simple changes, mostly to data files
rather than program modules, could adapt it to significantly
different NC programming applications.

5. IMPLEMENTING CORRECTIONS: THE QUALITY CONTROLLER AND THE RTEC

A program, running on the Quality Controller, interfaces with the
RTEC through a parallel interface card in the PC. For each axis,
the position is received by the PC in three bytes. For the machine
in use, the longest axis is one meter. With a position resolution
of one micrometer, twenty binary bits (to count to a million) are
required. A Correction Requested byte is sent from the PC to the
RTEC.   Since the correction resolution is 4 μm, the range is
$\pm$127 μm. A Correction Completed byte is sent back by the RTEC so
the PC knows if the corrections have been implemented. A Status
byte and error codes in the upper bits of the axis-position words
are exchanged.

COMPENSATION OF G-T ERRORS

For compensation of G-T errors, the Quality Controller would read the machine temperatures from thermocouples located on the machine at critical points prior to a cut. During the cut, the PC continuously reads the axis positions and calculates 11 error components for this two-axis machine using previously determined mathematical models which are polynomial functions of position and temperature. The error components are combined to form a single correction-component for each axis which is converted to correction counts and sent to the RTEC as a Correction Requested. A new correction is calculated about every 50 ms. During this time the cutting tool moves less than one millimeter.

COMPENSATION OF PART ERRORS

Correction of errors determined by in-process gauging may be performed separately or in addition to the G-T error correction. This compensation uses a look-up table, rather than a model or equation to determine the corrections. After fast-probing of the semifinish cut, the RTEC interface program reads the error.cor file. For the compensation of the errors in the hemisphere, this file contains an error magnitude in micrometers versus the angle in radians from the part (+Z) axis for each point probed. The program calculates the X and Z axis components of each error by multiplying the error magnitude by the sine and cosine of the angle and dividing these by 4 μm to convert them to correction counts.

These corrections are tabulated versus the X and Z tool positions in part coordinates used to cut the arc which are readily calculated from the known location of the arc center, arc radius and angle of the probing point. Intermediate correction values between the probing locations are determined by interpolation and inserted in the table to smooth the corrections. The X and Z part positions must then be converted to machine positions using the known X and Z tool dimensions. As the tool traverses the arc, the PC sends the appropriate Correction Requested signals to the RTEC. The correction is sent as the axes approach the positions in the tool position table to ensure that the correction is implemented when the tool cuts that point. Ideally, the residual errors in the finish cut (such as figure 6) should all be less than the 4 μm correction resolution.

6. ACKNOWLEDGEMENT

7. SUMMARY

An automated method for compensation of part errors determined by in-process gauging has been demonstrated. Fast-probing using a touch-trigger probe is used for the gauging. Probing points are interactively selected on a CAD program display of the part outline. The NC program to probe this part is automatically generated and down-loaded to the MTC. Errors, deviation from nominal dimensions, in a semifinish cut are determined and displayed using the CAD program. These errors are used to determine corrections, implemented by the RTEC, which modify the tool path during the finish cut. The system has the capability of reducing errors in the finished part to less than the correction resolution which for the machine used is 4 μm.

7. REFERENCES

1. "Progress Report of the Quality in Automation Project for FY88," C. D. Lovett ed., NISTIR 89-4045, National Institute of Standards and Technology, Gaithersburg MD, April 1989.

2. "Progress Report of the Quality in Automation Project for FY89," T. V. Vorburger and B. Scace eds., NISTIR 4322, National Institute of Standards and Technology, Gaithersburg MD, May 1990.

3. "Progress Report of the Quality in Automation Project for FY90," M. A. Donmez ed., NISTIR 4536, National Institute of Standards and Technology, Gaithersburg MD, March 1991.

4. "Implementing Fast Part Probing and Error Compensation on Machine Tools," K. W. Yee and R. J. Gavin, NISTIR 4447, National Institute of Standards and Technology, Gaithersburg MD, October 1990.

5. CADKEY Version 4.0 - 386, Revision 1, CADKEY, Inc., Manchester, CT 06040.

NIST-114A (REV. 3-90)	U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY	1. PUBLICATION OR REPORT NUMBER
		NISTIR 4854
		2. PERFORMING ORGANIZATION REPORT NUMBER
	BIBLIOGRAPHIC DATA SHEET	3. PUBLICATION DATE
		JUNE 1992

4. TITLE AND SUBTITLE

Automated Compensation of Part Errors Determined by In-Process Gauging

5. AUTHOR(S)

Kenneth Yee, Herbert Bandy, Jack Boudreaux, Neil Wilkin

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)	7. CONTRACT/GRANT NUMBER
U.S. DEPARTMENT OF COMMERCE NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY GAITHERSBURG, MD 20899	
	8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

10. SUPPLEMENTARY NOTES

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

An automated method for compensation of part errors determined by in-process gauging using a real-time error corrector (RTEC) has been demonstrated. Fast-probing using a touch-trigger probe is used for the on machine gauging. Probing points are interactively selected on a computer-aided design (CAD) program display of the part outline. The numerical control (NC) program to probe this part is automatically generated and down loaded to the machine-tool controller. Errors, deviation from nominal dimensions, in a semifinish cut are determined and displayed using the CAD program. These errors are used to determine corrections, implemented by the RTEC, which modify the tool path during the finish cut. The system has the capability of reducing errors in the finished part to less than the correction resolution which for the machine used is 4 micrometers.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

CAD; controller; error compensation; error; machine tool; part; real time

13. AVAILABILITY	14. NUMBER OF PRINTED PAGES
[XX] UNLIMITED	21
[ ] FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).	
[ ] ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.	15. PRICE
[X] ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.	A02

ELECTRONIC FORM