ABSTRACT

HOOD, DAVID WAYNE. Force Feedback Control of Tool Deflection in Miniature Ball End Milling. (Under the direction of Dr. Gregory Dale Buckner.)

Previous research at North Carolina State University and other institutions worldwide focused on open-loop compensation of machining errors associated with tool deflection in miniature ball end milling. These methods utilized tool force models to predict deflections and pre-compensate tool paths off-line to achieve dimensional tolerance and accuracy in finished parts. Accuracy depended on the tool force model, its cutting parameters, and workpiece alignment and dimensional accuracy. Real-time force feedback has the potential to further improve the accuracy of profiles created during machining. This paper demonstrates that force feedback can be used to predict tool deflection and compensate for deflection during the milling operation, reducing susceptibility to uncertainties in model parameters and workpiece alignment. Two specific force feedback approaches are presented here: cutting depth prediction (based on a non-dynamic cutting force model) and tool deflection prediction (using a non-dynamic model of tool stiffness). Real-time control algorithms incorporating both methods were implemented and evaluated on a high-speed air bearing spindle. A non-dynamic tool force model developed previously at the Precision Engineering Center used measured forces to predict depth of cut. A separate tool stiffness model was developed to predict tool deflections (axial and radial) based on measured forces. Experiments involving machined grooves in hard steel workpieces, including simple slotting cuts and three-dimensional finishing operations, were conducted at various tool tilt angles to evaluate the effectiveness of force feedback control. Results indicate that
profile errors can be reduced up to 80% compared to non-compensated cases. These results confirm that real-time force feedback control can significantly improve the dimensional tolerance and accuracy of injection molds created using miniature ball end mills.
FORCE FEEDBACK CONTROL OF TOOL DEFLECTION IN MINIATURE BALL END MILLING

by

DAVID WAYNE HOOD

A thesis submitted to the Graduate Faculty of North Carolina State University in partial fulfillment of the requirements for the Degree of Master of Science

MECHANICAL ENGINEERING

Raleigh

2003

APPROVED BY:

__________________________________________________________

Chair of Advisory Committee
This work is dedicated to the glory and honor of

my personal Lord and Savior, Jesus Christ.

For He alone is worthy of praise.
BIOGRAPHY

David Wayne Hood began his post-secondary education at North Carolina State University where he earned a Bachelor of Science in Mechanical Engineering graduating valedictorian of the class of 2001. In August 2001, he started graduate school at North Carolina State University majoring in mechanical engineering working at the Precision Engineering Center under the advisement of Dr. Gregory D. Buckner. His research involves an investigation of force feedback control of tool deflection in miniature ball end milling. He graduated in August 2003 and began work at Evernham Motorsports in research and development.
ACKNOWLEDGMENTS

There are many people I would like to thank who have been instrumental in my completion of this degree. The patience and support of all of those around me have been a tremendous help and is greatly appreciated. I would first like to thank Dr. Gregory D. Buckner, my advisor, for taking me as a student and being an excellent mentor: thanks also for being a good friend. I would like to thank Dr. Thomas A. Dow for his support and insight at the Precision Engineering Center. His help in developing the necessary skills to conquer real world problems and thoroughness is the foundation of the Precision Engineering Center and is greatly appreciated. Thank you to Alex for help with machining and for offering support in the weekly discussions about which is better – model or stiffness based closed-loop control. Thanks to Ken for all the PMAC help as well as Maple and MATLAB programming – he never minded taking time to help with the project. Thank you to Dr. Ron O. Scattergood for being on my committee and for his insight. Thank you as well to Laura for all her administrative help and purchase orders. Also to all the students at the PEC I want to thank you for the great times we had, the great discussions, and friendships developed over the past year. I especially want to thank Stuart Clayton, David Gill, David Kametz, Patrick Morrissey, Nobu Negishi, Witoon Panusittikorn, and Travis Randall.

I want to thank my mom, dad, and brother who have always been there for me with love and support, telling me I can do anything, and that I should never compare myself to anyone else, but to just be the best I can be. Thank you especially to my dad for allowing me to work with him the past 10 years – his attitude of perfection, work ethic, and hand skills I’ve both inherited and learned from watching him, and he has given me an education not available in school. You have no idea how much you all mean to me and I will always remember your words of wisdom.

And thanks to my girlfriend Kelly Lambert. Your constant encouragement, support, patience, and love throughout this time were invaluable– words cannot explain how much
you mean to me and I want you to know I love you. You’re an inspiration and I sincerely appreciate all your prayers, support, and for always being there for me.

Most of all, thank you to my personal Lord and Savior Jesus Christ. All that I have and all I’ve been given are gifts from Him and He gave me this opportunity and many blessings. Thank you for the patience, love, understanding, peace, joy, kindness, goodness, faithfulness, gentleness, and self-control needed to make it through. Education is wonderful but if you don’t know the Lord you are missing the most important thing of all.
TABLE OF CONTENTS

List of Tables ......................................................................................................................viii
List of Figures .......................................................................................................................ix
1 Introduction ....................................................................................................................1

2 Modeling Tool Forces and Deflections .........................................................................4
   2.1 Non-Dynamic Tool Force Model ..............................................................................4
   2.2 Measured Tool Forces ...............................................................................................7
   2.3 Tool Stiffness and Deflection ....................................................................................8

3 Closed-Loop Compensation of Tool Deflection..........................................................12
   3.1 Cutting Depth Prediction .........................................................................................13
     3.1.1 Advantages of Cutting Depth Prediction ..........................................................14
     3.1.2 Disadvantages to Cutting Depth Prediction ......................................................14
   3.2 Tool Deflection Prediction ......................................................................................15
     3.2.1 Advantages to Deflection Prediction ...............................................................16
     3.2.2 Disadvantages to Deflection Prediction ...........................................................16

4 Experimental Implementation .....................................................................................17
   4.1 Axis Control on the Nanoform 600 .........................................................................17
     4.1.1 Axes Controller Tuning ...................................................................................18
   4.2 Deflection Compensation on the Nanoform 600 .....................................................20
     4.2.1 PMAC Limitations ...........................................................................................22
   4.3 External Data Acquisition and Signal Processing ...................................................23
     4.3.1 First-order Digital Filter ..................................................................................28

5 Experimental Evaluations ............................................................................................35
   5.1 Part Measurement on Talysurf Profilometer ...........................................................36
   5.2 Linear Slotting Cut Experiments ............................................................................37
     5.2.1 Experimental Multiplication Factor ..................................................................38
     5.2.2 Predicted Depth Compensation ......................................................................40
     5.2.3 Predicted Deflection Compensation ................................................................43
   5.3 Modified Sinusoidal Slotting Cut Experiments .......................................................46
     5.3.1 One-Period Modified Sinusoidal Profile .........................................................46
     5.3.2 Two-Period Modified Sinusoidal Profile .........................................................50
   5.4 Repeatability Experiments .......................................................................................53
     5.4.1 Linear Slotting Cut Experiments .....................................................................53
     5.4.2 One-Period Modified Sinusoidal Profile .........................................................55
     5.4.3 Two-Period Modified Sinusoidal Profile .........................................................56
   5.5 Investigating profile Errors in Small Groove Experiments .....................................58
     5.5.1 Cross-Correlation Coefficients .......................................................................58
     5.5.2 Uncompensated Linear Slotting Cut: Cross-correlations .................................59
     5.5.3 Compensated Linear Slotting Cut: Cross-correlations ....................................63
     5.5.4 Improved Maximum Force Capture Algorithm .................................................66
     5.5.5 Uncompensated Linear Slotting Cut: Cross-correlations using Improved
         Maximum Force Capture Algorithm .................................................................68
**LIST OF TABLES**

Table 1  Measured and computed stiffness values for long shank, ball end tools

Table 2  Ziegler-Nichols ultimate stability tuning criteria [13] and gains for axes

Table 3  System frequencies

Table 4  Cross-correlation coefficients for uncompensated linear slotting cut

Table 5  Cross-correlation coefficients for compensated linear slotting cut

Table 6  Cross-correlation coefficients for compensated linear slotting cut with new maximum force capture

Table 7  Large groove experimental results: deflection compensation vs. uncompensated

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Measured and computed stiffness values for long shank, ball end tools</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Ziegler-Nichols ultimate stability tuning criteria [13] and gains for axes</td>
<td>19</td>
</tr>
<tr>
<td>3</td>
<td>System frequencies</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>Cross-correlation coefficients for uncompensated linear slotting cut</td>
<td>63</td>
</tr>
<tr>
<td>5</td>
<td>Cross-correlation coefficients for compensated linear slotting cut</td>
<td>65</td>
</tr>
<tr>
<td>6</td>
<td>Cross-correlation coefficients for compensated linear slotting cut with new maximum force capture</td>
<td>73</td>
</tr>
<tr>
<td>7</td>
<td>Large groove experimental results: deflection compensation vs. uncompensated</td>
<td>86</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

Figure 1: Long shank, ball end milling tool: geometry and force components .................. 5
Figure 2: Depth vs. force curve derived from the non-dynamic cutting force model .......... 6
Figure 3: Nanoform DTM machine and setup ................................................................. 7
Figure 4: Workpiece mounted on three-axis piezoelectric load cell ............................... 8
Figure 5: Measured and predicted orthogonal cutting forces: 25 degree tool tilt .............. 8
Figure 6: Experimental force vs. deflection curve at 25 degree tilt (added linear trendline) ......................................................................................................................... 11
Figure 7: Tool deflection with desired and actual depth .................................................. 12
Figure 8: Diagram of predicted depth algorithm ............................................................. 13
Figure 9: Diagram of predicted deflection algorithm ..................................................... 15
Figure 10: Ziegler-Nichols ultimate stability tuning criteria [13] ................................. 19
Figure 11: PID axis controllers: tuned step responses, 0.1 mm ...................................... 20
Figure 12: z-force at 25 degree tilt cutting .................................................................... 24
Figure 13: Diagram of dSPACE peak and hold algorithm ............................................. 26
Figure 14: Bode plot of Butterworth 5th order filter characteristics ............................... 27
Figure 15: dSPACE cutting force and maximum force capture ..................................... 28
Figure 16: z-plane for a discrete system ........................................................................ 30
Figure 17: Unit step input responses of a first-order digital filter with differing coefficients ................................................................................................................. 32
Figure 18: Time constant of first-order digital filter ....................................................... 33
Figure 19: Small groove experimental setup ............................................................... 35
Figure 20: Small groove Talysurf alignment procedure ............................................... 37
Figure 21: PMAC acquired cutting force vs. actual cutting force .................................. 38
Figure 22: PMAC acquired cutting force and filtered force .......................................... 39
Figure 23: Predicted deflection during uncompensated cut without multiplication factor 40
Figure 24: Experimental profile measurements: predicted depth compensation, linear slotted cut ................................................................................................................ 41
Figure 25: Experimental profile errors: predicted depth compensation, linear slotted cut 42
Figure 26: Experimental profile measurements: predicted deflection compensation, linear slotted cut ........................................................................................................ 44
Figure 27: Experimental force measurements: predicted deflection compensation, linear slotted cut ........................................................................................................ 45
Figure 28: Experimental profile errors: predicted deflection compensation, linear slotted cut ................................................................................................................. 46
Figure 29: Experimental profile measurements: predicted deflection compensation, one period sinusoidal slotted cut ............................................................... 47
Figure 30: Experimental force measurements: predicted deflection compensation, one period sinusoidal slotted cut ............................................................... 48
Figure 31: Experimental profile errors: predicted deflection compensation, one period sinusoidal slotting cut ............................................................... 49
Figure 32: Experimental profile measurements: predicted deflection compensation, two period sinusoidal slotting cut ......................................................... 50
Figure 33: Experimental force measurements: predicted deflection compensation, two period sinusoidal slotting cut ......................................................... 51
Figure 34: Experimental profile errors: predicted deflection compensation, two period sinusoidal slotting cut ............................................................... 52
Figure 35: Experimental profile repeatability errors: predicted deflection compensation, linear slotting cut ............................................................... 54
Figure 36: Experimental profile repeatability errors: predicted deflection compensation, one period sinusoidal slotting cut ......................................................... 56
Figure 37: Experimental profile repeatability errors: predicted deflection compensation, two period sinusoidal slotting cut ......................................................... 57
Figure 38: Experimental force measurements: filtered and captured maximum cutting force, uncompensated linear slotting cut ............................................................... 60
Figure 39: Experimental profile measurement: uncompensated linear slotting cut .......... 61
Figure 40: Experimental workpiece surface profile measurement before cutting .......... 62
Figure 41: Experimental force measurements: Butterworth and maximum force capture, compensated linear slotting cut ............................................................... 64
Figure 42: Experimental profile measurement: compensated linear slotting cut .......... 65
Figure 43: Old maximum force capture algorithm ......................................................... 67
Figure 44: Improved maximum force capture algorithm ................................................ 68
Figure 45: Experimental force measurements: filtered and improved captured maximum cutting force, uncompensated linear slotting cut ............................................................... 69
Figure 46: Experimental force measurements: filtered and improved captured maximum cutting force, compensated linear slotting cut ............................................................... 71
Figure 47: Experimental profile measurements: compensated linear slotting cut profile, old and improved maximum force capture ............................................................... 72
Figure 48: Repeatability measurements made on Talysurf ................................................ 74
Figure 49: Removing workpiece tilt from Talysurf profile measurement data ............................................................... 76
Figure 50: Transient response of z-axis to a linear groove profile (non-cutting) .......... 77
Figure 51: Transient response of z-axis to a modified two-period sinusoidal profile (non-cutting) ............................................................... 78
Figure 52: Experimental profile measurement: compensated linear slotting cut profile with improved maximum force capture without first-order filter ............................................................... 79
Figure 53: Large groove experimental setup ............................................................... 81
Figure 54: Cross sectional sketch of the tool path and thrust force for large groove experiment (where $F_t$ = thrust force and $\gamma$ = sweep angle) ............................................................... 82
Figure 55: Rough pass large groove error from 1.5 mm radius ............................................................... 83
Figure 56: 3.0 mm (top) and 0.8 mm (bottom) ball end milling tools ............................................................... 84
Figure 57: Experimental profile measurement: uncompensated large groove error from 1.6 mm radius ............................................................... 85
Figure 58: Experimental profile errors: predicted deflection compensation, large groove experiment ............................................................... 86
Figure 59: Experimental profile repeatability errors: predicted deflection compensation, large groove experiment .............................................................................................. 87
1 INTRODUCTION

Injection molding is an important manufacturing process for optical and mechanical components. The hard steel dies used in this process play a direct role in the quality of molded parts. Traditionally, fabricating these dies involved rough milling followed by heat treatment, grinding, and polishing to the desired shape. Recently, high-speed machining of heat-treated steel (hardness > 55 R_c) has become a viable approach for reducing fabrication times while retaining the necessary shape control. However, as feature sizes drop below 1 mm with dimensional tolerances on the order of 10 µm, tool deflection can create significant errors in the shape of mold surfaces. Deflections associated with miniature ball end tools can exceed 30 µm, rendering finished dies out of tolerance.

Dimensional errors during milling result from three basic sources: errors in the alignment and geometry of the workpiece, limited machine stiffness, and deflection of the cutting tool [1]. Alignment and geometric surface errors result from dimensional uncertainties associated with the workpiece being cut. Machine compliance causes position errors along machine axes as cutting forces change with time. The last source of machining error is the one most heavily researched at North Carolina State University (NCSU). It involves deflection of the tool tip during milling based on forces imparted to the tool.

To date, most of the research associated with milling tool deflection compensation has involved open-loop techniques that use non-dynamic models of the cutting process to modify the desired tool path prior to cutting [3,4,5,6,7]. Research conducted at NCSU’s Precision Engineering Center (PEC) resulted in open-loop correction techniques for tool deflection of miniature ball end mills [2]. This effort used a non-dynamic tool force model developed for diamond turning and modified it for ball end milling. These modeled cutting forces were used to predict tool deflections for specified machining conditions, and to modify the tool path before cutting. The specified tool path, together
with a CAD model of the workpiece surface, were used to determine the depth of cut, feed rate, and normal cutting force vector. This information was then used to predict the magnitude and direction of cutting forces and tool deflections, which were used to modify the tool path off-line (prior to cutting). Form errors were reduced from 50 µm to less than 10 µm with accurate knowledge of the cutting conditions and parameters.

Research conducted at other institutions has resulted in similar open-loop strategies for larger milling tools and spindles [3,4,5,6,7]. These techniques also rely on non-dynamic models of the cutting process and CAD models of the workpiece to predict tool forces and tool deflections, resulting in off-line tool path modifications. Error reductions have been reported to exceed 80% when compared with non-compensated cases (typically non-compensated errors are on the order of 100 µm). Other approaches use predicted deflections to modify cutting conditions (primarily federates) at various locations on the workpiece, with similar results reported [8,9]. Experimental results typically show significant reductions in form errors using these open-loop approaches; however form errors smaller than 20 µm have proven difficult to obtain.

Despite the promising results obtained from off-line tool deflection prediction and open-loop compensation, these methods rely on accurate models of the cutting process and cannot adapt to changes in model parameters, disturbances in the cutting process, or uncertainties associated with workpiece alignment. Thus, open-loop compensation results are only as accurate as the assumed model parameters and workpiece characteristics. For example, a worn tool may have wearland and radius dimensions different than expected or the workpiece may be inaccurately positioned, resulting in cutting depths that are larger or smaller than expected. Critical model parameters such as tool wearland, workpiece material properties, and instantaneous spindle speed are difficult to estimate, and may change dramatically during milling.

Closed-loop investigations have been conducted in the past with applications and goals differing from those presented here. Previous force feedback approaches focused on
maintaining or limiting cutting forces to keep dimensional errors below a specified threshold [10]. Other research has incorporated force feedback during drilling operations to modify the drilling rate to avoid fractures in composite workpieces [11]. These closed-loop methods do not address tool deflection compensation specifically, but seek to maintain cutting forces below a threshold to limit machining errors or to reduce tool and workpiece damage.

This paper demonstrates that force feedback can be used to predict tool deflection and compensate for deflection during the milling operation, reducing susceptibility to uncertainties in model parameters and workpiece alignment. Two specific force feedback approaches are presented here: cutting depth prediction (based on a non-dynamic cutting force model) and tool deflection prediction (based on a non-dynamic model of tool stiffness). Real-time control algorithms incorporating both methods were implemented and evaluated on a high-speed air bearing spindle. A non-dynamic tool force model developed previously at the PEC used measured forces to predict depth of cut. A separate tool stiffness model was developed to predict tool deflections (axial and radial) based on measured forces. Experiments involving machined grooves in hard steel workpieces, including simple slotting cuts and three-dimensional finishing operations, were conducted at various tool tilt angles to evaluate the effectiveness of force feedback control. Results indicate that profile errors can be reduced up to 80% compared to non-compensated cases. These results confirm that real-time force feedback control can significantly improve the dimensional tolerance and accuracy of injection molds created using miniature ball end mills.
2 MODELING TOOL FORCES AND DEFLECTIONS

2.1 NON-DYNAMIC TOOL FORCE MODEL

The tool force model developed by Clayton [12] can be used to calculate cutting and thrust forces during milling operations. Inputs to this model include material properties (Young’s modulus, volumetric work, etc.) and friction at the rake and flank faces of the tool. Tool geometry (ball end radius, wearland, and number of flutes) and cutting conditions (spindle speed, up feed, cross feed, depth of cut, and tool tilt angle) are used to find the cross-sectional area of the chip and the area of contact between the flank face of the tool and the workpiece. The model can be expressed:

\[
F_c = \frac{35WA_c}{3} \left( \frac{\cot(\gamma)}{\sqrt{3}} + 1 \right) + \mu_f A_f \left( 21.7W \sqrt{\frac{1505W}{E}} \right)
\]

(1)

\[
F_t = \frac{35\mu WA_c}{3} \left( \frac{\cot(\gamma)}{\sqrt{3}} + 1 \right) + A_f \left( 21.7W \sqrt{\frac{1505W}{E}} \right)
\]

(2)

where:

- \(F_c\) = cutting force
- \(F_t\) = thrust force
- \(A_c\) = cross-sectional area of the chip (function of depth, \(d\))
- \(A_f\) = area of the tool flank face (function of depth, \(d\))
- \(\mu_f\) = friction coefficient on the flank face
- \(\mu\) = friction coefficient on the rake face
- \(W\) = volumetric work
- \(\phi\) = shear angle in the workpiece
- \(E\) = Young’s modulus of the workpiece

These forces rotate with the flank face of the milling tool, but can be readily converted to orthogonal forces (in the x, y, and z machine directions) for comparison to experimental...
measurements using the three-axis load cell. For the two-flute, miniature ball end mills commonly used to fabricate injection mold dies (Figure 1), cutting forces and cutting depths can be readily calculated for given cutting conditions.

Figure 1: Long shank, ball end milling tool: geometry and force components

where:

\[ L = \text{length of tool shank} \]
\[ D = \text{diameter of ball end mill} \]
\[ \phi = \text{tool tilt angle} \]
\[ F_n = \text{force acting normal to workpiece surface} \]
\[ F_{na} = \text{orthogonal force component in axial direction of tool} \]
\[ F_{nr} = \text{orthogonal force component in radial direction of tool} \]
A typical plot of predicted cutting depth \( (d) \) vs. normal tool force \( (F_n) \) is presented in Figure 2 (spindle speed = 10,000 rpm, feed rate = 100 mm/min, material hardness = 55 \( R_c \), tool tilt angle = 25 degrees, tool wearland = 3 \( \mu \)m). Note that normal force is defined to be perpendicular to the workpiece surface. A quadratic trendline has been added to approximate this curve.

\[
d = 0.7783F_n^2 + 1.2398F_n
\]

**Figure 2:** Depth vs. force curve derived from the non-dynamic cutting force model
2.2 MEASURED TOOL FORCES

Preliminary cutting experiments were conducted on a Nanoform 600 Diamond Turning Machine (DTM) (Figure 3) to validate the tool force Equations (1) and (2). Cutting forces were measured using a Kistler three-axis piezoelectric load cell, shown in Figure 4. This load cell was mounted below the workpiece on the x-axis of the diamond turning machine, while the high-speed spindle was mounted on the y-axis slideway (Figure 3).

Figure 3: Nanoform DTM machine and setup

Experimental results for an S-7 steel workpiece machined at a spindle speed of 10,000 rpm, a feed rate of 100 mm/min, a tool tilt of 25 degrees with respect to the z-axis, and a cutting depth of 100 µm, are presented in Figure 5. This plot compares modeled cutting forces with measured cutting forces for a single revolution of the tool. The z-component of force in Figure 5 is dominated by thrust force (2). The x and y force components are
influenced more by cutting force (1), which rotates in the plane of the workpiece and changes from an x-direction force to a y-direction force every quarter rotation of the tool.

![Figure 4: Workpiece mounted on three-axis piezoelectric load cell](image)

![Figure 5: Measured and predicted orthogonal cutting forces: 25 degree tool tilt](image)

2.3 TOOL STIFFNESS AND DEFLECTION

The long shank, ball end milling tools used in this research have a 4 mm shank with a 0.8 mm ball diameter end (Figure 1). When used to fabricate free-form surfaces, the tool can be loaded in the axial direction, the radial direction, or both. The tool stiffness is significantly lower in the radial direction (see Table 1), therefore regions of a machined surface where the tool is loaded primarily in the radial direction will be subject to large tool deflections and form errors.
Table 1: Measured and computed stiffness values for long shank, ball end tools

<table>
<thead>
<tr>
<th></th>
<th>Value (N/m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axial Stiffness</td>
<td>1421000</td>
</tr>
<tr>
<td>Radial Stiffness</td>
<td>98930</td>
</tr>
<tr>
<td>25 degree Tilt Stiffness - Calculated</td>
<td>419565</td>
</tr>
<tr>
<td>25 degree Tilt Stiffness - Measured</td>
<td>455120</td>
</tr>
</tbody>
</table>

Tool stiffness measurements for the axial and radial direction were taken by loading the tool (mounted in the air bearing spindle) with a static weight. A cord was secured to the tool tip with adhesive, and a weight was suspended from this cord. The weight was suspended perpendicular to the tool axis to measure radial deflections, and parallel to the tool axis (away from the collet) for radial deflections. Displacement measurements at the tool tip were made using a federal gage device [12].

Air pressure in the spindle had an effect on the apparent stiffness of the tool, as the measurements accounted for the combined stiffness of the milling tool, tool holder, and air bearing spindle. For the experimentally determined stiffness of the air bearing and tool system in the axial and radial directions, the air pressure was set to 60 psi.

For most of the experiments presented in this paper, the tool was tilted at a 25 degree angle with respect to the z-axis to emphasize the effects of tool deflection. Figure 1 shows the resulting axial and radial force components for this tool configuration. For arbitrary cutting conditions, the tool deflection in the direction orthogonal to the workpiece (normal tool deflection) can be determined by:

\[
\delta_n = F_n \left( \frac{\cos^2 \phi}{k_a} + \frac{\sin^2 \phi}{k_r} \right)
\]

(3)

where:
- \(d_n\) = normal tool deflection
- \(F_n\) = normal tool force
- \(k_a\) = axial tool stiffness
- \(k_r\) = radial tool stiffness
- \(\phi\) = tool tilt angle
The normal tool stiffness \( (k_n) \) is derived from equation (3):

\[
\frac{F}{n} = k_n = \frac{1}{\frac{\cos^2 \phi}{k_a} + \frac{\sin^2 \phi}{k_r}}
\]

To experimentally validate this tool stiffness expression at a 25 degree tilt, experiments were conducted on the DTM that involved loading a non-rotating tool against a workpiece on the three-axis load cell. This tilt angle (25 degrees to the workpiece normal) was the only one investigated since most of the experiments were conducted at this tilt value. The z-axis was “zeroed” when the tool tip made initial contact with the workpiece surface. The tool’s rotational orientation with respect to the workpiece surface was adjusted so that neither flute cutting edge made direct contact, thus minimizing the contact area between the tool and workpiece. Instead, the backside of a flute made contact with the workpiece to produce as close to a spherical indentation as possible during the experiment.

A customized motion program then advanced the tool 40 \( \mu \)m in the z-direction at a feedrate of 0.25 mm/min while measuring the normal force imparted on the load cell. Spindle air bearing pressure was maintained at 60 psi, as in the axial and radial stiffness measurements described above. Since the tool was not rotating, cutting did not occur, resulting in large deflections at the tool tip. A subsequent measurement of the workpiece indentation depth (caused by the tool) was then subtracted from the 40 \( \mu \)m displacement. This indentation depth was measured using a Zygo white light interferometer, and determined to be approximately 4 \( \mu \)m.

The resulting elastic deformation of the workpiece during loading was calculated using Hertzian stress contact formulas. The elastic deformation from a ball end mill was
calculated to be approximately 2.25 µm. This value of workpiece deformation was also subtracted from the 40 µm encoder displacement to determine the actual tool tip deflection.

These measurements and calculations resulted in experimental force vs. deflection curves for two long shank tools (Tool 1 and Tool 2), as shown in Figure 6. This figure reveals an average orthogonal tool stiffness of 485800 N/m, which compared to the calculated value of 419565 N/m (derived from axial and radial stiffness measurements of Table 1), is a 13.6% difference.

\[
F_n = 0.4858 \delta_n
\]

**Figure 6**: Experimental force vs. deflection curve at 25 degree tilt (added linear trendline)
3 CLOSED-LOOP COMPENSATION OF TOOL DEFLECTION

Real-time force feedback can be used to predict and compensate for tool deflection during milling operations, reducing susceptibility to uncertainties in the model parameters and workpiece alignment. Two specific force feedback approaches are presented here: cutting depth prediction (based on a non-dynamic cutting force model) and tool deflection prediction (using a non-dynamic model of tool stiffness). Figure 7 illustrates the effect that tool deflection has on a groove profile. The tool tip is programmed to follow a desired path. However, due to deflection the tool tip, the tool actually creates a depth of cut less than desired (labeled “actual depth” in the figure). The shaded area in this figure represents material not removed due to tool deflection.

Figure 7: Tool deflection with desired and actual depth
3.1 CUTTING DEPTH PREDICTION

Figure 8 shows a block diagram for the cutting depth prediction control algorithm. The concept behind this algorithm is straightforward: start with a desired tool path, measure real-time cutting force, use cutting conditions and a non-dynamic force model to predict the instantaneous depth of cut, and then calculate an error equal to desired depth minus predicted depth. Once this error is known a motion program either holds z-axis position (depth of cut), advances z-axis position, or reduces z-axis position. The error calculated by the PID control algorithm is:

\[ \text{error} = \text{desired depth} - \text{predicted depth} \]

\[ \varepsilon = x_d - \hat{x}(F, \psi) \]  

(5)

**Figure 8**: Diagram of predicted depth algorithm
Using the validated cutting force model, cutting depth can accurately be predicted based on cutting conditions (Figure 2). This predicted cutting depth $\hat{d}(F, \psi)$ is a function of the cutting force model that includes 14 cutting parameters $\psi$. In this way the system can correct for errors associated with tool deflection and misalignment of the workpiece.

3.1.1 ADVANTAGES OF CUTTING DEPTH PREDICTION

- Precise alignment of the workpiece with respect to the axes of the machine is not necessary as this method uses only force feedback in the control algorithm without reference to machine axes
- The created profile is referenced to the workpiece surface; therefore knowledge of the workpiece surface before cutting is not necessary
- Cutting parameters required by the cutting force model (spindle speed, feed rate, material properties, etc.) are not difficult to determine

3.1.2 DISADVANTAGES TO CUTTING DEPTH PREDICTION

- Wearland of the tool is difficult to measure and changes during machining
- Because it relies solely on force feedback to predict cutting depth, stability is a major issue for this control algorithm. During tool breaks, interruption of cuts (reaching workpiece edges, etc.) the instantaneous force goes to zero and thus the predicted depth of cut is zero. If the desired depth of cut is not zero, a large error exists in the control algorithm, resulting in large corrective federates and possible tool breakage and damage to the machine.
- Implementation and changes from encoder feedback to strictly force feedback is difficult because of the disadvantages listed above. To implement force feedback, both desired depth of cut and predicted depth of cut need to be approximately the same value, otherwise large errors can appear in the control algorithm.
- The algorithm is highly sensitive to drift and electrical noise in the load cell
3.2 TOOL DEFLECTION PREDICTION

A block diagram of the control algorithm predicting deflection using a model of tool stiffness is shown in Figure 9. The concept behind this algorithm is straightforward: start with a desired tool path, measure real-time cutting force, use the tool stiffness model and measured force to predict tool deflection, calculate an error equal to desired position minus DTM encoder position plus predicted deflection. Once this error has been calculated, PID control either holds position, moves further into the workpiece in the z-direction increasing the depth of cut, or moves out from the workpiece decreasing the depth of cut. With a validated stiffness model, deflection can accurately be predicted based on tool stiffness and a measured real time cutting force. In this way the system uses real-time force feedback to correct for errors associated with tool deflection and uses encoder feedback to maintain tool path stability. Error calculated by the PID control algorithm is:

\[
\text{error} = \text{desired tool position} - \text{encoder axis position} + \text{predicted tool deflection}
\]

\[
\varepsilon = \frac{d}{n} - x + \delta_n \quad \left(\delta_n = \text{predicted normal tool deflection}\right)
\]

(6)

**Figure 9:** Diagram of predicted deflection algorithm
3.2.1 ADVANTAGES TO DEFLECTION PREDICTION

- Wearland, spindle speed, feed rate, material properties, and depth are no longer required inputs to the cutting force model
- Deflection is dependent only on tool stiffness (which is a function tilt angle)
- Tool stiffness is easily calculated from measurements of axial and radial tool stiffness
- Encoder feedback ensures stable and reliable execution because it is a continuous, additional feedback mechanism. If cutting force goes to zero, the predicted deflection equals zero and the machine operates as if there was no force feedback in the control algorithm
- Load cell noise and drift are still critical to machining stability and accuracy, however, they affect predicted deflection only (which is typically of a smaller magnitude than DTM encoder position)

3.2.2 DISADVANTAGES TO DEFLECTION PREDICTION

- Workpiece alignment with respect to the machine axes plays a direct result on the completed cut profile
4 EXPERIMENTAL IMPLEMENTATION

This section describes the implementation and tuning of real-time, force feedback
deflection compensation algorithms on a DTM. Both methods of force feedback
presented previously, depth prediction and deflection prediction, are experimentally
evaluated and compared.

All experiments were conducted using a Nanoform 600 DTM with 3 orthogonal linear
axes and a high-speed milling spindle (Figure 3). The spindle is a Westwind air bearing,
turbine unit with a maximum speed of 60,000 rpm. The cutting tools are two-flute, long
shank, ball end milling cutters with a diameter of 0.8 mm and a length of 4 mm. A
Kistler three-axis piezoelectric load cell supports the workpiece, and is used to measure
the tool forces in real-time. A Delta Tau Programmable Multi-Axis Controller (PMAC)
system collects data from the load cell in real time, controls the DTM milling machine,
computes the corrected slide command, and incorporates constant feedback for all three
axes.

4.1 AXIS CONTROL ON THE NANOFORM 600

The DTM used for these experiments was not designed for high feedrates or
accelerations. Originally, implementation of force feedback algorithms involved taking a
desired profile and breaking it into a large number of small incremental steps. Thus a cut
might be broken down into 1000 increments each spanning tens of micrometers
depending on the desired resolution and groove profile. Each increment was
programmed as a separate line in a PMAC motion program. Cutting force was acquired
at each line execution in the motion program, corresponding to an increment along the
desired groove profile. Thus the motion program was required to make small incremental
advances along the groove profile based on deflections calculated from instantaneous
force measurements.
Experiments were conducted to determine if the DTM could make these small incremental steps while implementing force feedback. It was determined that this method of implementation was not feasible. In a PMAC motion program, the controller “looks ahead” two lines of code to assure its ability to stop at the program’s end. Axis acceleration limits are imposed to limit overshoots and improve tracking. By breaking a groove into small increments, the force feedback algorithms require that small distances be traveled at relatively high federates, resulting in large accelerations that are prohibited by PMAC limits. For the high feedrates and positional resolution required, this implementation approach was completely inadequate, and other approaches were pursued.

Due to PMAC limitations, a custom motion program was written to enable rapid accelerations of the x, y, and z-axes. The PMAC was setup such that all three axes were in a “dwell state”, allowing the motion program to completely determine the voltage commands for each axis servo. Each of the three axes was controlled using custom digital proportional+integral+derivative (PID) algorithms, as detailed below.

4.1.1 Axes Controller Tuning

PID gains in the custom motion program were tuned to provide stable, accurate tracking of each axis with the fastest possible response times (highest possible accelerations). The gains were chosen to achieve satisfactory tracking performance to 100 μm step inputs. Experimental controller tuning began by evaluating the stability limits for each axis based on the Ziegler-Nichols ultimate stability technique [13]. This technique provides baseline controller gains, but typically requires additional tuning. Using only proportional feedback control, the proportional gain ($K_p$) was increased until continuous oscillations occurred and the system became marginally unstable. This gain was designated the ultimate gain, $K_u$, and the associated period of oscillation was designated the ultimate period, $P_u$ (Figure 10). The Ziegler-Nichols tuning criteria (Table 2) used these constants.
to determine baseline PID gains for each DTM axis [13]. These rules provided a starting point for the PID gains, but further tuning was necessary to achieve desired stability and performance.

**Figure 10**: Ziegler-Nichols ultimate stability tuning criteria [13]

**Table 2**: Ziegler-Nichols ultimate stability tuning criteria [13] and gains for axes

<table>
<thead>
<tr>
<th>Gain</th>
<th>Ziegler-Nichols Criteria</th>
<th>x-axis</th>
<th>y-axis</th>
<th>z-axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proportional Gain</td>
<td>$0.6^*K_u$</td>
<td>39000</td>
<td>45000</td>
<td>48000</td>
</tr>
<tr>
<td>Integral Gain</td>
<td>$k_i/(0.5^*P_u)$</td>
<td>190000</td>
<td>140000</td>
<td>200000</td>
</tr>
<tr>
<td>Derivative Gain</td>
<td>$k_d*(0.125^*P_u)$</td>
<td>373</td>
<td>15</td>
<td>422</td>
</tr>
</tbody>
</table>

Each axis was tuned based on 100 µm steps, with final closed-loop step responses for each axis shown in Figure 11.
4.2 DEFLECTION COMPENSATION ON THE NANOFORM 600

Implementation of force feedback algorithms was accomplished using a custom motion program, written in machine g-code, which performed the following functions:

- read axis encoders
- calculate desired tool path
- calculate tracking error
- acquire force measurements
- filter force measurements
- calculate predicted cutting depth/tool deflection
- compensate for error in depth/deflection
• output control voltage to each axis servomotor

There was a need for this algorithm to be “user friendly” and applicable to a wide range of cutting conditions and g-code functions on industrial CNC machines. Therefore, various subroutines were created to perform machine movements with both methods of feedback compensation. These movements included:

• x-direction groove with linearly varying depth (z-direction)
  • without compensation
  • with predicted depth compensation
  • with predicted deflection compensation

• x-direction groove with sinusoidally varying depth (z-direction)
  • without compensation
  • with predicted depth compensation
  • with predicted deflection compensation

These subroutines are called from a main motion program where the user selects a particular cutting operating with the appropriate g-code call statement (“callxx x0.0 y0.0 z0.0 f0.0”, see Appendix A). Any profile can be created as long as it is parametrically defined as a function of time with respect to each appropriate axis. Helical cuts, sinusoidal cuts, parabolic profiles, and other complex cuts can be easily made as long as the movement of each axis is defined as a mathematical function of time.
4.2.1 PMAC LIMITATIONS

The PMAC digital controller includes an Accessory-28A analog-to-digital (A/D) conversion board for sampling of input signals. This board provides four channels of high-speed, high-resolution analog input capability to the PMAC controller. The ±10 V inputs are converted to 16-bit signed values. Conversion of the analog signal occurs at time intervals specified by the clock frequency on the PMAC board, which is 2273 Hz (440 µs). A spindle speed of 10,000 revolutions per minute (RPM) (167 Hz) was selected because this was the minimum operational spindle speed that produced adequate torque to make cuts. Thus, the A/D sampling frequency is 13.7 times that of the spindle frequency. Therefore, for each spindle rotation, force data from the A/D board is acquired approximately 13 times.

Despite this apparently acceptable sampling ratio, software execution in the PMAC results in significantly slower force sampling. As discussed in section 4.1, the deflection compensation algorithms are implemented in a g-code motion program on the PMAC. This motion program places the servo axes in “dwell mode”, then executes a “while loop” for the duration of the cutting experiment. The execution time of this “while loop” determines the true A/D sampling rate. Thus, force measurements are acquired and axis controls are updated only once per execution of the “while loop” (752 Hz), not at the PMAC clock frequency (2273 Hz).

The impact of relatively slow sampling rates imposed by the motion program on controller performance was investigated. Control theory states that if the sampling rate of a digital controller is 20 times faster than the bandwidth of the axis, then the digital control system will behave like a continuous system [13 pg. 609]. The bandwidth of the z-axis servo was measured using the Ziegler-Nichols ultimate stability technique discussed in section 4.1.1. This bandwidth is the frequency of closed-loop oscillation at marginal instability and was found to be 14.2 Hz (Table 3). Because the motion program
execution frequency is 52.9 times the bandwidth of the z-axis, the digital controller performance should not be affected by the relatively slow sampling rates. Similar conclusions can be drawn for the x and y axes, as shown in Table 3.

<table>
<thead>
<tr>
<th>Table 3: System frequencies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spindle frequency</td>
</tr>
<tr>
<td>Tooth pass frequency</td>
</tr>
<tr>
<td>A/D board conversion frequency</td>
</tr>
<tr>
<td>Motion program execution frequency</td>
</tr>
<tr>
<td>Force sampling frequency</td>
</tr>
<tr>
<td>x-axis bandwidth</td>
</tr>
<tr>
<td>y-axis bandwidth</td>
</tr>
<tr>
<td>z-axis bandwidth</td>
</tr>
</tbody>
</table>

4.3 EXTERNAL DATA ACQUISITION AND SIGNAL PROCESSING

Both deflection compensation algorithms require accurate, real-time cutting force measurements. Because the A/D sample rate is limited by PMAC software and hardware constraints, it is important to investigate the nature of actual cutting forces and the effects of suboptimal sampling on algorithm performance.

Ideally, the normal cutting force profile resembles a shifted, interrupted sine wave due to the rotational motion of the end mill and depth of cut. The z-force component increases as the tool rotates, but due to the tool tilt the flutes on the tool do not cut for the entire rotation. Thus, for a 25 degree tool tilt, the normal force increases to a maximum and then decreases to zero twice with every rotation of the tool (because it has two flutes). Measured force data (Figure 12) at a feedrate of 100 mm/min, a spindle speed of 10,000 rpm, and a depth of cut of 100 µm reveals that normal cutting force does in fact resemble a shifted, interrupted sine wave.
Differences in the magnitude of cutting force from flute to flute are evident in Figure 12. In the fabrication of miniature ball end mills, no two flutes on the tool are exactly the same. Thus one flute may be larger than the other, resulting in larger chip removal by one flute than the other. These manufacturing variations account for differences in force magnitude from one flute to another. Also the wearlands of each flute may vary, further contributing to these differences. Finally, the air bearing spindle has runout on the order of 8-10 µm which would further accounts for differences in magnitude of cutting force on the different tool flutes during a revolution.

The impact of relatively slow sampling rates imposed by the motion program on controller performance was investigated. As stated previously, force measurements are acquired only once per execution of the “while loop” (752 Hz), not at the PMAC clock
frequency (2273 Hz). Therefore, for each spindle rotation at 10,000 rpm, force data from the A/D board is acquired only 4.5 times. For accurate deflection compensation, one would like to compensate for the maximum cutting force along the direction of interest. Inherent in the tool stiffness model is that the greater the force, the greater the deflection. Because the PMAC samples force only 4.5 times per spindle rotation (or 2.3 times per flute), the true peak force is not likely to be measured. Therefore, an external method of measuring peak force was needed to accurately compensate for tool deflection.

To ensure that maximum cutting forces are measured in real-time, a dSPACE 1102 data acquisition (DAQ) system was used in conjunction with the PMAC. This system acquires the x, y, and z-forces during machining at 10,000 Hz, filters the force data, captures the maximum force, and then calculates either the predicted cutting depth or predicted tool deflection. This prediction is then transferred as an analog signal to the PMAC for axis control. Because the dSPACE system operates at 10,000 Hz, no fewer than 60 force measurements are acquired per rotation of the tool (30 per tooth). By sampling at this frequency, the maximum force is likely to be read and thus the maximum deflection of the tool can be compensated and the desired profile achieved.

Figure 13 shows a diagram of the dSPACE algorithm used to filter the input data (see also Appendix B), calculate and hold the maximum force, and transfer a predicted cutting depth or predicted tool deflection to the PMAC controller.
Cutting forces are sampled and filtered using $5^{th}$ order Butterworth low-pass filters (Figure 14). In the vicinity of the cutoff frequency, this filter delivers a rounded amplitude response and gives excellent flatness in the lower portion of the passband. The high-frequency roll-off is $-20\text{dB}$ per pole above the cutoff frequency, which is set to 400 Hz to eliminate noise above the spindle frequency (333 Hz).
After filtering, a Simulink “maximum capture” function determines the peak force per spindle rotation. This maximum force is reset with every two tool rotations so that deflection compensation is local to two revolutions of the tool. A trigger pulse is acquired from a pulse generator in dSPACE. The logistics of Simulink’s “maximum capture” block necessitate an “absolute value” operation before capturing the peak. Figure 15 shows a plot of cutting force and maximum force captured by dSPACE during an experiment (feedrate = 100 mm/min, spindle speed = 10,000 rpm, depth = 70 µm).

Figure 14: Bode plot of Butterworth 5th order filter characteristics
Next, the predicted cutting depth or predicted tool deflection is calculated, and this value is transferred to the PMAC motion program as an analog signal. In the motion program, a first-order digital filter is implemented as part of the force feedback compensation. Details of this low-pass filter design are presented below.

### 4.3.1 First-order Digital Filter

A first-order digital filter is used to filter the predicted cutting depth or tool deflection. Since the maximum force is reset with every two rotations of the tool, step changes in this predicted value occur from one revolution to the next. These changes could destabilize the axes controllers if the error into the control algorithm becomes larger than the step size used during axes tuning. The low-pass filter reduces abrupt variations in this predicted value, improving the robustness of the compensation algorithm. A first-order filter was implemented because of its simplicity and minimal computational cost of
PMAC calculation time. If a very complex filter was implemented, the execution time of the while loop would further decrease. The following filter form was used:

\[
\hat{\delta}_n (k + 1) = \beta \cdot \hat{\delta}_n (k + 1) + \alpha \cdot \hat{\delta}_n (k)
\]  

(7)

where:
- \( \hat{\delta}_n (k + 1) \) = the filtered prediction of normal tool deflection at time (k+1)
- \( \hat{\delta}_n (k + 1) \) = the unfiltered prediction of normal tool deflection at time (k+1)
- \( \hat{\delta}_n (k) \) = the filtered prediction of normal tool deflection at time (k)
- \( 0 < \alpha, \beta < 1 \) = filter coefficients \( \beta = 1 - \alpha \)

The determination of filter coefficients is non-trivial; coefficients were determined according to frequency-domain stability criteria. The transfer function of this filter is determined by taking the z-transform:

\[
z\{F(k)\} = f(z) = \sum_{k=1}^{\infty} f(k)z^{-1}
\]

(8)

\[
z\{f(k-1)\} = z^{-1}F(z)
\]

(9)

where:
- \( f(k) \) = discrete-time function at a particular sample (k)
- \( F(z) \) = z-transform of the discrete function

The transfer function of this first-order digital filter is:

\[
\frac{\Delta_n (z)}{\hat{\Delta}_n (z)} = \frac{\beta}{1 - \alpha z^{-1}}
\]

(10)
where:
\[
\Delta (z) = \text{filtered output of first-order digital filter transfer function}
\]
\[
\hat{\Delta} (z) = \text{unfiltered input to first-order digital filter transfer function}
\]

Setting the denominator of Equation (10) equal to zero gives the filter pole location in the z-plane (Figure 16).

Figure 16: z-plane for a discrete system

The following characteristics apply for the z-plane:

- The stability boundary is the unit circle \( |z| = 1 \)
- The small vicinity around the \( z = +1 \) in the z-plane is essentially identical to the vicinity around \( s = 0 \) in the s-plane
- The z-plane locations give response information normalized to the sample rate
- The negative real z-axis always represents a frequency of \( \omega/2 \) where \( \omega_s = 2\pi/T = \) sample rate in radians per second (where T is the sample time of controller)
- Vertical lines in the left half of the s-plane (the constant real part or time constant) map into circles within the unit circle of the z-plane and represent a stable system
- Horizontal lines in the s-plane (the constant imaginary part of the frequency) map into radial lines from the point of intersection of the unit circle and the real axis at +1.0 on the z-plane

The root of the denominator in equation (10), and hence the filter pole, is located at \( z = \alpha \). Therefore, choosing \( \alpha \) determines the filter’s cutoff frequency. Because the bandwidth of the z-axis was experimentally determined to be 14.2 Hz, this is the frequency used in determining the proper \( \alpha \) best suitting the bandwidth of the system. Setting the lead coefficient to \( \alpha = 0.99 \) (and consequently \( \beta = 0.01 \)) should theoretically give proper filtering for the natural frequency of the z-axis and the sample time of the motion program.

This lead coefficient has a significant effect on filter performance. Figure 17 shows a plot of filtered output to unfiltered input for a first-order digital filter with different lead coefficients. The input to this filter was a unit step function. Notice that the smaller the lead coefficient, the longer the time required for the filtered output to equal the input value. Thus if the input to the filter is a periodic function, the output magnitude will be attenuated if the lead coefficient is small. This attenuation effect must obviously be taken into account for closed-loop compensation of deflection.
The time constant of a first-order digital filter is the time needed for its step response to reach 63% of its final (steady-state) value. It is given by the following equation:

$$\tau = \frac{T_s(1 - \beta)}{\beta}$$

where:
- $T_s$ = sample time
- $\beta$ = coefficient of input to the filter

Note that the time constant is independent of the input step amplitude. The low-pass filter used in the deflection compensation experiments had a relatively strong filter coefficient of $\beta = 0.01$, resulting in a time constant, $\tau$, is 0.1316 s. A typical unit step
filter response is shown in Figure 18 where the delayed response is apparent ($\tau_s = 0.1316$ s, $\beta = 0.01$).

**Figure 18**: Time constant of first-order digital filter

The attenuation factor of a low-pass first-order filter is given by:

$$a = \frac{1}{\sqrt{1 + (\omega \tau)^2}}$$

where:

- $a$ = amplitude factor ($a < 1$)
- $\omega$ = input signal input frequency
- $\tau$ = time constant of the filter
This attenuation factor is a multiplier by which the filter output signal is scaled in relation to the input signal. Notice it depends only on frequency of the input and time constant. Thus, inputs with slowly-varying characteristics are less likely to be attenuated, meaning the filtered output closely resembles the unfiltered input. Inputs with rapidly-varying characteristics will be strongly attenuated.
5 EXPERIMENTAL EVALUATIONS

To evaluate the effectiveness of the force feedback deflection compensation algorithms, extensive series of machining experiments were conducted on the Nanoform 600 DTM. Groove profiles were machined in S-7 tool steel samples (hardness > 55 Rc) mounted to the three-axis load cell (Figure 4). The workpiece was aligned with the axes of the Nanoform and cuts were made along the x-axis (left to right) with the tool tilted at 25 degrees from the z-axis to emphasize the effects of tool deflection (Figure 19). The following sections detail these experimental evaluations.

Figure 19: Small groove experimental setup
5.1 PART MEASUREMENT ON TALYSURF PROFILOMETER

Each of the experiments presented in this paper required post-machining measurements of groove profiles to assess controller performance. Each of these measurements was made using a Talysurf profilometer. The measurement axis the Talysurf probe had to be carefully aligned parallel with the machined groove. Otherwise a skewed profile would result, as the probe would not travel parallel to the center of the groove. Since the bottom of the groove is a radius, the cross-sectional minima of the groove also had to be found to ensure the length measurement was along the groove bottom.

This alignment of the groove with the Talysurf measurement axis was accomplished through an iterative process of measuring the groove at its start and end, and rotating the workpiece until alignment was achieved. Once this alignment was achieved, the groove’s cross-sectional minima at a particular point was found, and measurement of the groove profile along its length was made accordingly.

Figure 20 illustrates this alignment process. First, the workpiece was placed on the Talysurf and contact was made at point B. The minima of the groove’s cross-section was then found by moving the workpiece rectilinearly in the y-direction. The probe was then moved to point A where the minima was again found while making note of the relative position along the y-direction of the minima at point A with respect to the minima at point B. The workpiece was then rotated clockwise or counterclockwise, depending on the relative location of the two minimas. This procedure was then repeated and the position along the y-direction of the minimas at points A and B compared. Once these minimas had the same relative y-position, the groove was assumed to be aligned with the Talysurf measurement axis. The part was then moved rectilinearly in the y-direction until the minima of the profile cross-section was found, and measurement along the groove length was then made.
5.2 LINEAR SLOTTING CUT EXPERIMENTS

Slotting cuts with linearly varying depth were made with and without compensation to evaluate the performance of real-time deflection compensation. Slots spanning 20 mm and 0-80 μm in depth were programmed, using a spindle speed of 10,000 rpm and a feed rate of 100 mm/min. This spindle speed was chosen because this was the minimum operational spindle speed to still have enough torque to make cuts in tool steel, and the feedrate was chosen to give a chip load of 5 μm/flute (which is on the order of the wearland of the tools and is acceptable machining conditions). Groove profiles were measured on the Talysurf profilometer as described in the previous section. All experiments were performed with the tool tilted at an angle of 25 degrees. This angle was chosen to emphasize the effects of tool deflection, and other tilt angles were not investigated.

Figure 20: Small groove Talysurf alignment procedure
5.2.1 **Experimental MultipliCation Factor**

First, the predicted depth compensation method (5) was compared to uncompensated machining. For this specific experiment, the dSPACE 1102 DAQ system was not yet available, thus a PMAC implementation was used. This experiment tested the feasibility of deflection compensation with existing PMAC hardware and software before incorporating the dSPACE system.

When using the PMAC to acquire data, the actual maximum cutting force during each sample is not likely to be acquired. As stated earlier, the PMAC “while loop” executes only 4.5 times as fast as the spindle frequency. Therefore, PMAC acquisition of cutting force gives approximately four data points for every revolution of the tool. This causes the gathered force to resemble a square wave (Figure 21).

![Figure 21: PMAC acquired cutting force vs. actual cutting force](image-url)
Real-time force filtering gives average force information that can be scaled to extrapolate the actual maximum force. This scaling of the gathered force to achieve the maximum force was done using a multiplication factor determined experimentally. Actual force data for a cutting operation acquired with the PMAC is shown in Figure 22, which also shows the effects of force filtering.

![Figure 22: PMAC acquired cutting force and filtered force](image)

The extreme variations in sampled cutting force (Figure 22) made the analytical determination of a multiplication factor difficult. This multiplication factor was determined experimentally from an uncompensated groove test. For this case, a cut was made to a depth of 80 µm without force feedback. During the cut, the force was measured using PMAC and deflection predicted based on a model of tool stiffness. Then the groove profile was measured on the Talysurf profilometer, and the actual deflection
was compared to the predicted deflection based on the real-time cutting force without a multiplication factor. By comparing these deflections, an experimental multiplication factor was determined. A plot of the predicted deflection without a multiplication factor added during machining is given in Figure 23.

![Graph showing predicted deflection](image)

**Figure 23**: Predicted deflection during uncompensated cut without multiplication factor

It is seen that the maximum predicted deflection without a multiplication factor is 9.23 µm. The actual deflection measured on the Talysurf profilometer is 26.86 µm. Therefore the experimental multiplication factor was determined to be 2.90, and this value was used in the predicted depth compensation experiments described below.

### 5.2.2 Predicted Depth Compensation
Typical results for closed-loop compensation using predicted cutting depth and a non-dynamic cutting force model are shown in Figure 24, with the upper plot representing the uncompensated cutting profile, the middle plot representing the compensated profile, and the lower plot representing the desired profile.

![Graph showing uncompensated, compensated, and desired profiles.](image)

**Figure 24:** Experimental profile measurements: predicted depth compensation, linear slotting cut

Profile errors (Figure 25) were not significantly improved at the start of the groove, as both the compensated and uncompensated cases reveal similar errors (approximately 0-6 µm) for the first 8 mm of cutting. From this point forward, however, predicted depth compensation significantly reduced profile errors and produced more desirable results. Maximum error in the compensated groove is on the order of 8 µm (at a horizontal...
location of 8 mm), while maximum error (deflection) in the uncompensated groove reaches a maximum of 14 \( \mu \text{m} \) at a location of 18 mm. This equates to a 43\% reduction in groove profile error.

![Graph showing uncompensated and compensated profile errors](image)

**Figure 25:** Experimental profile errors: predicted depth compensation, linear slotting cut

There are several reasons for this lack of error reduction at the start of the groove. One involves implementation issues associated with the Nanoform 600 DTM. When transitioning from encoder feedback to force feedback, measurable force needs to exist for cutting depth prediction to prevent a tool crash into the workpiece. Therefore, this method of compensation must be implemented after the tool begins to impart a force on the workpiece. A rotating tool is incremented toward the workpiece until a small force (on the order of 0.1 N) is measured. The axes are disabled and then the mode of machine operation is changed from encoder feedback to depth prediction compensation (force
feedback). The axes are enabled, and the experiment begins. Initially, the desired cutting depth is zero. However, there is a measurable force, thus the predicted cutting depth is greater than zero. As a result, the control algorithm commands an initial move away from the workpiece, resulting in significant profile errors at the start of the groove.

Another source of error in the compensated profile involves the theoretical cutting depth vs. normal cutting force relationship (Figure 2). This plot reveals a very small slope at small cutting depths, making it difficult to measure forces at this operating condition due to their intermittent behavior. Thus, a small change in depth has a large effect on the force and limits this method’s accuracy. As the depth increases, the forces become less intermittent (more consistent), resulting in more accurate predictions of cutting depth.

It should be noted that the results shown in Figure 24 and Figure 24 represent the only acceptable results out of approximately 35 experiments conducted using this method of compensation. As a result, the second method of compensation using deflection was investigated to a much greater extent.

5.2.3 Predicted Deflection Compensation

Next, the predicted deflection compensation method (6) was compared to uncompensated machining. A typical set of results is shown in Figure 26, with the upper plot representing the uncompensated cutting profile, the middle plot representing the compensated profile, and the lower plot representing the desired profile.
Normal cutting force during a typical linear slotting cut is shown in Figure 27. This cutting force data shows unfiltered force as well as maximum force and filtered maximum force.
Profile errors (Figure 28) were significantly and consistently improved throughout the cutting experiment. Maximum profile errors with predicted deflection compensation are approximately 4 µm, compared to 21 µm with the uncompensated cut. This equates to an 80% reduction in groove profile error. In contrast to the predicted depth compensation, errors are initially small and remain small throughout the experiment (between +4 µm and 0 µm), resulting in depth error at the end of the groove of -2 µm.
Slotting cuts with harmonically varying depth were also made with and without predicted deflection compensation to evaluate the performance of this approach. Trials were not conducted using predicted depth compensation due to problems with initial transients leading to tool breakage.

5.3.1 ONE-PERIOD MODIFIED SINUSOIDAL PROFILE

Grooves spanning 20 mm with a one-period modified sinusoidal profile and peak depth of 80 µm were programmed, using a spindle speed of 10,000 rpm and a feed rate of 100
mm/min with the long tool. The resulting sine wave frequency was 0.52 Hz. As before, groove profiles were measured on the Talysurf profilometer.

A typical set of results is shown in Figure 29, with the upper plot representing the uncompensated cutting profile, the middle plot representing the compensated profile, and the lower plot representing the desired profile.

![Diagram showing uncompensated, compensated, and desired profiles.](image)

**Figure 29**: Experimental profile measurements: predicted deflection compensation, one period sinusoidal slotting cut

Normal cutting force during a typical one period sinusoidal slotting cut is shown in Figure 30. This cutting force data shows unfiltered force as well as maximum force and filtered maximum force.
Profile errors (Figure 31) were significantly and consistently improved throughout the cutting experiment. Maximum profile errors with predicted deflection compensation are approximately 11 µm, compared to 23 µm with the uncompensated cut, a 53% reduction in error. Profile errors at the start of machining are approximately –4 µm, meaning the actual profile depth is greater than desired.
Figure 31: Experimental profile errors: predicted deflection compensation, one period sinusoidal slotting cut
5.3.2 Two-Period Modified Sinusoidal Profile

Grooves spanning 20 mm with a two-period modified sinusoidal profile and peak depth of 80 µm were programmed, using a spindle speed of 10,000 rpm and a feed rate of 100 mm/min with the long tool. The resulting sine wave frequency was 1.05 Hz. As before, groove profiles were measured on the Talysurf profilometer.

A typical set of results is shown in Figure 32, with the upper plot representing the uncompensated cutting profile, the middle plot representing the compensated profile, and the lower plot representing the desired profile.

Figure 32: Experimental profile measurements: predicted deflection compensation, two period sinusoidal slotting cut
Normal cutting force during a typical one period sinusoidal slotting cut is shown in Figure 33. This cutting force data shows unfiltered force as well as maximum force and filtered maximum force.

![Normal Force graph](image)

**Figure 33**: Experimental force measurements: predicted deflection compensation, two period sinusoidal slotting cut

Error in the groove profile shown in Figure 34 is significantly improved throughout and the results show that this method of compensation reduced the form error from a maximum of 27 µm to 13 µm. This is a 52% reduction in error. It can be seen that groove profile error at the start as well as during the groove cut is bound between +14 µm and −1 µm.
Profile errors in all experiments result from many different sources. Capturing the maximum cutting forces and filtering these forces both make significant impacts on the profile created. Also, there are significant transient responses in the DTM axes that create variations in the modified sine wave cuts. Another source of error involves interpretation of the data measured on the Talysurf profilometer. These and other error sources are discussed in greater detail in section 5.5.
5.4 REPEATABILITY EXPERIMENTS

To evaluate the repeatability of the force feedback deflection compensation results of Figures 25-33, each of these cutting experiments was repeated with different tools. Closed-loop cutting tests were conducted with one tool, then repeated with a different tool on the same workpiece in a different location. The workpiece was not removed from the DTM fixture from one experiment to the next to eliminate misalignment errors.

5.4.1 LINEAR SLOTTING CUT EXPERIMENTS

A typical set of repeatability results is shown in Figure 35, with the upper plot representing the compensated profile errors for Tool 1, and the lower plot representing the compensated profile errors for Tool 2. These results are directly comparable to those of Figure 28.
Although these results indicate excellent repeatability, profile errors associated with Tool 1 are less than those associated with Tool 2. These variations can be attributed to differences in the workpiece surface before cutting as well as differences in tool stiffness and flute wear from one tool to the next.

As stated previously, the workpiece was not removed from the DTM fixture between cutting tests. However, “touching off” the surface of the part was necessary in each case, possibly introducing errors into the experiment. It is not likely that the precise surface location was found with the same accuracy each time, resulting in profile errors from one tool to the next that could not be determined from a Talysurf measurement only.
This “touching off” the workpiece surface is the same procedure described in section 5.2.2 where a rotating tool was incremented 1 µm in the z-direction toward the workpiece surface while measuring z-force on the load cell. Incremental movements were continued until a force of approximately 0.1 N was seen. At this point it was determined that the surface of the workpiece was found.

5.4.2 ONE-PERIOD MODIFIED SINUSOIDAL PROFILE

A typical set of repeatability results is shown in Figure 36, with the upper plot representing the compensated profile errors for Tool 1, and the lower plot representing the compensated profile errors for Tool 2. These results are directly comparable to those of Figure 31.
Figure 36: Experimental profile repeatability errors: predicted deflection compensation, one period sinusoidal slotting cut

As before, these experimental results reveal excellent correspondence between Tool 1 and Tool 2 in all areas of the groove.

5.4.3 Two-Period Modified Sinusoidal Profile

Another typical set of results is shown in Figure 37, with the upper plot representing the compensated profile errors for Tool 1, and the lower plot representing the compensated profile errors for Tool 2. These results are directly comparable to those of Figure 34.
This experiment reveals that the correspondence between both experiments with the same compensation algorithm and cutting parameters is very good. There is some error at the end of the groove profile for Tool 1 that is not seen for Tool 2. The reason for this is not clear however, as all parameters are the same and the part was not realigned between each experiment.
5.5 INVESTIGATING PROFILE ERRORS IN SMALL GROOVE EXPERIMENTS

The causes of profile errors in the small groove experiments are difficult to determine exclusively from Talysurf measurements. To investigate the effects of controller inputs on groove profiles, cuts were made while the following variables and parameters were measured:

- unfiltered cutting force
- filtered cutting force
- captured maximum cutting force
- encoder position
- filtered tool deflection
- proportional deflection compensation error
- derivative deflection compensation error
- integral deflection compensation error
- axes servo control voltages
- workpiece surface before cutting

Groove profile measurements were made and compared statistically with these acquired measurements. To make these statistical comparisons, cross-correlation coefficients were computed between each gathered variable and the measured groove profiles.

5.5.1 CROSS-CORRELATION COEFFICIENTS

The cross-correlation coefficient is a measure of the degree of linear relationship between two variables. The emphasis in cross-correlation is on the degree to which a linear model describes the relationship between two variables. This statistical measure is non-directional, and can take on any value between plus and minus unity ($|r| \leq 1.0$). The sign of the cross-correlation coefficient defines the direction of the relationship: a
positive coefficient means that as one variable increases, the other variable increases, and vice-versa. A negative coefficient indicates that as one variable increases, the other decreases, and vice-versa.

By taking the absolute value of the correlation coefficient, the strength of the relationship between one variable to another is quantified. The closer the absolute value of the correlation coefficient is to unity, the stronger the relationship between the two variables. A correlation coefficient of $\pm1.0$ indicates a perfect linear relationship whereas a correlation coefficient of 0 indicates the absence of a linear relationship.

The cross-correlation coefficient is also the slope of the regression line when both variables have been converted to z-scores. Conversion to a z-score takes place by subtracting out the mean from each raw variable and then dividing the result by the sample standard deviation. The cross-correlation coefficient is invariant to linear transformations, meaning that changing the scale of either variable in the cross-correlation does not affect the cross-correlation coefficient [14].

**5.5.2 UNCOMPENSATED LINEAR SLOTTING CUT: CROSS-CORRELATIONS**

An uncompensated linear slotting cut was made as outlined in section 5.2, and the variables highlighted in section 5.5 were acquired during cutting. For this uncompensated experiment, the correlations between cutting force to groove profile and controller variables to groove profile are easily separated, as encoder position is the only feedback variable (cutting force is measured but not used in the algorithm). Therefore, the effects of controller variables and cutting force on groove profile measurements should be apparent, as should the relationship between each. Figure 38 shows the filtered cutting force and maximum capture force for this uncompensated experiment.
The resulting profile measurement is shown in Figure 39, where the maximum cutting force appears to be strongly correlated to this profile. Comparing Figure 37 to Figure 38 at approximately 3 mm shows that both profiles have similar irregularities. A similar irregularity occurs at 11 mm and again near the end of the groove. Thus it is expected that the maximum cutting force and groove profile should have a strong correlation.

**Figure 38:** Experimental force measurements: filtered and captured maximum cutting force, uncompensated linear slotting cut
The workpiece surface profile before cutting is shown in Figure 40. This surface appears to have irregularities as well, and could possibly play an important role in the groove profile created. This figure reveals a significant variation in workpiece surface between 3 and 4 mm that appears to have an effect on cutting force. Therefore, it seems likely that cutting forces are dependent on workpiece surface profiles, and measured groove profiles are dependent on cutting forces.
To validate these expectations, experimental cross-correlations were performed. The cross-correlation results between cutting parameters, control algorithm variables, and groove profile measurements are presented in Table 4.
Table 4: Cross-correlation coefficients for uncompensated linear slotting cut

<table>
<thead>
<tr>
<th>Variable</th>
<th>Cross-correlation coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unfiltered cutting force (dSPACE)</td>
<td>-0.4808</td>
</tr>
<tr>
<td>Filtered cutting force (dSPACE)</td>
<td>-0.5134</td>
</tr>
<tr>
<td>Captured maximum force (dSPACE)</td>
<td>-0.8634</td>
</tr>
<tr>
<td>Filtered tool deflection (PMAC)</td>
<td>-0.9693</td>
</tr>
<tr>
<td>Z-axis encoder position (PMAC)</td>
<td>-0.9989</td>
</tr>
<tr>
<td>Proportional error (PMAC)</td>
<td>0.1881</td>
</tr>
<tr>
<td>Integral error (PMAC)</td>
<td>-0.2892</td>
</tr>
<tr>
<td>Derivative error (PMAC)</td>
<td>0.0612</td>
</tr>
<tr>
<td>Servo voltage (PMAC)</td>
<td>0.0499</td>
</tr>
<tr>
<td>Workpiece surface before cutting</td>
<td>-0.2097</td>
</tr>
</tbody>
</table>

The variables with the strongest correlation coefficients are PMAC z-axis encoder position (-0.9989), PMAC filtered tool deflection (-0.9693), and dSPACE captured maximum force (-0.8634). The strong correlation between PMAC encoder position and groove profile is expected; it implies that the tool tip is following axes movement and creating the profile measured on the Talysurf. There is also a strong cross-correlation between PMAC tool deflection and dSPACE captured maximum force with measured groove profile, so it can be concluded that the maximum force during machining has a strong relationship with groove profile created.

5.5.3 Compensated Linear Slotting Cut: Cross-correlations

A linear slotting cut was then made using the deflection compensation algorithm and the same cutting conditions outlined in section 5.6. The variables highlighted in section 5.5 were again acquired during cutting. In this experiment, the correlations between cutting force to groove profile are related, as the control algorithm includes force feedback. Figure 41 shows the filtered cutting force and the capture maximum force for the compensated experiment.
The resulting groove profile is shown in Figure 42. It can be seen that the maximum cutting force appears to be correlated to the groove profile. Comparing Figure 40 to Figure 41 at approximately 3 mm shows that both profiles have similar irregularities. A similar irregularity occurs at 11 mm and again near the end of the groove. Thus it is expected that the maximum cutting force and groove profile should have a strong correlation, and this is validated in the correlation coefficients presented below (Table 5).
Figure 42: Experimental profile measurement: compensated linear slotting cut

Table 5: Cross-correlation coefficients for compensated linear slotting cut

<table>
<thead>
<tr>
<th>Variable</th>
<th>Cross-correlation coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unfiltered cutting force (dSPACE)</td>
<td>-0.4623</td>
</tr>
<tr>
<td>Filtered cutting force (dSPACE)</td>
<td>-0.4936</td>
</tr>
<tr>
<td>Captured maximum force (dSPACE)</td>
<td>-0.8569</td>
</tr>
<tr>
<td>Filtered tool deflection (PMAC)</td>
<td>-0.9707</td>
</tr>
<tr>
<td>Z-axis encoder position (PMAC)</td>
<td>-0.9989</td>
</tr>
<tr>
<td>Proportional error (PMAC)</td>
<td>0.183</td>
</tr>
<tr>
<td>Integral error (PMAC)</td>
<td>-0.3003</td>
</tr>
<tr>
<td>Derivative error (PMAC)</td>
<td>0.0415</td>
</tr>
<tr>
<td>Servo voltage (PMAC)</td>
<td>0.0235</td>
</tr>
<tr>
<td>Workpiece surface before cutting</td>
<td>-0.2299</td>
</tr>
</tbody>
</table>

The variables with the strongest correlation to groove profile are the same as in the uncompensated case: PMAC z-axis encoder position (-0.9989), PMAC filtered tool deflection (-0.9693), and dSPACE captured maximum force (-0.8634). The strong cross-correlation between encoder position and groove profile is expected, and implies that the
tool tip is again following axes movements and creating variations in measured groove profiles.

Because of the high cross-correlation between PMAC tool deflection and dSPACE captured maximum force with groove profile, and the direct feedback of cutting force in the algorithm controlling the DTM axes, improvements to the peak force measurements during result in groove profiles with less error and reduced variance. This hypothesis was tested in further experiments (presented below) to further pinpoint and reduce machining errors.

5.5.4 Improved Maximum Force Capture Algorithm

As shown in Figure 43, the maximum capture algorithm used in the previous experiments resets with every two tool rotations. As a result, there are abrupt changes in measured force at every reset point. Due to the strong correlation between PMAC filtered tool deflection and groove profiles, an improved maximum capture algorithm (without these abrupt changes) should result in smoother profiles. Therefore, an improved maximum capture algorithm was devised by combining two maximum capture functions in dSPACE with different phases. The idea is that one maximum capture function looks for the local maximum force over two tool revolutions, and before this maximum capture resets to look for another local maximum, another maximum capture function grabs the maximum force from the first capture before its reset. This second maximum capture function would normally reset simultaneously with the first maximum capture, but because of the phase delay (90% of the period) it captures a maximum force just before the first maximum capture resets. A Simulink diagram of this improved maximum capture function is presented in Appendix C.
Figure 43: Old maximum force capture algorithm

Figure 44 shows how the improved maximum capture algorithm (with nested capture functions) reduces abrupt changes. The transition from one reset to another is much smoother, thus the captured maximum force should have less variation between tool rotations.
With this reduced abruptness and variation, the cross-correlation between maximum cutting force and groove profile is expected to be smoother with less profile error.

**5.5.5 Uncompensated Linear Slotting Cut: Cross-Correlations Using Improved Maximum Force Capture Algorithm**

The improved maximum capture algorithm was implemented in dSPACE for an uncompensated cutting experiment to determine whether the correlation between groove profile and maximum force improved. This cross-correlation of the improved maximum capture (to groove profile) was then compared to the cross-correlation of the previous capture method. Figure 45 shows the filtered cutting force and captured maximum force for the uncompensated experiment.
Figure 45: Experimental force measurements: filtered and improved captured maximum cutting force, uncompensated linear slotting cut

The reduced variation in captured maximum cutting force for this improved algorithm is clearly revealed by comparing Figure 45 with Figure 41. Specifically, the vertical width of the captured maximum force band is much smaller than that associated with the previous capture algorithm.

In comparison with the resulting groove profile (Figure 39), it can be seen that this improved maximum cutting force also appears to be strongly correlated to the measured groove profile. The maximum force outline of Figure 45 closely resembles the groove profile at approximately 4 mm into the cut, where both the force and groove profile exhibit similar variations. This same phenomena occurs at approximately 16 mm and again near the end of the groove.
Again it is expected that the improved captured maximum cutting force and groove profile should be strongly correlated. It is also expected that the cross-correlation coefficient should be significantly higher than the previous one (associated with the old captured maximum cutting force) because there are fewer abrupt changes. This expectation is confirmed by comparing cross-correlation coefficients for the uncompensated groove profile: the improved captured maximum force coefficient increases to –0.9693 as compared with –0.8634 for the previous algorithm.

This increased correlation between improved captured maximum force and groove profile should result in a smoother profiles and consequently less error. A conclusion can be made that the smoother this captured maximum force is the smoother and more accurate the groove profile will be.
5.5.6 **COMPENSATED LINEAR SLOTTING CUT: CROSS-CORRELATIONS WITH NEW MAXIMUM FORCE CAPTURE ALGORITHM**

To evaluate the improved maximum force capture algorithm for deflection compensation, a closed-loop linear slotting cut was made as outlined in section 5.5.3. Variables were acquired during cutting with the improved capture function to determine reductions in profile errors with this new method. Cross-correlation with the resulting groove profile were compared with past experiments to quantify performance improvements.

Figure 46 shows the filtered cutting force and captured maximum force for the compensated experiment with improved capture algorithm implemented.

![Graph showing experimental force measurements](image)

**Figure 46**: Experimental force measurements: filtered and improved captured maximum cutting force, compensated linear slotting cut
The resulting groove profile is shown in Figure 47. Also shown is the groove profile created for a cut made using the old maximum capture algorithm. The improved maximum capture algorithm clearly results in a smoother groove profile. There is less variation in the groove profile - a direct result of the reduced variation in captured maximum force used to calculate deflection in the compensation algorithm. As before, the maximum cutting force appears to be strongly correlated to the groove profile.

**Figure 47:** Experimental profile measurements: compensated linear slotting cut profile, old and improved maximum force capture

This relationship between cutting force and groove profile was determined through a cross-correlation between force and profile created. Statistical results, including those for control algorithm variables, are presented in Table 6.
Table 6: Cross-correlation coefficients for compensated linear slotting cut with new maximum force capture

<table>
<thead>
<tr>
<th>Variable</th>
<th>Cross-correlation coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unfiltered cutting force (dSPACE)</td>
<td>-0.4323</td>
</tr>
<tr>
<td>Filtered cutting force (dSPACE)</td>
<td>-0.4645</td>
</tr>
<tr>
<td>Captured maximum force (dSPACE)</td>
<td>-0.8348</td>
</tr>
<tr>
<td>Filtered tool deflection (PMAC)</td>
<td>-0.963</td>
</tr>
<tr>
<td>Z-axis encoder position (PMAC)</td>
<td>-0.9991</td>
</tr>
<tr>
<td>Proportional error (PMAC)</td>
<td>0.2484</td>
</tr>
<tr>
<td>Integral error (PMAC)</td>
<td>-0.2852</td>
</tr>
<tr>
<td>Derivative error (PMAC)</td>
<td>0.0415</td>
</tr>
<tr>
<td>Servo voltage (PMAC)</td>
<td>0.0434</td>
</tr>
<tr>
<td>Workpiece surface before cutting</td>
<td>-0.2263</td>
</tr>
</tbody>
</table>

The variables with the strongest cross-correlation with groove profile are the same as in the previous cases: PMAC encoder position (-0.9991), PMAC filtered deflection (-0.963), and maximum capture force (-0.8384).

It might be expected that a strong correlation exists between workpiece the surface before cutting and the groove profile, as the irregularities in groove profile tend to occur around irregularities in workpiece surface. Intuitively, less variation in workpiece surface should result in less variation in groove profile, and vice versa. However, the cross-correlation data suggests that there is not a strong relationship, as the correlation coefficient in only –0.2263.

5.6 ERROR SOURCES IN SMALL GROOVE CUTTING

The investigation of cross-correlation between different machining variables and resulting groove profiles for small groove experiments led to several important results. As discussed in sections 5.5.2-5.5.6, dSPACE captured maximum force and PMAC filtered tool deflection have the strongest relationships with groove profile. This suggests that, for closed-loop machining, removing variations and inaccuracies in peak cutting force results in reduced tool deflection variations and therefore smoother and more.
accurate groove profiles. Other sources of profile error were identified in these experiments, however, and are addressed in the following sections.

5.6.1 TALYSURF PROFILE MEASUREMENT ERRORS

One source of profile error involves Talysurf measurement processing and comparison to desired profiles. This error should not be confused with workpiece alignment error which occur while taking Talysurf measurements. Instead it involves interpreting the measured profiles. Figure 48 shows three groove measurements made on the Talysurf.

![Figure 48: Repeatability measurements made on Talysurf](image)

These three profile measurements were made on the same groove. For each measurement, the workpiece was removed from the Talysurf stage, repositioned and aligned according to the process outlined in section 5.1. Figure 48 shows that
measurements 2 and 3 are well aligned (left to right). However, measurement 1 is considerably shifted with respect to the other two. This alignment error is a result of data interpretation after making a Talysurf measurement, not the actual measurement process. Misalignments in measured data are difficult to separate from error introduced through compensation technique.

This misalignment problem arises from the fact that it is difficult to determine exactly where a groove begins and ends. Thus, the proper relationship between desired groove profile and measured groove profile from Talysurf data is difficult to determine. Small variations in alignment of the measured groove profile with respect to the desired can result in drastically different measurements of the error. All such errors are interpreted as performance limitations of the compensation algorithm, when in fact they result from misalignment of the measured groove profile to desired profile.

Another source of error related to processing and interpreting Talysurf profile measurements involves workpiece tilt. Following each measurement, a linear regression is performed to remove “workpiece tilt” from data. This regression is performed separately for each experimental groove, thus points used from one groove to the next are different and could lead to differences in measurement. Figure 49 shows profile measurement data with (modified profile) and without (unmodified profile) workpiece tilt removed.
Figure 49: Removing workpiece tilt from Talysurf profile measurement data

Figure 49 shows that before removing workpiece tilt from the data, the unmodified profile starts at \(-10\ \mu\text{m}\) and ends at \(10\ \mu\text{m}\). This effect results from the Talysurf stage or the workpiece surface being misaligned with the Talysurf probe tip axis. By removing this tilt from the data, the measurement can be compared to desired profile that starts at \(0\ \mu\text{m}\) deep and ends at \(80\ \mu\text{m}\).

The highlighted regions on the unmodified profile show the portions of the measurement used (by the linear regression) to remove workpiece tilt. Clearly the selection of reference points used in the regression process could lead to variations in the comparison to desired profiles.
5.6.2 Transient Response of DTM Axes

The DTM axes used in the small groove experiments have inherent transient response characteristics that affect the machining performance. These transient responses would typically be neglected in experiments with limited travel and federates. However, in experiments where the z-axis travel is on the order of 100 µm with federates of along the groove of 100 mm/min, the z-axis transient response has an effect on groove profile created. Figure 50 shows the transient response of the z-axis (without cutting) tracking a linear groove profile.

![Figure 50: Transient response of z-axis to a linear groove profile (non-cutting)](image)

At the start of motion, the z-axis has a small transient characteristic, an undershoot, where the axis actually moves backwards and then along the correct path. This transient effect is more evident in the two-period modified sinudoidal profile shown in Figure 51.
Similar undershoots are evident at the start of the groove and upon changes in direction at approximately 3, 6, and 9 seconds. Ultimately, these undershoots appear as groove profile errors and due to the limited bandwidth of the DTM, they are difficult to remove. Each DTM axis weighs approximately 500 pounds, thus controlling transients requires very large actuation forces and torques. As stated previously, the DTM is a high-precision machine not designed for rapid changes in cutting direction. Although this machine performs well when moving in straight lines at constant velocities, it is expected to have transient response problems when making cuts requiring direction reversals (modified sine waves, etc.).
5.6.3 First-Order Digital Filter Error

Even with the improved maximum force capture algorithm, the need for a first-order PMAC filter was still apparent. Cutting experiments made using the improved maximum force capture without this first-order filter are shown in Figure 52.

![Graph](image)

**Figure 52:** Experimental profile measurement: compensated linear slotting cut profile with improved maximum force capture without first-order filter

When compared to Figure 47, this figure clearly shows the benefits of PMAC filtering. Even though the improved maximum force capture algorithm significantly reduces measured force variations and improves profile characteristics, filtering is still necessary for acceptable performance.
5.7 LARGE GROOVE EXPERIMENTS

The experimental slotting cut results presented in sections 5.1-5.3 indicate that the tool stiffness model (6) accurately predicts tool deflections, as every profile measurement created with predicted deflection compensation was within 20% of the desired profile.

These cutting experiments were conducted using a fixed tool tilt and varying depths of cut. However, the situation for most machining operations is more complicated and involves varying tilt angles as well as varying depths of cut. To verify that predicted deflection compensation is effective for more general machining, more comprehensive “large groove” experiments were conducted. The goal of these experiments was to create surfaces in which the cutting forces on the ball end mill vary in direction and magnitude during machining. These experiments also investigated the effects of forces that act in the plane of the workpiece surface.

These “large groove” experiments involved fabricating a 0.5 mm deep groove in a test specimen using a 3.0 mm ball end mill and enlarging the groove by 100 µm using a 0.8 mm ball end mill. Figure 53 is a schematic of the large groove experimental setup, where finishing cuts were made in the +y direction (bottom to top).
The sweep angle ($\gamma$) was defined to be the angle between the vertical direction and the normal surface vector at the point of interest (as defined in Figure 54). The sweep angle range for these experiments was approximately ± 51°, and the desired depth of cut was 100 µm. The spindle speed was 10,000 rpm, the feedrate was 50 mm/min, and the cross feed was 25 µm/pass. The magnitude and direction of cutting forces during machining depended on the depth of cut as well as the sweep angle. The thrust force was assumed to start out primarily as an axial force and end primarily as a radial force. This change in tool force direction is illustrated in Figure 54, where the tool sweeps along the workpiece from left to right. This change in force direction has a dramatic effect on the deflection of the tool during machining due to the significant difference in axial and radial tool stiffness.
The air bearing spindle of the Nanoform 600 DTM has very limited torque at operating speeds below 60,000 rpm. However, the DTM’s y-axis has a limited feedrate capability, requiring low spindle speeds to achieve proper chip removal per revolution. Proper chip removal is defined as a feed per tool revolution on the order of the wearland of the tool. For these reasons, single-pass rough grooves were machined on a HAAS VF-1 CNC machine using a 3.0 mm ball end tool.

Figure 55 shows the resulting groove profile error for a single rough pass groove from the desired radius of 1.5 mm.
Subsequent finishing passes were performed on the Nanoform DTM using a 0.8 mm ball end tool, requiring accurate alignment of the workpiece with the x, y, and z axes. Figure 56 shows the 3.0 mm and 0.8 mm diameter ball end mills used in this experiment.
Finishing passes were made with and without compensation. Since the tool was oriented with the z-axis of the machine as shown in Figure 53 (zero tool tilt), deflection in the axial direction was assumed negligible to that in the radial direction. Hence compensation was implemented in the x-direction only. It was also assumed that the x-direction forces caused only radial deflections and the z-direction forces caused only axial deflections of the tool. Therefore, in the “large groove” experiments the tool path was altered in the x-direction, whereas in previous experiments the tool path was modified in the z-direction. Figure 57 shows profile measurements, made on a Talysurf profilometer, from an uncompensated cutting experiment.

For this cut, the tool entered the part at a sweep angle of -51° and moved through the workpiece. Cuts were made in 25 µm increments from left to right (Figure 54). The tool then exited the workpiece at a sweep angle of 51° (on the right side). The effects of tool flutes leaving the workpiece were evident when the sweep angle approaches 46° and tool
deflections begin to decrease. The maximum profile errors in the uncompensated case range from 35 µm to –28 µm, giving a peak-to-valley error of 63 µm.

![Graph showing error vs. sweep angle](image)

**Figure 57:** Experimental profile measurement: uncompensated large groove error from 1.6 mm radius

Results using predicted deflection compensation are presented in Figure 58. As before, the cutting parameters were a feedrate of 50 mm/min, 10,000 rpm spindle rotational speed, cross feed of 25 µm/pass, and a 0.1 mm desired depth. Figure 58 shows the error from a best-fit radius of 1.6 mm for both the compensated and uncompensated profile.
Figure 58: Experimental profile errors: predicted deflection compensation, large groove experiment

Figure 58 reveals that the peak error is reduced from 63 µm in the uncompensated case to 18 µm using force feedback compensation, a 71% reduction. The 1.6 mm desired radius was achieved, producing an error plot that is neither convex nor concave. Peak-to-valley error was reduced to 20 µm, a 68% reduction. Table 7 shows the peak-to-valley error comparison for the compensated and uncompensated cases. A “best fit radius” was determined for both cases, also shown in Table 7. Compensated groove profiles match the desired radius by 1.6%, while uncompensated profiles are 7% from the desired radius.

| Table 7: Large groove experimental results: deflection compensation vs. uncompensated |
|---------------------------------|------------------|----------------|
| experiment                      | best fit radius (mm) | P-V error (µm) |
| uncompensated                   | 1.491             | 63             |
| compensated: case 1             | 1.573             | 20             |
| compensated: case 2             | 1.523             | 28             |
5.7.1 REPEATABILITY OF PREDICTED DEFLECTION COMPENSATION FOR LARGE GROOVE EXPERIMENTS

Ten large grooves were fabricated on each workpiece using the HAAS machine. Thus once the part was aligned with the Nanoform, several different finishing cuts could be made without having to realign the workpiece. This made repeatability comparisons feasible without having to account for alignment issues. Two different error profile measurements using deflection compensation are presented in Figure 59.

![Graph showing error profile](image_url)

**Figure 59:** Experimental profile repeatability errors: predicted deflection compensation, large groove experiment

Differences in profile errors can be attributed to having to “touch off” the tool for each experiment. It is unlikely that the touch off for all grooves was exactly the same, thus
some errors were introduced. The accuracy of the Nanoform is considerably better than
the HAAS machine, thus additional errors were introduced because grooves machined on
the HAAS were not perfectly parallel. Nevertheless, these “large groove” experiments
showed that force feedback compensation reduced overall error of the finished part.
However, it is difficult to differentiate between error introduced through the HAAS
machine, misalignment of the part with respect to the Nanoform, and error from the
compensation algorithm.

Another source of error was the limited bandwidth of the Nanoform, resulting in transient
axes responses. When making cuts starting out of the part and moving into the part, a
force and material is suddenly encountered, and transient response occurs leading to
error.
6 CONCLUSION

The forces generated during milling with a miniature ball end tool are relatively small (less than 10N) because of the limited size and strength of the tool edge. However, tool deflections can be a significant source of profile error because of low radial stiffness. Two force feedback approaches to tool deflection compensation are presented with advantages and disadvantages to each. The compensation algorithms presented here provide an avenue to reduce fabrication times and improve surface accuracy for hardened steel mold dies.

Predicting cutting depth using real-time cutting force measurements and a non-dynamic cutting force model can be used to compensate for errors arising from tool deflections and workpiece misalignment. Predicting tool deflection allows for compensation in profile error using force as well as machine encoder position feedback with good reliability. In this research, predicting deflection proved to be more robust and effective than predicting depth when implemented on the Nanoform DTM.

There are several reasons that predicting deflection yields better results than predicting depth in the compensation of profile errors. These include limited execution rate of the control algorithm, limited knowledge of actual cutting parameters, use of a multiplication factor in the depth prediction experiments, and use of laser feedback in the predicted deflection algorithm. When implemented on a DTM with higher bandwidth and faster execution rates, depth prediction may prove to be a viable approach to deflection compensation.

Reduction in error in small groove profiles over non-compensated groove profiles can be as great as 80% depending on the profile and method of compensation. The success of force feedback compensation depends heavily on the acquisition, processing, and filtering of cutting forces during machining. As shown in the cross-correlation analysis,
maximum cutting force has a strong relationship to the groove profile created. Therefore, smoother and more accurate maximum force measurements will result in smoother and more accurate groove profiles.

For the large groove experiments, reductions in profile error were not as significant as the small groove experiments. Several factors account for this difference: fabrication of the grooves on two different machines, alignment issues, touch off procedures, force measurement, and inherent complexity of the large groove experiments. Results show a groove was created with a best-fit radius differing by 27 µm from the desired radius in comparison with 109 µm for the uncompensated case. Peak to valley error from a radius of 1.6 mm was reduced from 63 µm to 20 µm when compared with a non-compensated cutting experiment.

Overall, force feedback control of miniature ball milling proved applicable in many situations for correcting errors in machined parts due to tool deflection. Significant reductions in profile error for large and small groove experiments were documented, with error reduction as high as 80%. The primary contribution of this research is a real-time feedback compensation algorithm that is capable of reducing form errors in workpieces machined using miniature ball end mills. In summary, the experimental results show that the force feedback control algorithms developed for this project were successful.
7 REFERENCES


APPENDIX A – CUSTOM G-CODE PROGRAM FOR NANOFORM

A custom g-code program was written allowing the user to input a starting and ending point in familiar “g-code” CNC language (callxx x0.0 y0.0 z0.0 f100.0). This program was written due to PMAC having internal acceleration limits not allowing fast enough execution of a program and experiment if a groove/cut was broken up into many small increments. Thus this g-code program takes a starting and ending point in (x,y,z) coordinates and transforms this starting (x,y,z) and ending (x,y,z) to a mathematical function of position of axes vs. move time depending on the feedrate. Therefore, the resolution that a groove can be broken down into is a function of the execution time of the PMAC processor rather than acceleration limits internally in the controller. This g-code program runs in motion programs on the PMAC card while the PMAC believes all axes to be in a dwell state. Numerous programs were written to perform sinusoidal cuts, etc., hence, any motion is possible if parametrically defined as a function of time based on the internal PMAC clock.
A.1 Outline of Custom G-code

open the program in a buffer
clear out program buffer
delete gather buffer
define gather buffer

define the following variables:
safety variable to determine maximum depth into part (mm)
lead coefficient for first-order digital filter

start "open loop" control of three axes

call program 10 (initialization, open loop mode, hold starting position)
set p1001 = 1 to start manual movement (after resetting amplifiers)
call program 14 (manual movement of axes, set p1002 = 1 to start cutting)

set (x,y,z) position to 0 mm

call subroutines for cutting and movement (main body of program)

…
…
…
call holding position at end of program (program 14)
end of program

Subroutine Descriptions

program 11 - open loop linear move
-pass ending x, y, z positions and feed rate
-syntax: call11 x10. y10. z10. f100.

program 12 - model prediction compensation linear move
-pass ending x, y, z positions and feed rate
-syntax: call12 x10. y10. z10. f100.

program 13 - stiffness prediction compensation linear move
-pass ending x, y, z positions and feed rate
-syntax: call13 x10. y10. z10. f100.

program 17 - open loop sine wave move
-pass amplitude, frequency, phase, depth offset, ending x, y, z, positions and feedrate
-syntax: call17 a0.04 w1.047 p4.71 d0.04 x0.0 y0.0 z0.0 f100.0
program 18
-stiffness prediction compensation sine wave move
-pass amplitude, frequency, phase, depth offset, ending x, y, z, positions and feedrate
-syntax: call17 a0.04 w1.047 p4.71 d0.04 x0.0 y0.0 z0.0 f100.0
A.2 Main Program and User Interface

This is the user interface where the operator specifies starting and ending positions, feedrates, and compensation methods. Also in this motion program the values for tool stiffness, lead coefficient for the first-order digital filter, safety limits for the axes, and other factors in the program are specified. This program was designed so that motion program 9 is the only program an operator would change to make different movements with different compensation, etc..

open prog 9
clear
command"delete gather"
command"define gather"

// set up of cutting parameters
p955 = 0.15  // safety variable to determine max depth into part (mm)
             // safety triggered: p1005 = 1
p914 = 0.1   // lead coefficient for force filtering

// start "open loop" control of three axes

call10       // calls program 10
             // set p1001 = 1 to start manual movement
call14       // manual movement of axes, set p1002 = 1 to start

m900 = 0     // set x position to 0 mm
m901 = 0     // set y position to 0 mm
m902 = 0     // set z position to 0 mm

// call subroutines for cutting and movement (main body of program)
// enclose move command with 'command"gather"' and 'command"endg"' to gather data

command"gather"
call11 x20.0 y1.0 z-1.0 f100.0  // open loop move - no compensation
command"endg"
call11 x20.0 y0.0 z-1.0 f100.0  // open loop move - no compensation
call11 x0.0 y0.0 z-1.0 f200.0   // open loop move - no compensation

//call12 x0.0 y0.0 z0.0 f100.0 // closed loop model depth prediction compensation
//call13 x0.0 y0.0 z0.0 f100.0 // closed loop stiffness deflection compensation

//call17 a0.025 w1.05 p4.71239 d0.025 x20.0 y0.0 f100.0 // open loop sine wave cutting

//call18 a0.025 w1.05 p4.71239 d0.025 x20.0 y0.0 f100.0 // closed loop sine wave cutting

...  
...
...

// hold positioning at end of all movement

call14  // holds last programmed position...at this point user may
        // manually enter (x,y,z) based on (p909,p910,p911)
        // to make axes move to desired new position...however,
        // this now equivalent to step movement command (careful
        // execution for stability

close   // end of program
A.3 \textbf{INITIALIZATION PROGRAM}

// variable definitions of registers (m-variables)

m0->X:$0,24 // 24 bit counter (once per servo)
m801->Y:$c006,8,16,s // x force register
m802->Y:$c007,8,16,s // y force register
m803->Y:$c00e,8,16,s // z force register
m900->D:$002b // measured position of motor number 1 (x-axis)
m901->D:$0067 // measured position of motor number 2 (y-axis)
m902->D:$00a3 // measured position of motor number 3 (z-axis)
m910->X:$07f0,24 // free 24 bit register
m911->X:$07f1,24 // free 24 bit register
m912->X:$07f2,24 // free 24 bit register
m913->X:$07f3,24 // free 24 bit register
m914->X:$07f4,24 // free 24 bit register
m915->X:$07f5,24 // free 24 bit register
m921->Y:$c003,8,16,s // DAC output 1 register
m922->Y:$c002,8,16,s // DAC output 2 register
m923->Y:$c00b,8,16,s // DAC output 3 register

open prog 10
clear

// variable definitions of various p-variables (initialization)

m900 = 0 // set x position to 0 mm
m901 = 0 // set y position to 0 mm
m902 = 0 // set z position to 0 mm

p900 = 800000 // scaling factor of x axis to get position in mm
p901 = 100000 // scaling factor of y axis to get position in mm
p902 = 800000 // scaling factor of z axis to get position in mm

p903 = 2.534 // scaling factor of load cell in x direction (N/V)
p904 = 2.818 // scaling factor of load cell in y direction (N/V)
p905 = 1.883 // scaling factor of load cell in z direction (N/V)

p906 = 0 // measured encoder position of x axis in mm
p907 = 0 // measured encoder position of y axis in mm
p908 = 0 // measured encoder position of z axis in mm

p909 = 0 // desired x position (as a function of time)
p910 = 0 // desired y position (as a function of time)
p911 = 0       // desired z position (as a function of time)
p912 = 0       // calculated total move distance from last position (mm)
p913 = 0       // calculated total move time (ms)
p914           // lead coefficient for filtering of force
p915           // model prediction quadratic coefficient
p916           // model prediction linear coefficient
p917           // stiffness value of tool in z direction (m/N)
p918 = 0       // model predicted z depth (mm)
p919 = 0       // calculated stiffness deflection in z direction (mm)
p920 = 0       // total error in x direction (mm)
p921 = 0       // total error in y direction (mm)
p922 = 0       // total error in z direction (mm)
p923 = 0       // derivative error x axis
p924 = 0       // derivative error y axis
p925 = 0       // derivative error z axis
p926 = 0       // integral error x axis
p927 = 0       // integral error y axis
p928 = 0       // integral error z axis
p929 = 0       // last error for comparison with derivative in x axis
p930 = 0       // last error for comparison with derivative in y axis
p931 = 0       // last error for comparison with derivative in z axis
p932 = 120000  // proportional gain for x axis
p933 = 55000   // proportional gain for y axis
p934 = 150000  // proportional gain for z axis
p935 = 1100    // derivative gain for x axis
p936 = 150     // derivative gain for y axis
p937 = 1200    // derivative gain for z axis
p938 = 600000  // integral gain for x axis
p939 = 1400000 // integral gain for y axis
p940 = 600000  // integral gain for z axis
p941 = 0       // filtered x force average (N)
p942 = 0       // filtered y force average (N)
p943 = 0       // filtered z force average (N)
p944 = 0       // scaled raw x force (N)
p945 = 0       // scaled raw y force (N)
p946 = 0           // scaled raw z force (N)
p950 = 0           // z proportional output
p951 = 0           // z integral output
p952 = 0           // z derivative output
p953 = 0           // z total output
p955   // safety variable to determine max depth into part (mm)
p956   // stiffness value of tool in y direction (m/N)
p957   // multiplication factor for stiffness based on filtering
p960 = 0           // incremental x movement
p961 = 0           // incremental y movement
p962 = 0           // incremental z movement
p963 = 0           // last position in x
p964 = 0           // last position in y
p965 = 0           // last position in z
p1000 = 1          // emergency stop variable number 1
p1001 = 0          // synchronous variable number 1
p1002 = 0          // synchronous variable number 2
p1003 = 0          // synchronous variable number 3
p1004 = 0          // synchronous variable number 4
p1005 = 0          // synchronous variable number 5
p1006 = 0          // synchronous variable number 6
p1007 = 0          // synchronous variable number 7
p1008 = 0          // synchronous variable number 8
p1009 = 0          // synchronous variable number 9
p1010 = 0          // synchronous variable number 10

// put motors in open loop

// put motor 1 in open loop mode

i129 = 0            // output offset = DAC output
i130 = 0            // proportional gain = 0
i163 = 0            // integration limit = 0
i133 = 0            // integral gain = 0
i111 = 0            // disable following limit
i131 = 0            // derivative error = 0

// put motor 2 in open loop mode

i229 = 0            // output offset = DAC output
i230 = 0 // proportional gain = 0
i263 = 0 // integration limit = 0
i233 = 0 // integral gain = 0
i211 = 0 // disable following limit
i231 = 0 // derivative error = 0

// put motor 3 in open loop mode
i329 = 0 // output offset = DAC output
i330 = 0 // proportional gain = 0
i363 = 0 // integration limit = 0
i333 = 0 // integral gain = 0
i311 = 0 // disable following limit
i331 = 0 // derivative error = 0

// loop to hold position at start of cutting program

// initialize timer variables specific to cutting program
m910 = m0 // initialize timer
m911 = 0 // initialize second timer
m912 = 0 // initialize third timer

While (p1000 = 1 And p1001 = 0)
    // start timer
    m912 = m911 // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608 // elapsed time since start of cutting (ms)

// calculation of measured position in mm
p906 = m900/(i108*32)/p900 // measured encoder x position (mm)
p907 = m901/(i208*32)/p901 // measured encoder y position (mm)
p908 = m902/(i308*32)/p902 // measured encoder z position (mm)

// calculation of desired position
p909 = 0 // desired x position (mm)
p910 = 0 // desired y position (mm)
p911 = 0 // desired z position (mm)

// alignment position error
p920 = p909 - p906 // total x error (mm)
p921 = p910 - p907 // total y error (mm)
p922 = p911 - p908     // total z error (mm)

// calculate derivative and integral error

p923 = p920 - p929     // derivative x error (mm)
p924 = p921 - p930     // derivative y error (mm)
p925 = p922 - p931     // derivative z error (mm)

p929 = p920         // reset last x error (mm)
p930 = p921         // reset last y error (mm)
p931 = p922         // reset last z error (mm)

If (p926 > 1)     // watch for integral error x saturation
    p926 = 1     // integral error x (saturated)
Else
    p926 = p926 + p920 // integral error x (unsaturated)
EndIf

If (p927 > 1)     // watch for integral error y saturation
    p927 = 1     // integral error y (saturated)
Else
    p927 = p927 + p921 // integral error y (unsaturated)
EndIf

If (p928 > 1)     // watch for integral error z saturation
    p928 = 1     // integral error z (saturated)
Else
    p928 = p928 + p922 // integral error z (unsaturated)
EndIf

// calculate control voltage to motor

i129 = p932*p920 + p938*(m911 - m912)/1000*p926 + p935/((m911 - m912)/1000)*p923
i229 = p933*p921 + p939*(m911 - m912)/1000*p927 + p936/((m911 - m912)/1000)*p924
i329 = p934*p922 + p940*(m911 - m912)/1000*p928 + p937/((m911 - m912)/1000)*p925
EndWhile

p1001=0

return

close
A.4 OPEN-LOOP LINEAR MOVEMENT PROGRAM

open prog 11
clear
read(x,y,z,f)  // x = q124 (mm), y = q125 (mm), z = q126 (mm), f = q106 (mm/min)

// calculate the incremental movement distance to be used in the desired position

p960 = q124-p963
p961 = q125-p964
p962 = q126-p965

// calculation of move time for each cut

p912 = sqrt(((p960)*(p960))+((p961)*(p961))+((p962)*(p962)))  // total distance of groove length (mm)
p913 = p912/q106*60*1000  // total move time to complete groove (ms)

// loop to cut open loop or just make open loop move

// initialize timer variables specific to cutting program

m910 = m0  // initialize timer
m911 = 0  // initialize second timer
m912 = 0  // initialize third timer
p926 = 0  // initialize x integral error
p927 = 0  // initialize y integral error
p928 = 0  // initialize z integral error
p929 = 0  // initialize last x error (derivative)
p930 = 0  // initialize last y error (derivative)
p931 = 0  // initialize last z error (derivative)

While (p1000 = 1 And p1002 = 0)
  // start timer
  p1023=m911  // time to execute one while loop

  m912 = m911  // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608  // elapsed time since start of cutting (ms)

  // calculation of measured position in mm

  p906 = m900/(i108*32)/p900  // measured encoder x position (mm)
p907 = m901/(i208*32)/p901  // measured encoder y position (mm)
p908 = m902/(i308*32)/p902  // measured encoder z position (mm)
// calculation of desired position

\[
p_{909} = \left(\frac{p_{960}}{p_{913}}\right)m_{911} + p_{963} \quad // \text{desired x position (mm)}
\]

\[
p_{910} = \left(\frac{p_{961}}{p_{913}}\right)m_{911} + p_{964} \quad // \text{desired y position (mm)}
\]

\[
p_{911} = \left(\frac{p_{962}}{p_{913}}\right)m_{911} + p_{965} \quad // \text{desired z position (mm)}
\]

// alignment position error

\[
p_{920} = p_{909} - p_{906} \quad // \text{total x error (mm)}
\]

\[
p_{921} = p_{910} - p_{907} \quad // \text{total y error (mm)}
\]

\[
p_{922} = p_{911} - p_{908} \quad // \text{total z error (mm)}
\]

// calculate derivative and integral error

\[
p_{923} = p_{920} - p_{929} \quad // \text{derivative x error (mm)}
\]

\[
p_{924} = p_{921} - p_{930} \quad // \text{derivative y error (mm)}
\]

\[
p_{925} = p_{922} - p_{931} \quad // \text{derivative z error (mm)}
\]

\[
p_{929} = p_{920} \quad // \text{reset last x error (mm)}
\]

\[
p_{930} = p_{921} \quad // \text{reset last y error (mm)}
\]

\[
p_{931} = p_{922} \quad // \text{reset last z error (mm)}
\]

//If (p_{926} > 1)  // watch for integral error x saturation
//   p_{926} = 1  // integral error x (saturated)
//Else
\[
p_{926} = p_{926} + p_{920} \quad // \text{integral error x (unsaturated)}
\]
//EndIf

//If (p_{927} > 1)  // watch for integral error y saturation
//   p_{927} = 1  // integral error y (saturated)
//Else
\[
p_{927} = p_{927} + p_{921} \quad // \text{integral error y (unsaturated)}
\]
//EndIf

//If (p_{928} > 1)  // watch for integral error z saturation
//   p_{928} = 1  // integral error z (saturated)
//Else
\[
p_{928} = p_{928} + p_{922} \quad // \text{integral error z (unsaturated)}
\]
//EndIf

// calculate control voltage to motor

\[
i_{129} = p_{932}p_{920} + p_{938}(m_{911} - m_{912})/1000p_{926} + p_{935}/((m_{911} - m_{912})/1000)p_{923}
\]
\[ i229 = p933*921 + p939*(m911 - m912)/1000*927 + p936/((m911 - m912)/1000)*924 \]
\[ i329 = p934*922 + p940*(m911 - m912)/1000*928 + p937/((m911 - m912)/1000)*925 \]

If (m911 > p913)  // checks to see if time has expired for move
  p1002 = 1
EndIf

//If (p908 > p955)  // safety variable into part
  // call16
  //EndIf
EndWhile

p1002 = 0

p963 = p906  // last position in x
p964 = p907  // last position in y
p965 = p908  // last position in z

return

close
A.5 Predicted Depth Linear Movement Program

open prog 12
clear
read(x,y,z,f)  // x = q124 (mm), y = q125 (mm), z = q126 (mm), f = q106 (mm/min)

// calculate the incremental movement distance to be used in the desired position

p960 = q124-p963
p961 = q125-p964
p962 = q126-p965

// calculation of move time for each cut

p912 = sqrt(((p960)*(p960))+((p961)*(p961))+((p962)*(p962)))  // total distance of groove length (mm)
p913 = p912/q106*60*1000  // total move time to complete groove (ms)

// loop to cut with model based compensation and depth prediction

// initialize timer variables specific to cutting program

m910 = m0  // initialize timer
m911 = 0  // initialize second timer
m912 = 0  // initialize third timer
p926 = 0  // initialize x integral error
p927 = 0  // initialize y integral error
p928 = 0  // initialize z integral error
p929 = 0  // initialize last x error (derivative)
p930 = 0  // initialize last y error (derivative)
p931 = 0  // initialize last z error (derivative)

While (p1000 = 1 And p1002 = 0)
  // start timer

  m912 = m911  // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608  // elapsed time since start of cutting (ms)

  // calculation of measured position in mm

  p906 = m900/(i108*32)/p900  // measured encoder x position (mm)
//p907 = m901/(i208*32)/p901  // measured encoder y position (mm)
p908 = m902/(i308*32)/p902  // measured encoder z position (mm)
// calculation of desired position

p909 = (p960/p913)*m911+p963 // desired x position (mm)
//p910 = (p961/p913)*m911+p964 // desired y position (mm)
p911 = (p962/p913)*m911+p965 // desired z position (mm)

// calculation of predicted depth of cut

p943 = m803*0.01 + 0.99*p943
p919 = p943*10/32768/50 // depth prediction in mm in z direction
// converted from dspace

i429 = 10000 // reset dspace maximum

// alignment position error

p920 = p909 - p906 // total x error (mm)
//p921 = p910 - p907 // total y error (mm)
p922 = p911 - p919 // total z error (mm)

// calculate derivative and integral error

p923 = p920 - p929 // derivative x error (mm)
//p924 = p921 - p930 // derivative y error (mm)
p925 = p922 - p931 // derivative z error (mm)

p929 = p920 // reset last x error (mm)
//p930 = p921 // reset last y error (mm)
p931 = p922 // reset last z error (mm)

If (p926 > 1) // watch for integral error x saturation
  p926 = 1 // integral error x (saturated)
Else
  p926 = p926 + p920 // integral error x (unsaturated)
EndIf

//If (p927 > 1) // watch for integral error y saturation
//  p927 = 1 // integral error y (saturated)
//Else
//  p927 = p927 + p921 // integral error y (unsaturated)
//EndIf

If (p928 > 1) // watch for integral error z saturation
  p928 = 1 // integral error z (saturated)
Else
    p928 = p928 + p922   // integral error z (unsaturated)
EndIf

// calculate control voltage to motor

i129 = p932*p920 + p938*(m911 - m912)/1000*p926 + p935/((m911 - m912)/1000)*p923
    //i229 = p933*p921 + p939*(m911 - m912)/1000*p927 + p936/((m911 - m912)/1000)*p924
i329 = p934*p922 + p940*(m911 - m912)/1000*p928 + p937/((m911 - m912)/1000)*p925
i429 = 0   // reset DSPACE maximum

If (m911 > p913)   // checks to see if time has expired for move
    p1002 = 1
EndIf

If (p908 > p955)               // safety variable of 150 um into part
    call16
EndIf
EndWhile

p1002 = 0

p963 = q124   // last position in x
p964 = q125   // last position in y
p965 = q126   // last position in z

return

close
A.6 PREDICTED DEFLECTION LINEAR MOVEMENT PROGRAM

open prog 13
clear
read(x,y,z,f)  // x = q124 (mm), y = q125 (mm), z = q126 (mm), f = q106 (mm/min)

// calculate the incremental movement distance to be used in the desired position
p960 = q124-p963
p961 = q125-p964
p962 = q126-p965

// calculation of move time for each cut
p912 = sqrt(((p960)*(p960))+((p961)*(p961))+((p962)*(p962)))  // total distance of groove length (mm)
p913 = p912/q106*60*1000  // total move time to complete groove (ms)

// loop to cut with stiffness compensation and deflection prediction

// initialize timer variables specific to cutting program
m910 = m0  // initialize timer
m911 = 0  // initialize second timer
m912 = 0  // initialize third timer
p926 = 0  // initialize x integral error
p927 = 0  // initialize y integral error
p928 = 0  // initialize z integral error
p929 = 0  // initialize last x error (derivative)
p930 = 0  // initialize last y error (derivative)
p931 = 0  // initialize last z error (derivative)

While (p1000 = 1 And p1002 = 0)
    // start timer
    m912 = m911  // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608  // elapsed time since start of cutting (ms)

    // calculation of measured position in mm
    p906 = m900/(i108*32)/p900  // measured encoder x position (mm)
p907 = m901/(i208*32)/p901  // measured encoder y position (mm)
p908 = m902/(i308*32)/p902  // measured encoder z position (mm)
// calculation of desired position
p909 = (p960/p913)*m911+p963 // desired x position (mm)
p910 = (p961/p913)*m911+p964 // desired y position (mm)
p911 = (p962/p913)*m911+p965 // desired z position (mm)

// calculation of predicted deflection
p943 = m803*0.01 + 0.99*p943
p919 = p943*0.000007803623 // deflection in mm in z direction converted
// from dspace
i429 = 10000 // reset dspace maximum

// alignment position error
p920 = p909 - p906 // total x error (mm) = desired - deflection - measured
p921 = p910 - p907 // total y error (mm)
p922 = p911 + p919 - p908 // total z error (mm) = desired + deflection - measured

// calculate derivative and integral error
p923 = p920 - p929 // derivative x error (mm)
p924 = p921 - p930 // derivative y error (mm)
p925 = p922 - p931 // derivative z error (mm)

p929 = p920 // reset last x error (mm)
p930 = p921 // reset last y error (mm)
p931 = p922 // reset last z error (mm)

//If (p926 > 1) // watch for integral error x saturation
//    p926 = 1 // integral error x (saturated)
//Else
p926 = p926 + p920 // integral error x (unsaturated)
//EndIf

//If (p927 > 1) // watch for integral error y saturation
//    p927 = 1 // integral error y (saturated)
//Else
p927 = p927 + p921 // integral error y (unsaturated)
//EndIf

//If (p928 > 1) // watch for integral error z saturation
// p928 = 1                      // integral error z (saturated)
// Else
p928 = p928 + p922            // integral error z (unsaturated)
// EndIf

// calculate control voltage to motor

i129 = p932*p920 + p938*(m911 - m912)/1000*p926 + p935/((m911 - m912)/1000)*p923
i229 = p933*p921 + p939*(m911 - m912)/1000*p927 + p936/((m911 - m912)/1000)*p924
i329 = p934*p922 + p940*(m911 - m912)/1000*p928 + p937/((m911 - m912)/1000)*p925
i429 = 0

If (m911 > p913)  // checks to see if time has expired for move
    p1002 = 1
EndIf

// If (p908 > p955)            // safety variable of 150 um into part
//     call16                     //EndIf
EndWhile

p1002 = 0

p963 = p906                      // last position in x
p964 = p907                      // last position in y
p965 = p908                      // last position in z

return
close
A.7 MANUAL LINEAR MOVEMENT PROGRAM

open prog 14
clear

// loop to hold ending program position based on last move
// position can be input by user on the fly

// initialize timer variables specific to cutting program

m910 = m0 // initialize timer
m911 = 0 // initialize second timer
m912 = 0 // initialize third timer
p926 = 0 // initialize x integral error
p927 = 0 // initialize y integral error
p928 = 0 // initialize z integral error
p929 = 0 // initialize last x error (derivative)
p930 = 0 // initialize last y error (derivative)
p931 = 0 // initialize last z error (derivative)

While (p1000 = 1 And p1002 = 0)
    // start timer

    m912 = m911 // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608 // elapsed time since start of cutting (ms)

    // calculation of measured position in mm

    p906 = m900/(i108*32)/p900 // measured encoder x position (mm)
p907 = m901/(i208*32)/p901 // measured encoder y position (mm)
p908 = m902/(i308*32)/p902 // measured encoder z position (mm)

    // calculation of desired position

    //p909 = p909 // desired x position (mm)
    //p910 = p910 // desired y position (mm)
    //p911 = p911 // desired z position (mm)

    // alignment position error

    p920 = p909 - p906 // total x error (mm)
p921 = p910 - p907 // total y error (mm)
p922 = p911 - p908 // total z error (mm)
// calculate derivative and integral error

p923 = p920 - p929  // derivative x error (mm)
p924 = p921 - p930  // derivative y error (mm)
p925 = p922 - p931  // derivative z error (mm)

p929 = p920        // reset last x error (mm)
p930 = p921        // reset last y error (mm)
p931 = p922        // reset last z error (mm)

If (p926 > 1)  // watch for integral error x saturation
    p926 = 1  // integral error x (saturated)
Else
    p926 = p926 + p920 // integral error x (unsaturated)
EndIf

If (p927 > 1)  // watch for integral error y saturation
    p927 = 1  // integral error y (saturated)
Else
    p927 = p927 + p921 // integral error y (unsaturated)
EndIf

If (p928 > 1)  // watch for integral error z saturation
    p928 = 1  // integral error z (saturated)
Else
    p928 = p928 + p922 // integral error z (unsaturated)
EndIf

// calculate control voltage to motor

i129 = p932*p920 + p938*(m911 - m912)/1000*p926 + p935/((m911 - m912)/1000)*p923
i229 = p933*p921 + p939*(m911 - m912)/1000*p927 + p936/((m911 - m912)/1000)*p924
i329 = p934*p922 + p940*(m911 - m912)/1000*p928 + p937/((m911 - m912)/1000)*p925
EndWhile

p1002 = 0

return

close
A.8 **OPEN-LOOP SINE WAVE MOVEMENT PROGRAM**

open prog 17
clear
read(a,w,p,d,x,y,f) // a = q101 (mm), w = q123 (Hz), d = q104 (mm), p = q116 (rad), x = q124 (mm), y = q125 (mm), f = q106 (mm/min)

// calculate the incremental movement distance to be used in the desired position

p960 = q124-p963
p961 = q125-p964

// calculation of move time for each cut

p912 = sqrt(((p960)*(p960))+((p961)*(p961))) // total distance of groove length (mm)
p913 = p912/q106*60*1000 // total move time to complete groove (ms)

// loop to cut with stiffness compensation and deflection prediction

// initialize timer variables specific to cutting program

m910 = m0 // initialize timer
m911 = 0 // initialize second timer
m912 = 0 // initialize third timer
p926 = 0 // initialize x integral error
p927 = 0 // initialize y integral error
p928 = 0 // initialize z integral error
p929 = 0 // initialize last x error (derivative)
p930 = 0 // initialize last y error (derivative)
p931 = 0 // initialize last z error (derivative)
i15 = 1 // make sure argument to sine function is in radians

While (p1000 = 1 And p1002 = 0)
  // start timer
  m912 = m911 // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608 // elapsed time since start of cutting (ms)

  // calculation of measured position in mm

  p906 = m900/(3072)/p900 // measured encoder x position (mm)
p907 = m901/(3072)/p901 // measured encoder y position (mm)
p908 = m902/(3072)/p902  // measured encoder z position (mm)

// calculation of desired position

p909 = (p960/p913)*m911+p963  // desired x position (mm)
p910 = (p961/p913)*m911+p964  // desired y position (mm)
p911 = (q101*sin((q123/1000*m911)+q116)+q104)+p965  // desired z position (mm)

// alignment position error

p920 = p909 - p906             // total x error (mm)
p921 = p910 - p907             // total y error (mm)
p922 = p911 - p908      // total z error (mm)

// calculate derivative and integral error

p923 = p920 - p929  // derivative x error (mm)
p924 = p921 - p930  // derivative y error (mm)
p925 = p922 - p931  // derivative z error (mm)

p929 = p920  // reset last x error (mm)
p930 = p921  // reset last y error (mm)
p931 = p922  // reset last z error (mm)

//If (p926 > 1)  // watch for integral error x saturation
//     p926 = 1  // integral error x (saturated)
//Else
//     p926 = p926 + p920 // integral error x (unsaturated)
//EndIf

//If (p927 > 1)  // watch for integral error y saturation
//     p927 = 1  // integral error y (saturated)
//Else
//     p927 = p927 + p921 // integral error y (unsaturated)
//EndIf

//If (p928 > 1)  // watch for integral error z saturation
//     p928 = 1  // integral error z (saturated)
//Else
//     p928 = p928 + p922 // integral error z (unsaturated)
//EndIf

// calculate control voltage to motor
i129 = p932*p920 + p938*(m911 - m912)/1000*p926 + p935/((m911 - m912)/1000)*p923 
\[ \text{If } (m911 > p913) \quad \text{// checks to see if time has expired for move} \]
\[ \text{p1002} = 1 \quad \text{EndIf} \]

\[ \text{If } (p908 > p955) \quad \text{// safety variable of 150 um into part} \]
\[ \text{call16} \quad \text{EndIf} \]
\[ \text{EndWhile} \]

p1002 = 0

p963 = p906 \quad \text{// last position in x}
p964 = p907 \quad \text{// last position in y}
p965 = p908 \quad \text{// last position in z}

return

close
A.9 CLOSED-LOOP SINE WAVE MOVEMENT PROGRAM

open prog 18
clear
read(a,w,p,d,x,y,f)  // a = q101 (mm), w = q123 (Hz), d = q104 (mm), p = q116 (rad), x = q124 (mm), y = q125 (mm), f = q106 (mm/min)

// calculate the incremental movement distance to be used in the desired position
p960 = q124-p963
p961 = q125-p964

// calculation of move time for each cut
p912 = sqrt(((p960)*(p960))+((p961)*(p961))) // total distance of groove length (mm)
p913 = p912/q106*60*1000 // total move time to complete groove (ms)

// loop to cut with stiffness compensation and deflection prediction

// initialize timer variables specific to cutting program
m910 = m0  // initialize timer
m911 = 0   // initialize second timer
m912 = 0   // initialize third timer
p926 = 0   // initialize x integral error
p927 = 0   // initialize y integral error
p928 = 0   // initialize z integral error
p929 = 0   // initialize last x error (derivative)
p930 = 0   // initialize last y error (derivative)
p931 = 0   // initialize last z error (derivative)
i15 = 1   // make sure argument to sine function is in radians

While (p1000 = 1 And p1002 = 0)
    // start timer
    m912 = m911 // capture time for delta time (ms)
m911 = (m0-m910)*i10/8388608 // elapsed time since start of cutting (ms)

// calculation of measured position in mm
p906 = m900/(i108*32)/p900 // measured encoder x position (mm)
p907 = m901/(i208*32)/p901  // measured encoder y position (mm)
\[ p908 = \frac{m902}{(i308 \times 32)} \] // measured encoder z position (mm)

// calculation of desired position
\[ p909 = \frac{(p960/p913) \times m911 + p963}{2} \] // desired x position (mm)
\[ p910 = \frac{(p961/p913) \times m911 + p964}{2} \] // desired y position (mm)
\[ p911 = ((q101 \times \sin(q123/1000 \times m911 + q116)) + q104) + p965 \] // desired z position (mm)

// calculation of deflection
\[ p943 = m803 \times 0.01 + 0.99 \times p943 \]
\[ p919 = p943 \times 0.000007259184 \] // deflection in mm in z direction converted
// from dspace
\[ i429 = 10000 \] // reset DSPACE maximum

// alignment position error
\[ p920 = p909 - p906 \] // total x error (mm)
\[ p921 = p910 - p907 \] // total y error (mm)
\[ p922 = p911 + p919 - p908 \] // total z error (mm) = desired + deflection - measured

// calculate derivative and integral error
\[ p923 = p920 - p929 \] // derivative x error (mm)
\[ p924 = p921 - p930 \] // derivative y error (mm)
\[ p925 = p922 - p931 \] // derivative z error (mm)
\[ p929 = p920 \] // reset last x error (mm)
\[ p930 = p921 \] // reset last y error (mm)
\[ p931 = p922 \] // reset last z error (mm)

//If (p926 > 1) // watch for integral error x saturation
//   p926 = 1 // integral error x (saturated)
//Else
//   p926 = p926 + p920 // integral error x (unsaturated)
//EndIf

//If (p927 > 1) // watch for integral error y saturation
//   p927 = 1 // integral error y (saturated)
//Else
//   p927 = p927 + p921 // integral error y (unsaturated)
//EndIf
// If (p928 > 1)  // watch for integral error z saturation
    // p928 = 1  // integral error z (saturated)
// Else
    p928 = p928 + p922 // integral error z (unsaturated)
// EndIf

// calculate control voltage to motor
  i129 = p932*p920 + p938*(m911 - m912)/1000*p926 + p935/((m911 - m912)/1000)*p923
  i229 = p933*p921 + p939*(m911 - m912)/1000*p927 + p936/((m911 - m912)/1000)*p924
  i329 = p934*p922 + p940*(m911 - m912)/1000*p928 + p937/((m911 - m912)/1000)*p925
  i429 = 0

  If (m911 > p913)  // checks to see if time has expired for move
  p1002 = 1
// EndIf

  // If (p908 > p955)  // safety variable of 150 um into part
    // call16
  // EndIf
EndWhile

p1002 = 0
p963 = p906  // last position in x
p964 = p907  // last position in y
p965 = p908  // last position in z

return
close
Due to the relatively low execution frequency of the PMAC motion program, a dSPACE 1102 external data acquisition system was used to acquire the cutting force, filter this force measurement, capture the peak force, and predict a depth/deflection. This dSPACE system was programmed using the Simulink diagram given below:
APPENDIX C – DSPACE NEW MAXIMUM FORCE CAPTURE

A new maximum force capture algorithm was built in dSPACE for a smoother maximum force to be used in closed-loop compensation. The simulink diagram for this new maximum force capture algorithm is given below. The algorithm is the same as the old maximum force capture algorithm (in Appendix B), except there are cascading maximum capture blocks (1) and (2)). Both maximum capture blocks are reset at the same period but (2) has a phase delay equal to 90% of the period so that (2) is reset right before (1) resets. Therefore, maximum block (2) grabs the maximum of (1) before (1) resets to zero and gives a much smoother capture of the maximum cutting force during machining.