D-472

Y-1920

COMPUTER SIMULATION STUDIES OF
PARABOLIC INTERPOLATION CIRCUITRY

C. M. Davenport

UNION
CARBIDE

OAK RIDGE Y-12 PLANT
OAK RIDGE, TENNESSEE

prepared for the U.S. ATOMIC ENERGY COMMISSION
under U.S. GOVERNMENT Contract W-7405 eng 26

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road, Springfield, Virginia 22151
Price: Printed Copy \$4.00; Microfiche \$1.45

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

COMPUTER SIMULATION STUDIES OF
PARABOLIC INTERPOLATION CIRCUITRY

C. M. Davenport

Oak Ridge Y-12 Plant

P.O. Box Y, Oak Ridge, Tennessee 37830

Date Issued - December 28, 1973

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Atomic Energy Commission, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

Prepared for the U.S. Atomic Energy Commission
Under U.S. Government Contract W-7405-eng-26

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED
leg

ABSTRACT

A computer simulation study of parabolic interpolation circuitry for machine tool controllers has been completed. Results showed that a tool center can be moved along any parabolic arc to within any realistic tolerance. Circuit design requirements were determined and input data required by the parabolic interpolators were defined. Simulation results showed that tool motion along a pronounced parabolic path would stay within a 1/2-microinch tolerance. An application of the parabolic interpolation toward reducing the large input data volumes required by conventional numerical controllers was achieved.

CONTENTS

SUMMARY	4
INTRODUCTION	5
Problem of Long Data Tapes	5
Causes of Large Data Volumes	5
COMPUTER SIMULATION STUDIES OF PARABOLIC INTERPOLATION CIRCUITRY	6
Parabolic Interpolation Circuitry	6
Definitions and Advantages	6
Parabolic Interpolation Circuitry	7
Modifications Necessary for High Precision	8
Simulation Study and Results	9
Computer Simulation Program	9
Simulation Example	11
Specifics of Parabolic Interpolator Operation	12
Extension to Cubic Interpolation	13
ACKNOWLEDGEMENTS	16
APPENDIX A	17
Simulation Program Flow Chart	17
Parabolic Interpolator Hardware Simulator (MACRO Subroutine DDCB3)	20
APPENDIX B	21
Simulation Example Printout	21
APPENDIX C	33
Summary of Input Data Calculations	33

SUMMARY

A large or complicated part, to be machined using a conventional NC controller and machine tool, requires a very large amount of machining data for input to the controller. Decreasing the tolerance on the finished part also makes the input data volume increase sharply. In some high-precision work, the punched tape containing the machining data has become so long that the normal error rate of the tape reader makes it almost a certainty that the tape will be misread at some point, causing the part which is being machined to be damaged.

The reason that so much input data are required is mainly because the part contour must be approximated by straight-line segments to within a tolerance of, typically, one half the machine (pulse) resolution. Each straight-line segment creates one block of data for the data tape. Straight-line approximation must be used because a standard controller can direct tool motion only along a line. Now, if the controller could direct tool motion along some curved path, say a parabolic arc, then a part contour could be approximated using curved segments. Thus, fewer segments would be required, resulting in fewer blocks of data and a much shorter data tape.

Electronic circuitry, which will allow a controller to direct tool motion along a parabolic path, has been completely simulated in a computer study. Results showed that any realistic tolerance on the parabolic motion of the tool center can be achieved. Hardware design requirements were worked out, and all characteristics were defined. Simulated tool motions, along pronounced parabolic paths to within a 1/2-microinch tolerance, were obtained. The study indicates that a high-precision controller with parabolic interpolation hardware would not be beyond present technology and would not be significantly more expensive to fabricate than a controller with linear interpolation hardware.

An experimental application is being prepared for a high-precision turning machine for use in cutting parabolic mirrors. In this project, a 1,600-foot data tape is expected to be reduced to less than 160 feet through use of parabolic interpolation. In addition, special high-speed circuitry will be used so that a 10-inch part can be finish cut to within a 1/4-microinch pulse resolution in a total time of 30 minutes or less.

INTRODUCTION

PROBLEM OF LONG DATA TAPES

Standard NC (numerical control) machines are programmed by means of punched tape. The tape is read a block at a time, as the machining operation is taking place, so that if the reader fails to read the tape properly the part being machined and the machine tool itself may be damaged. Error rates of one error per million bits transferred are typical for devices such as a tape reader. Since the standard tape format has 80 bits per inch, the error rate works out to about one error per 1,000 feet of tape. The result is that, if a part requires a tape of about 1,000 feet or more in length, the risk of machining failure is high.

Tape lengths for parts machined in the Oak Ridge Y-12 Plant^(a) have steadily increased over the years, until lengths of 1,000 feet or more are not unusual. Machining failures have occurred because of tape-reader errors, and ways are being sought to eliminate the problem. One technique which, aside from its other advantages, will solve the problem is direct digital numerical control (DDNC). Local core memory at the machine tool is loaded with the entire file of machining data from a local computer which has lines to the machine tool. The stored information may be checked for errors before machining begins. The data are then read from the core as machining proceeds. There is still a problem in the core memory approach, however. The increasingly large volumes of data will require large blocks of core, which is an expensive item. The core presently costs \$1,800 per 8K module, and up to 8 modules may be required in extreme cases for a total core cost of \$14,400. What is needed is some way to reduce the volume of data required by the controller.

CAUSES OF LARGE DATA VOLUMES

Large data volumes may be required by large or complicated parts, by parts specified to close tolerances, or by parts which require many specialized operations. In the Y-12 Plant, the longest tapes are usually a result of close-tolerance requirements. A typical situation is illustrated in Figure 1, where a part surface is described by a set of discrete points along the contour, typically spaced two degrees apart. These part-description points must be processed by the APT computer program, which outputs the machining tape. As depicted in Figure 2, the APT program fits some smooth mathematical curve through the data, then locates points between the design points such that straight-line motion between the intermediate points will not deviate by more than $1/2$ the machine pulse resolution, typically, from the fitted curve. Each straight-line segment creates a block of data for the machining tape. The troublesome thing here is that, for small pulse resolutions, the number of blocks required becomes very large. For example, the curved surface of a ten-inch-radius hemisphere requires about 130,000 straight-line segments if the approximation is to be good to 0.15 microinch. Whatever the part description, whether point-defined or a geometrical curve, the straight-line approximation must be used because present machine controllers can generate only straight-line motion (or linear interpolation). True radii may sometimes be the exception, because circular interpolation is available on some machines.

(a) Operated by the Union Carbide Corporation's Nuclear Division for the US Atomic Energy Commission.

Figure 1. POINT-DEFINED PART DESCRIPTION.

Figure 2. APT TREATMENT OF POINT-DEFINED PART DESCRIPTIONS.

COMPUTER SIMULATION STUDIES OF PARABOLIC INTERPOLATION CIRCUITRY

PARABOLIC INTERPOLATION

Definitions and Advantages

Tool motion along a parabolic path would be called "parabolic interpolation." The NC unit would need special electronic and digital circuitry which would produce pulse trains for the X-Y slides in such a way that a parabolic motion would be obtained. If such a feature were available, the part contour could be approximated by a series of parabolic arcs. As emphasized in Figure 3, one parabolic arc will cover a much greater length of the contour than a straight-line segment. For example, a ten-inch hemishell can be approximated by 90

Figure 3. APPROXIMATING A CONTOUR WITH PARABOLIC ARCS.

parabolic arcs to within a tolerance of 0.15 microinch. As stated previously, 130,000 straight-line segments would be necessary for the same tolerance—a reduction of 1,400 to 1 in data block requirements. The problem of correction for tool center offset will reduce the ratio somewhat, but a desirable target ratio is only 10 to 1. For example, if the 10-to-1 ratio is applied to a previously given example wherein a 1,600-foot tape was read into eight modules of core memory in a DDNC application, a 160-foot tape could now be read into one module of core. Tape reader error would be negligible, and core costs would be reduced from \$14,400 to \$1,800.

Parabolic Interpolation Circuitry

Figure 4 provides an outline of one commonly used digital circuit for linear interpolation. A review of its operation will help clarify the operation of the parabolic interpolator circuitry which is discussed in the paragraph that follows. Only one axis is shown, for brevity. The circuit acts as a digital integrator, performing the discrete analog of integrating a constant X velocity to get X position as a linear function of time. This action is depicted by the mathematical expressions on the right-hand side of the figure. The output is in the form of motor drive pulses which occur each time the integrated sum (or the remainder of the integrated sum) exceeds the register size. The clock rate controls the maximum feedrate. A separate countdown circuit keeps track of the position and shuts down the integrator when the desired endpoint is reached.

Figure 5 gives an expanded outline of the same circuit, which can be used for parabolic interpolation. In this circuit, X velocity does not remain constant but is constantly being modified by the output of another integrator. In fact, a constant X acceleration is being integrated twice to obtain the X position as a function of time, as depicted by the mathematical expressions on the right-hand side. X slide motion will be in the form:

$$X = X_1 + vt + \frac{a}{2}t^2$$

Figure 4. DIGITAL CIRCUITRY FOR LINEAR INTERPOLATION.

Figure 5. DIGITAL CIRCUITRY FOR PARABOLIC INTERPOLATION.

which is a parabola. Combined with a similar Y-axis motion, a parabolic arc in space can be executed. The motion is controlled by proper selection of the constants a , v , and X_1 for both axes.

Modifications Necessary For High Precision

There is one overriding principle which must be adhered to in order to obtain high precision, namely: that every quantity entered into the interpolator registers, or generated by the interpolator circuitry, must be carried to the same precision as is required of the output. For example, if an output precision of one part in one million is desired, then every input

constant must have six or more significant digits. The discrete approximation of the acceleration-versus-time curve cannot be "coarse", but must show high resolution. Lastly, the input constants a and v cannot be simply truncated, because this action amounts to subtracting some Δa and Δv respectively, from the constants. Any error in the input constants comes out as integration error in the output. In order to maintain the desired accuracy, a few places past the decimal point must be carried.

Components of a two-axis, high-precision parabolic interpolator are given in Figure 6. The first modification, which is necessary for high precision, is a longer register length. For tolerances in the microinch range, eight or nine decimal digits must be carried in the dominant integrands, which are usually the velocity integrands. Some additional length is also required to allow scaling of the subordinate integrands, usually the acceleration values. Required register length is also a direct function of total travel per block. A register length of 32 bits has proved to be adequate. A second modification is the addition of one more adder and one more register to allow for "floating" the acceleration overflow pointer. This change allows scaling of the acceleration integrand to obtain a "fine" discrete approximation to the acceleration-versus-time curve. The remaining modification is for the same purpose. A countdown register is added at each output to block out every other output pulse (divide the output by two). All the integrands are then (effectively) multiplied by two before loading so that all variables come out multiplied by two, and the outputs are rescaled by the countdown registers. The benefit in all this comes solely from the fact that the number of acceleration overflow pulses is doubled for a given travel, and so a finer discrete approximation to acceleration is obtained.

SIMULATION STUDY AND RESULTS

Computer Simulation Program

A computer program was written to simulate the high-precision interpolator circuitry shown in Figure 6. The program was written in FORTRAN and MACRO assembler-level language for the PDP-15/35 computer. The program is in two parts: (1) a FORTRAN main routine to handle input/output, to subdivide the curve to be simulated, and to calculate all data input to the interpolator; and (2) a MACRO subroutine which simulates the interpolator circuitry. Flow charts and program listings are given in Appendixes A and B, respectively.

The problem for which the computer program is written is illustrated in Figure 7. Two points and two corresponding slopes on a contour are given. These points might be point-defined data, or subdivisions of cubic splines fitted to point-defined data, or points on a true curve of some form between which a parabolic arc will give the desired accuracy. The program fits a parabolic equation in parametric form between the two points. The equation is:

$$X = A\theta^2 + B\theta + X_1, \text{ and}$$

$$Y = E\theta^2 + G\theta + Y_1, \quad 0 \leq \theta \leq 1,$$

where θ represents a time parameter, and the constants are given by:

$$\begin{aligned}\Delta X &= X_2 - X_1 \quad , \\ \Delta Y &= Y_2 - Y_1 \quad , \\ B &= 2(\Delta X \tan \alpha_2 - \Delta Y) / (\tan \alpha_2 - \tan \alpha_1) \quad , \\ A &= \Delta X - B \quad ,\end{aligned}$$

Figure 7. GEOMETRY FOR THE SIMULATION EXAMPLE. (Motion in any Block not to Exceed Maximum Travel or Feedrate Variation Limits)

$$G = B \tan \alpha_1, \text{ and}$$

$$E = \Delta Y - G.$$

The program allows the span, thus defined, to be subdivided into blocks such that motion in one block does not exceed some specified maximum travel or feedrate variation. (Feedrate variation is discussed in the topic that follows.) All the necessary data inputs to the interpolator are calculated and passed down to the MACRO simulator for each block in turn. The simulator goes through all the motions, as would the hardwired circuitry, and passes back the simulated position to the FORTRAN main routine. The main routine calculates a Y value from the parametric form which corresponds to the X value obtained from the simulation. A table is printed out as the simulation proceeds, showing add pulse number (or time), (X,Y) simulated, and Y calculated. In this way, a direct comparison may be made between the simulated path and the desired parabolic path.

The program is specific to the PDP-15 computer because of the MACRO language and because the MACRO simulator assumes that the computer has an 18-bit word length and the simulated interpolator has 32-bit registers. Double-precision floating-point arithmetic is used for all calculations in FORTRAN, yielding nine decimal digits of significance. Integer double-word arithmetic was written into the MACRO simulator to handle the 32-bit register size.

Simulation Example

A pronounced parabolic path was generated using the simulation program. Motion was from Point (1,1) to Point (0,0) with an initial slope of -115 degrees and a final slope of -180 degrees. The tolerance which was specified was 1/2 microinch. The maximum feedrate variation in one block for either axis was specified at 10 percent, resulting in the motion being generated in eight blocks. In order to keep the printout volume down to a practical

value, printout was set to occur on only every 10,000th output pulse on the X axis. At these points on X, the true value of Y was calculated, and a printout of X, Y (simulated), and Y (calculated) was made. Appendix B contains the complete computer input and output. Direct comparison of the simulated-versus-calculated positions shows that a 1/2-microinch tolerance was held.

Specifics of Parabolic Interpolator Operation

Only one intrinsic peculiarity showed up in the behavior of the parabolic interpolator, in the course of the simulation study, namely: that feedrate varies as a function of time on both axes, because:

$$X = A\theta^2 + B\theta + X_1$$

$$Y = E\theta^2 + G\theta + Y_1$$

$$\frac{dX}{d\theta} = 2A\theta + B = X \text{ axis feedrate, and}$$

$$\frac{dY}{d\theta} = 2E\theta + G = Y \text{ axis feedrate.}$$

Variation of the time parameter, θ , is set by the clock rate, which is fixed, so that the feedrate will vary within a block. A better visualization of the overall feedrate variation might be afforded by the following: If the radius of curvature of the parabolic path is designated by ρ , then the standard formula:

$$\rho = \frac{[(dx/d\theta)^2 + (dy/d\theta)^2]^{3/2}}{[(dx/d\theta)(d^2y/d\theta^2) - (d^2x/d\theta^2)(dy/d\theta)]}$$

from differential geometry may be solved to give:

$$\frac{ds}{d\theta} = (2[BE - AG])^{1/3} \rho^{1/3} = \text{const} \times \rho^{1/3}$$

where $ds/d\theta$ is the overall (surface) feedrate. The radius of curvature of a parabola is smallest at the vertex; hence, the tool motion will slow down as a vertex is approached. Although feedrate cannot be held constant within the block, the maximum feedrate and the allowable variation limits may be specified. Feedrate variation can be held down by limiting the travel in each block and resetting the desired feedrate between blocks. Feedrate variation does not appear to be a serious problem because, for example, a change by a factor of eight in the radius of curvature results in a change of only a factor of two in the feedrate.

The mathematics of the generation of the input data is relatively simple. Appendix C summarizes the results and provides a tabular list of the input data requirements. Preparing the data for proper action in the hardware was somewhat more complex. First of all, both

positive and negative values of acceleration and velocity had to be allowed, but all integrand quantities in the hardware must be positive because any integrator addition overflows must always signify that the result of the addition has exceeded the register size. Two's complement addition involving negative quantities may cause overflows which are meaningless and which are to be ignored as a byproduct of the method of addition. One otherwise troublesome quirk of machine tools (the fact that a slide reversal usually does not occur in the middle of a block because of backlash error) proved very fortuitous in the present problem. It means that no slide velocity will be required to change the sign in the middle of a block; thus, it is possible to load the absolute values of both the acceleration and velocity integrands and take care of their sign differences by having a signed (two's complement) quantity in the velocity increment registers. "Like" signs require a positive velocity increment, while "unlike" signs require a negative increment.

Secondly, proper shifting and scaling of the integrand quantities had to be achieved. Figure 8 shows typical integrand positions in the registers without acceleration scaling. All decimal points are aligned at a point which allows just enough open space on the left for any growth in magnitude of the velocity integrands. As many significant digits as are available past the decimal point are carried, up to the register capacity. Overflows are detected in the link (L) register. The signed acceleration overflow increment is carried in a separate register, as noted in Figure 6. It can be seen that, with small acceleration magnitudes and large velocity magnitudes (as depicted in Figure 8), the velocity integrators will overflow much more often than will be accelerator integrators. A relatively coarse approximation of the acceleration-versus-time curve will result. For high precision, the acceleration values may be scaled by shifting all the way to the left and moving the acceleration overflow pointer to the right by a compensating amount, as seen in Figure 9 (compare with Figure 8). Another effect allows further scaling on acceleration, namely: moving the overflow pointer to the right by one bit location is the same as assuming that the velocity integrand is doubled in size. Hence, if the pointer is moved and the velocity output is divided by two, the proper result (to greater precision) will be achieved. All shifting and scaling takes place in the software which generates the parabolic interpolator input data.

Lastly, although it is not peculiar to parabolic interpolation only, it should be mentioned that a parabolic tool center motion will not generate a parabolic part surface if the tool radius is not zero. The first solution is to use a machine tool which can set a zero tool radius as one axis of motion; a second solution is to approximate the part contour using the actual tool tip path. For a fixed tool radius, the part surface will be the sum of a parabola and a hyperbola—the latter opening in the same direction and having the same axis as the former. A third solution, less practical because of the hardware complications, is to build a set of integrators which can apply a correction to null out the hyperbolic deviation away from the true parabolic tool tip motion.

Extension To Cubic Interpolation

The parabolic interpolator layout of Figure 5 can be modified easily to obtain cubic interpolation, or tool motion along an arc of a cubic curve. Figure 10 gives the necessary modification. One additional integrator is required on each axis, connected in such a way that a constant third derivative of motion is integrated three times to obtain position as a function of time in the form:

Figure 8. POSITION OF THE INTEGRANDS WITHOUT SCALING THE ACCELERATION VALUES.

Figure 9. POSITION OF THE INTEGRANDS WITH SCALING OF THE ACCELERATION VALUES. (Requires $\times 1/2$ Divider on the Output)

ACKNOWLEDGEMENTS

The work reported herein is an outgrowth of prior work done by R. V. Miskell of the Y-12 Plant NC Engineering Department, who did a study on the data-reducing effects of cubic interpolation and sketched the basic hardware layouts. Results were published in the report: *Cubic Interpolation for Numerically Controlled Machines*, Y-1782; March 31, 1971.

APPENDIX A

SIMULATION PROGRAM FLOW CHART

PARABOLIC INTERPOLATOR HARDWARE SIMULATOR (Macro Subroutine DDCB3)

APPENDIX B

SIMULATION EXAMPLE PRINTOUT

A DKC1 -4/TT 6/LA 7/VP 10

\$GLOAD

LOADER V9A

>←PAR32,DDE\CB3,PLT,SET,FORT,LINE,NUVAL

PROGRAM PAR32:

SIMULATES A 2-AXIS PARABOLIC INTERPOLATOR.

INPUT X1,Y1,X2,Y2 ONE PER LINE, IN INCHES:

1.

1.

0.

0.

INPUT END SLOPES IN DEGREES, X1 END FIRST:

-115.

0.

INPUT MAX FEEDRATE IN IPM:

.5

INPUT MACHINE RESOLUTION
IN MICROINCHES PER PULSE:

.5

INPUT MAX TRAVEL ON EITHER AXIS, IN INCHES:

.25

ENTER # OF POINTS DESIRED IN THE COMPARISON TABLE. (0=NO TABLE):

0

INPUT MAX ALLOWABLE FEEDRATE CHANGE, IN PERCENT:

10.

HARDWARE SIMULATION WILL PRINT OUT ON EVERY NTH
OUTPUT PULSE ON THE X AXIS. INPUT N:

10000

START OVER?

NO

BEGIN HARDWARE SIMULATION.

END OF RUN.

STOP 000000

KM9-15 V5A

WHOLE ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES

1.00000000D+00

1.00000000D+00

0.00000000D-01

0.00000000D-01

A.B.E.G

-6.73846894D+04

-9.32815311D+05

1.00000000D+06

-2.00000000D+06

END SLOPES IN DEGREES. X1 END FIRST:

-1.15000000D+02

0.00000000D-01

ARC LENGTH = 1.48610481D+00 INCHES.

BEGIN HARDWARE SIMULATION.

BLOCK 1

SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES

1.00000000D+00

1.00000000D+00

8.77621000D-01

7.56900000D-01

A.B.E.G

-1.14144645D+03

-1.21237554D+05

1.68947742D+04

-2.58994774D+05

END SLOPES IN DEGREES. X1 END FIRST:

-1.15041963D+02

-1.18678921D+02

ARC LENGTH = 2.72211347D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 3.62948462D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	5.00000000D-01
1.00000000D-01	4.94838846D-01

2.00000000D-01	4.89694446D-01
3.00000000D-01	4.84567335D-01
4.00000000D-01	4.79458067D-01
5.00000000D-01	4.74367219D-01
6.00000000D-01	4.69285389D-01
7.00000000D-01	4.64243203D-01
8.00000000D-01	4.59211307D-01
9.00000000D-01	4.54200376D-01
1.00000000D+00	4.49211112D-01

THE INTEGRANDS ARE:

-4.56578580D+03
 -2.42475107D+05
 6.75790968D+04
 -5.19989548D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:
 5.19989548D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE *	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
0.00000000D-01	1.00000000D+00	1.00000000D+00	1.00000000D+00
4.32280000D+04	9.95000000D-01	9.89310500D-01	9.89310336D-01
8.64230000D+04	9.90000000D-01	9.78686500D-01	9.78686314D-01
1.29584000D+05	9.85000000D-01	9.68128000D-01	9.68127781D-01
1.72711000D+05	9.80000000D-01	9.57635000D-01	9.57634585D-01
2.15806000D+05	9.75000000D-01	9.47206500D-01	9.47206576D-01
2.58867000D+05	9.70000000D-01	9.36844000D-01	9.36843602D-01
3.01895000D+05	9.65000000D-01	9.26545500D-01	9.26545517D-01
3.44890000D+05	9.60000000D-01	9.16312500D-01	9.16312168D-01
3.87852000D+05	9.55000000D-01	9.06143500D-01	9.06143408D-01
4.30781000D+05	9.50000000D-01	8.96039500D-01	8.96039089D-01
4.73677000D+05	9.45000000D-01	8.85999500D-01	8.85999063D-01
5.16541000D+05	9.40000000D-01	8.76023500D-01	8.76023185D-01
5.59372000D+05	9.35000000D-01	8.66111500D-01	8.66111306D-01
6.02170000D+05	9.30000000D-01	8.56263500D-01	8.56263283D-01
6.44936000D+05	9.25000000D-01	8.46479500D-01	8.46478968D-01
6.87670000D+05	9.20000000D-01	8.36758500D-01	8.36758218D-01
7.30371000D+05	9.15000000D-01	8.27101000D-01	8.27100888D-01
7.73039000D+05	9.10000000D-01	8.17507500D-01	8.17506836D-01
8.15676000D+05	9.05000000D-01	8.07976500D-01	8.07975918D-01
8.58280000D+05	9.00000000D-01	7.98508500D-01	7.98507992D-01
9.00852000D+05	8.95000000D-01	7.89103500D-01	7.89102914D-01
9.43393000D+05	8.90000000D-01	7.79761000D-01	7.79760545D-01
9.85901000D+05	8.85000000D-01	7.70481000D-01	7.70480743D-01
1.02837700D+06	8.80000000D-01	7.61264000D-01	7.61263368D-01

BLOCK 2

SUB ARC CONSTANTS:

X1,Y1,X2,Y2, INCHES

8.77621000D-01
 7.56300000D-01
 7.62634500D-01
 5.62500000D-01

A.B.E.G

-9.69380127D+02
 -1.14017120D+05
 1.44016121D+04
 -2.08801612D+05

END SLOPES IN DEGREES. X1 END FIRST:

-1.18678921D+02
 -1.22831804D+02

ARC LENGTH = 2.25910424D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 3.01213898D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	5.00000000D-01
1.00000000D-01	4.94892964D-01
2.00000000D-01	4.89807835D-01
3.00000000D-01	4.84745301D-01
4.00000000D-01	4.79706079D-01
5.00000000D-01	4.74690911D-01
6.00000000D-01	4.69700566D-01
7.00000000D-01	4.64735846D-01
8.00000000D-01	4.59797580D-01
9.00000000D-01	4.54886629D-01
1.00000000D+00	4.50003888D-01

THE INTEGRANDS ARE:

-3.87752051D+03
 -2.28034240D+05
 5.76064484D+04
 -4.17603224D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:

4.17603224D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE #	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
1.04858200D+06	8.77620500D-01	7.56899500D-01	7.56898703D-01
1.09454800D+06	8.72620500D-01	7.47774000D-01	7.47773208D-01
1.14048400D+06	8.67620500D-01	7.38710500D-01	7.38709796D-01
1.18637800D+06	8.62620500D-01	7.29709000D-01	7.29708326D-01
1.23224200D+06	8.57620500D-01	7.20769000D-01	7.20768661D-01
1.27807100D+06	8.52620500D-01	7.11891000D-01	7.11890664D-01
1.32386700D+06	8.47620500D-01	7.03074500D-01	7.03074197D-01
1.36962900D+06	8.42620500D-01	6.94319500D-01	6.94319124D-01
1.41535700D+06	8.37620500D-01	6.85626000D-01	6.85625309D-01
1.46105200D+06	8.32620500D-01	6.76993000D-01	6.76992616D-01
1.50671300D+06	8.27620500D-01	6.68421500D-01	6.68420912D-01
1.55234000D+06	8.22620500D-01	6.59910500D-01	6.59910059D-01
1.59793400D+06	8.17620500D-01	6.51460500D-01	6.51459927D-01
1.64349500D+06	8.12620500D-01	6.43071000D-01	6.43070379D-01
1.68902200D+06	8.07620500D-01	6.34741500D-01	6.34741283D-01
1.73451600D+06	8.02620500D-01	6.26473000D-01	6.26472508D-01
1.77997700D+06	7.97620500D-01	6.18264500D-01	6.18263920D-01
1.82540500D+06	7.92620500D-01	6.10115500D-01	6.10115388D-01
1.87079900D+06	7.87620500D-01	6.02027000D-01	6.02026781D-01

1.91616100D+06	7.82620500D-01	5.93998500D-01	5.93997969D-01
1.96149000D+06	7.77620500D-01	5.86029000D-01	5.86028821D-01
2.00678600D+06	7.72620500D-01	5.78119500D-01	5.78119207D-01
2.05204900D+06	7.67620500D-01	5.70269500D-01	5.70268999D-01

BLOCK 3
SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES
7.62634500D-01
5.62500000D-01
6.45707500D-01
3.96900000D-01

A.B.E.G
-9.67627411D+02
-1.15959373D+05
1.44038093D+04
-1.80003809D+05

END SLOPES IN DEGREES. X1 END FIRST:
-1.0001004D+0E
-1.27987145D+02

ARC LENGTH = 2.02787880D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 2.70383839D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	5.00000000D-01
1.00000000D-01	4.94606009D-01
2.00000000D-01	4.89245461D-01
3.00000000D-01	4.83919465D-01
4.00000000D-01	4.78629177D-01
5.00000000D-01	4.73375792D-01
6.00000000D-01	4.68160554D-01
7.00000000D-01	4.62984751D-01
8.00000000D-01	4.57849721D-01
9.00000000D-01	4.52756851D-01
1.00000000D+00	4.47707580D-01

THE INTEGRANDS ARE:
-3.87050964D+03
-2.31918745D+05
5.76152371D+04
-3.60007619D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:
3.60007619D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE #	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
2.09715900D+06	7.62634000D-01	5.62499500D-01	5.62499023D-01
2.14235600D+06	7.57634000D-01	5.54767500D-01	5.54767080D-01
2.18752000D+06	7.52634000D-01	5.47094500D-01	5.47094159D-01
2.23265200D+06	7.47634000D-01	5.39480500D-01	5.39480130D-01
2.27775200D+06	7.42634000D-01	5.31925000D-01	5.31924868D-01

2.322819000D+06	7.376340000D-01	5.244285000D-01	5.24428246D-01
2.367854000D+06	7.326340000D-01	5.169905000D-01	5.16990138D-01
2.412857000D+06	7.276340000D-01	5.096110000D-01	5.09610420D-01
2.457828000D+06	7.226340000D-01	5.022895000D-01	5.02288964D-01
2.502767000D+06	7.176340000D-01	4.950260000D-01	4.95025647D-01
2.547675000D+06	7.126340000D-01	4.878205000D-01	4.87820344D-01
2.592550000D+06	7.076340000D-01	4.806730000D-01	4.80672933D-01
2.637393000D+06	7.026340000D-01	4.735835000D-01	4.73583289D-01
2.682205000D+06	6.976340000D-01	4.665515000D-01	4.66551289D-01
2.726985000D+06	6.926340000D-01	4.595770000D-01	4.59576812D-01
2.771734000D+06	6.876340000D-01	4.526600000D-01	4.52659735D-01
2.816451000D+06	6.826340000D-01	4.458000000D-01	4.45799936D-01
2.861137000D+06	6.776340000D-01	4.389975000D-01	4.38997295D-01
2.905791000D+06	6.726340000D-01	4.322515000D-01	4.32251690D-01
2.950414000D+06	6.676340000D-01	4.255630000D-01	4.25563002D-01
2.995005000D+06	6.626340000D-01	4.189310000D-01	4.18931110D-01
3.039566000D+06	6.576340000D-01	4.123560000D-01	4.12355895D-01
3.084095000D+06	6.526340000D-01	4.058375000D-01	4.05837238D-01
3.128594000D+06	6.476340000D-01	3.993750000D-01	3.99375021D-01

BLOCK 4
SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES
6.457075000D-01
3.969000000D-01
5.268395000D-01
2.601000000D-01

A.B.E.G
-9.708935000D+02
-1.17897106D+05
1.43993701D+04
-1.51199370D+05

END SLOPES IN DEGREES. X1 END FIRST:
-1.27987145D+02
-1.34436070D+02

ARC LENGTH = 1.81324094D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 2.41765458D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.000000000D-01	5.000000000D-01
1.000000000D-01	4.94414310D-01
2.000000000D-01	4.88880717D-01
3.000000000D-01	4.83400983D-01
4.000000000D-01	4.77976969D-01
5.000000000D-01	4.72610595D-01
6.000000000D-01	4.67303846D-01
7.000000000D-01	4.62058776D-01
8.000000000D-01	4.56877510D-01
9.000000000D-01	4.51762242D-01
1.000000000D+00	4.46715240D-01

THE INTEGRANDS ARE:

-3.88357404D+03
-2.35794213D+05
5.75974802D+04
-3.02398740D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:

3.02398740D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE #	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
3.14573600D+06	6.45707000D-01	3.96899500D-01	3.96899524D-01
3.10018000D+06	6.40707000D-01	3.90515500D-01	3.90515302D-01
3.23461400D+06	6.35707000D-01	3.84187000D-01	3.84187238D-01
3.27900600D+06	6.30707000D-01	3.77915500D-01	3.77915215D-01
3.32336800D+06	6.25707000D-01	3.71609000D-01	3.71699115D-01
3.36769900D+06	6.20707000D-01	3.65539000D-01	3.65538823D-01
3.41199900D+06	6.15707000D-01	3.59434000D-01	3.59434222D-01
3.45626800D+06	6.10707000D-01	3.53385500D-01	3.53385197D-01
3.50050700D+06	6.05707000D-01	3.47391500D-01	3.47391633D-01
3.54471500D+06	6.00707000D-01	3.41453500D-01	3.41453414D-01
3.58889300D+06	5.95707000D-01	3.35570500D-01	3.35570427D-01
3.63304100D+06	5.90707000D-01	3.29742500D-01	3.29742557D-01
3.67715800D+06	5.85707000D-01	3.23969500D-01	3.23969690D-01
3.72124400D+06	5.80707000D-01	3.18252000D-01	3.18251714D-01
3.76530100D+06	5.75707000D-01	3.12588500D-01	3.12588515D-01
3.80932700D+06	5.70707000D-01	3.06980000D-01	3.06979981D-01
3.85332300D+06	5.65707000D-01	3.01426000D-01	3.01426000D-01
3.89728900D+06	5.60707000D-01	2.95926500D-01	2.95926461D-01
3.94122500D+06	5.55707000D-01	2.90481500D-01	2.90481251D-01
3.98513200D+06	5.50707000D-01	2.85090500D-01	2.85090260D-01
4.02900800D+06	5.45707000D-01	2.79753500D-01	2.79753378D-01
4.07285500D+06	5.40707000D-01	2.74470500D-01	2.74470494D-01
4.11667100D+06	5.35707000D-01	2.69241500D-01	2.69241497D-01
4.16045900D+06	5.30707000D-01	2.64066500D-01	2.64066281D-01

BLOCK 5.

SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES

5.26839500D-01
2.60100000D-01
4.06031000D-01
1.52100000D-01

A.B.E.G

-9.69516182D+02
-1.19836984D+05
1.44007222D+04
-1.22400722D+05

END SLOPES IN DEGREES. X1 END FIRST:

-1.34436070D+02
-1.42495878D+02

ARC LENGTH = 1.62178735D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 2.16238313D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	5.00000000D-01
1.00000000D-01	4.94428891D-01
2.00000000D-01	4.88939511D-01
3.00000000D-01	4.83534643D-01
4.00000000D-01	4.78217152D-01
5.00000000D-01	4.72989986D-01
6.00000000D-01	4.67856173D-01
7.00000000D-01	4.62818818D-01
8.00000000D-01	4.57881105D-01
9.00000000D-01	4.53046292D-01
1.00000000D+00	4.48317709D-01

THE INTEGRANDS ARE:

-3.87806473D+03
 -2.39677968D+05
 5.76028888D+04
 -2.44801444D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:

2.44801444D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE #	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
4.19431100D+06	5.26839000D-01	2.60099500D-01	2.60099557D-01
4.21617800D+06	5.21839000D-01	2.55019500D-01	2.55019454D-01
4.23803100D+06	5.16839000D-01	2.49993000D-01	2.49992828D-01
4.25986800D+06	5.11839000D-01	2.45019500D-01	2.45019571D-01
4.28169200D+06	5.06839000D-01	2.40099500D-01	2.40099575D-01
4.30350000D+06	5.01839000D-01	2.35233000D-01	2.35232734D-01
4.32529400D+06	4.96839000D-01	2.30419000D-01	2.30418939D-01
4.34707300D+06	4.91839000D-01	2.25658000D-01	2.25658086D-01
4.36883800D+06	4.86839000D-01	2.20950000D-01	2.20950066D-01
4.39058800D+06	4.81839000D-01	2.16295000D-01	2.16294774D-01
4.41232400D+06	4.76839000D-01	2.11692000D-01	2.11692106D-01
4.43404500D+06	4.71839000D-01	2.07142000D-01	2.07141955D-01
4.45575200D+06	4.66839000D-01	2.02644500D-01	2.02644216D-01
4.47744500D+06	4.61839000D-01	1.98198500D-01	1.98198786D-01
4.49912300D+06	4.56839000D-01	1.93805500D-01	1.93805559D-01
4.52078600D+06	4.51839000D-01	1.89464500D-01	1.89464433D-01
4.54243600D+06	4.46839000D-01	1.85175000D-01	1.85175303D-01
4.56407100D+06	4.41839000D-01	1.80938000D-01	1.80938067D-01
4.58569100D+06	4.36839000D-01	1.76752500D-01	1.76752622D-01
4.60729800D+06	4.31839000D-01	1.72619000D-01	1.72618864D-01
4.62889000D+06	4.26839000D-01	1.68536500D-01	1.68536694D-01
4.65046800D+06	4.21839000D-01	1.64506000D-01	1.64506007D-01
4.67203200D+06	4.16839000D-01	1.60526500D-01	1.60526703D-01
4.69358200D+06	4.11839000D-01	1.56598500D-01	1.56598682D-01
4.71511800D+06	4.06839000D-01	1.52721500D-01	1.52721841D-01

BLOCK 6

SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES

4.06031000D-01
 1.52100000D-01
 2.62634500D-01
 6.25000000D-02

A. B. E. G

-1.32210979D+03
 -1.42074390D+05
 1.95991866D+04
 -1.09199187D+05

END SLOPES IN DEGREES. X1 END FIRST:

-1.42495878D+02
 -1.54228774D+02

ARC LENGTH = 1.69383709D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 2.25844946D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	5.00000000D-01
1.00000000D-01	4.94003866D-01
2.00000000D-01	4.88180283D-01
3.00000000D-01	4.82535499D-01
4.00000000D-01	4.77075080D-01
5.00000000D-01	4.71807793D-01
6.00000000D-01	4.66737785D-01
7.00000000D-01	4.61872359D-01
8.00000000D-01	4.57218046D-01
9.00000000D-01	4.52781357D-01
1.00000000D+00	4.48568748D-01

THE INTEGRANDS ARE:

-5.28843915D+03
 -2.84148780D+05
 7.83967466D+04
 -2.18398373D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:

2.89437220D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE #	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
4.71860100D+06	4.06030500D-01	1.52100000D-01	1.52099756D-01
4.75549100D+06	4.01030500D-01	1.48262500D-01	1.48282246D-01
4.79235800D+06	3.96030500D-01	1.44515500D-01	1.44515701D-01
4.82920000D+06	3.91030500D-01	1.40800000D-01	1.40800020D-01
4.86601800D+06	3.86030500D-01	1.37135000D-01	1.37135104D-01
4.90281200D+06	3.81030500D-01	1.33521000D-01	1.33520854D-01
4.93958200D+06	3.76030500D-01	1.29957500D-01	1.29957172D-01
4.97632800D+06	3.71030500D-01	1.26444000D-01	1.26443958D-01
5.01305100D+06	3.66030500D-01	1.22981000D-01	1.22981116D-01
5.04974900D+06	3.61030500D-01	1.19508500D-01	1.19568546D-01
5.08642400D+06	3.56030500D-01	1.16206500D-01	1.16206151D-01
5.12307500D+06	3.51030500D-01	1.12894000D-01	1.12893834D-01
5.15970300D+06	3.46030500D-01	1.09631500D-01	1.09631499D-01
5.19630700D+06	3.41030500D-01	1.06419000D-01	1.06419049D-01
5.23288700D+06	3.36030500D-01	1.03256500D-01	1.03256387D-01
5.26944400D+06	3.31030500D-01	1.00143500D-01	1.00143418D-01
5.30597700D+06	3.26030500D-01	9.70800000D-02	9.70800457D-02
5.34248700D+06	3.21030500D-01	9.40665000D-02	9.40661749D-02
5.37897300D+06	3.16030500D-01	9.11020000D-02	9.11017106D-02

5.41543700D+06	3.11030500D-01	8.81865000D-02	8.81865583D-02
5.45187600D+06	3.06030500D-01	8.53210000D-02	8.53206237D-02
5.48829300D+06	3.01030500D-01	8.25040000D-02	8.25038121D-02
5.52468700D+06	2.96030500D-01	7.97360000D-02	7.97360301D-02
5.56105700D+06	2.91030500D-01	7.70175000D-02	7.70171839D-02
5.59740400D+06	2.86030500D-01	7.43475000D-02	7.43471809D-02
5.63372800D+06	2.81030500D-01	7.17260000D-02	7.17259275D-02
5.67002900D+06	2.76030500D-01	6.91535000D-02	6.91533312D-02
5.70630700D+06	2.71030500D-01	6.66295000D-02	6.66292999D-02
5.74256200D+06	2.66030500D-01	6.41540000D-02	6.41537416D-02

BLOCK 7
SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES
2.62634500D-01
6.25000000D-02
1.06670000D-02
1.00000000D-04

A.B.E.G
-3.88133559D+03
-2.48086164D+05
5.76000000D+04
-1.20000000D+05

END SLOPES IN DEGREES. X1 END FIRST:
-1.54228774D+02
-1.78967159D+02

ARC LENGTH = 2.61641017D-01 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 3.48854689D+01 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	5.00000000D-01
1.00000000D-01	4.92550091D-01
2.00000000D-01	4.85889760D-01
3.00000000D-01	4.80051871D-01
4.00000000D-01	4.75066747D-01
5.00000000D-01	4.70061467D-01
6.00000000D-01	4.67759197D-01
7.00000000D-01	4.65478575D-01
8.00000000D-01	4.64133197D-01
9.00000000D-01	4.63731173D-01
1.00000000D+00	4.64274984D-01

THE INTEGRANDS ARE:
-1.55253423D+04
-4.96172329D+05
2.30400000D+05
-2.40000000D+05

THE LARGEST INTEGRAND MAGNITUDE WILL BE:
5.11697671D+05

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE *	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
5.767177000D+06	2.626340000D-01	6.250000000D-02	6.24996957D-02
5.788303000D+06	2.576340000D-01	6.010550000D-02	6.01053314D-02
5.809417000D+06	2.526340000D-01	5.775950000D-02	5.77591949D-02
5.830517000D+06	2.476340000D-01	5.546150000D-02	5.54611948D-02
5.851603000D+06	2.426340000D-01	5.321150000D-02	5.32112408D-02
5.872677000D+06	2.376340000D-01	5.100950000D-02	5.10092423D-02
5.893737000D+06	2.326340000D-01	4.885550000D-02	4.88551093D-02
5.914785000D+06	2.276340000D-01	4.674900000D-02	4.67487518D-02
5.935819000D+06	2.226340000D-01	4.469050000D-02	4.46900804D-02
5.956840000D+06	2.176340000D-01	4.267900000D-02	4.26790056D-02
5.977847000D+06	2.126340000D-01	4.071550000D-02	4.07154385D-02
5.998842000D+06	2.076340000D-01	3.879950000D-02	3.87992905D-02
6.019824000D+06	2.026340000D-01	3.693050000D-02	3.69304728D-02
6.040793000D+06	1.976340000D-01	3.510900000D-02	3.51088978D-02
6.061748000D+06	1.926340000D-01	3.333450000D-02	3.33344769D-02
6.082691000D+06	1.876340000D-01	3.160750000D-02	3.16071229D-02
6.103621000D+06	1.826340000D-01	2.992700000D-02	2.99267482D-02
6.124537000D+06	1.776340000D-01	2.829350000D-02	2.82932661D-02
6.145441000D+06	1.726340000D-01	2.670700000D-02	2.67065893D-02
6.166332000D+06	1.676340000D-01	2.516700000D-02	2.51666312D-02
6.187210000D+06	1.626340000D-01	2.367350000D-02	2.36733062D-02
6.208076000D+06	1.576340000D-01	2.222650000D-02	2.22265277D-02
6.228928000D+06	1.526340000D-01	2.082650000D-02	2.08262097D-02
6.249768000D+06	1.476340000D-01	1.947250000D-02	1.94722673D-02
6.270595000D+06	1.426340000D-01	1.816500000D-02	1.81646151D-02
6.291409000D+06	1.376340000D-01	1.690350000D-02	1.69031680D-02
6.312210000D+06	1.326340000D-01	1.568800000D-02	1.56878412D-02
6.332999000D+06	1.276340000D-01	1.451850000D-02	1.45185506D-02
6.353775000D+06	1.226340000D-01	1.339550000D-02	1.33952116D-02
6.374538000D+06	1.176340000D-01	1.231800000D-02	1.23177404D-02
6.395289000D+06	1.126340000D-01	1.128600000D-02	1.12860534D-02
6.416027000D+06	1.076340000D-01	1.030050000D-02	1.03000669D-02
6.436753000D+06	1.026340000D-01	0.936000000D-03	0.93596980D-03
6.457466000D+06	9.763400000D-02	8.465000000D-03	8.46486377D-03
6.478167000D+06	9.263400000D-02	7.615500000D-03	7.61548144D-03
6.498855000D+06	8.763400000D-02	6.811500000D-03	6.81146851D-03
6.519530000D+06	8.263400000D-02	6.053000000D-03	6.05274292D-03
6.540193000D+06	7.763400000D-02	5.339500000D-03	5.33922253D-03
6.560844000D+06	7.263400000D-02	4.671000000D-03	4.67082587D-03
6.581482000D+06	6.763400000D-02	4.047500000D-03	4.04747156D-03
6.602108000D+06	6.263400000D-02	3.469000000D-03	3.46907812D-03
6.622721000D+06	5.763400000D-02	2.935500000D-03	2.93556476D-03
6.643322000D+06	5.263400000D-02	2.447000000D-03	2.44685077D-03
6.663911000D+06	4.763400000D-02	2.003000000D-03	2.00285565D-03
6.684487000D+06	4.263400000D-02	1.603500000D-03	1.60349927D-03
6.705051000D+06	3.763400000D-02	1.249000000D-03	1.24870166D-03
6.725603000D+06	3.263400000D-02	9.385000000D-04	9.38382812D-04
6.746143000D+06	2.763400000D-02	6.725000000D-04	6.72463440D-04
6.766670000D+06	2.263400000D-02	4.510000000D-04	4.50864136D-04
6.787185000D+06	1.763400000D-02	2.735000000D-04	2.73506042D-04
6.807688000D+06	1.263400000D-02	1.405000000D-04	1.40310242D-04

BLOCK 8

SUB ARC CONSTANTS:

X1.Y1.X2.Y2. INCHES

1.066570000D-02

1.000000000D-04

0.000000000D-01

0.000000000D-01

A. B. E. C

-6.63004398D+00
 -1.06603700D+04
 1.00000000D+02
 -2.00000000D+02

END SLOPES IN DEGREES. X1 END FIRST:

-1.78967159D+02
 1.80041963D+02

ARC LENGTH = 1.06676251D-02 INCHES.

APPROXIMATE TIME OF TRAVERSAL = 1.42235001D+00 SECONDS.

NORMALIZED TIME (0-1)	FEEDRATE (IPM)
0.00000000D-01	4.99466718D-01
1.00000000D-01	4.99512128D-01
2.00000000D-01	4.99559300D-01
3.00000000D-01	4.99608233D-01
4.00000000D-01	4.99658925D-01
5.00000000D-01	4.99711377D-01
6.00000000D-01	4.99765586D-01
7.00000000D-01	4.99821554D-01
8.00000000D-01	4.99879280D-01
9.00000000D-01	4.99938762D-01
1.00000000D+00	5.00000000D-01

THE INTEGRANDS ARE:

-2.65201759D+01
 -2.13207399D+04
 4.00000000D+02
 -4.00000000D+02

THE LARGEST INTEGRAND MAGNITUDE WILL BE:

2.13472601D+04

HARDWARE SIMULATION FOR THIS BLOCK:

ADD PULSE #	X (INCHES)	Y (INCHES)	Y CALC (INCHES)
6.81575500D+06	1.06665000D-02	1.00000000D-04	0.99885864D-05
6.84648400D+06	5.66650000D-03	2.85000000D-05	2.50742658D+02
6.87719500D+06	6.66500000D-04	5.00000000D-07	2.50891088D+02
6.88128800D+06	0.00000000D-01	0.00000000D-01	0.00000000D-01

END OF RUN.

APPROXIMATE TIME OF TRAVERSAL FOR WHOLE ARC = 3.30245518D+00 MINUTES
 TOTAL NUMBER OF BLOCKS = 8

APPENDIX C

SUMMARY OF INPUT DATA CALCULATIONS

Parabolic Arc Parameters

$(X_1, Y_1), (X_2, Y_2)$ = end points of arc, rounded to nearest multiple of the pulse resolution,

a_1, a_2 = end slopes of the arc, and

R = machine pulse resolution.

Results of Calculations

$$\Delta X = X_2 - X_1,$$

$$\Delta Y = Y_2 - Y_1,$$

$$B = 2 (\Delta X \tan a_2 - \Delta Y) / (\tan a_2 - \tan a_1),$$

$$A = \Delta X - B,$$

$$G = B \tan a_1, \text{ and}$$

$$E = \Delta Y - G \text{ (only three of the four coefficients are independent).}$$

The equation of motion is:

$$X = A\theta^2 + B\theta + X_1, \text{ and}$$

$$Y = E\theta^2 + G\theta + Y_1, 0 \leq \theta \leq 1.$$

The interpolator integrands are:

$$\text{FINT (1)} = 2A/R = X \text{ acceleration,}$$

$$\text{FINT (2)} = B/R = X \text{ velocity,}$$

$$\text{FINT (3)} = 2E/R = Y \text{ acceleration, and}$$

$$\text{FINT (4)} = G/R = Y \text{ velocity.}$$

The largest magnitude which any integrand will reach in the current block of travel is:

$$\begin{aligned} \text{FMAX} = \max [& (\text{FINT (1)}), (\text{FINT (2)}), (\text{FINT (3)}), (\text{FINT (4)}), \\ & (\text{FINT (1)} + \text{FINT (2)}), (\text{FINT (3)} + \text{FINT (4)})]. \end{aligned}$$

Figure C-1 lists, in tabular form, all input data values along with the proper scaling-shift values. It is assumed that shifting begins from the standard floating-point format, in which the left-most bit in the register is a sign bit, immediately followed by the mantissa of the quantity to be shifted. The sign bit must be set to zero before shifting.

Data Word Number	Register Name	Quantity	Shift Required Before Loading, In Bits (Note 1) (+ = right, - = left)
1	X acceleration integrand	$ 2A/R $	-1
2	X velocity integrand	$ B/R $	$e_{\max} - e_1 - 1$
3	Y acceleration integrand	$ 2E/R $	-1
4	Y velocity integrand	$ G/R $	$e_{\max} - e_3 - 1$
5	X acceleration overflow increment	± 1 (Note 2)	$2e_{\max} - e_0 - e_2 + 1$
6	Y acceleration overflow increment	± 1 (Note 2)	$2e_{\max} - e_0 - e_4 + 1$
7	X output sign	+ or -	0
8	Y output sign	+ or -	0
9	X distance countdown	$-\Delta X/R$	0
10	Y distance countdown	$-\Delta Y/R$	0
11	Clock rate	(Note 3)	0

Note 1: All integrand sign bits are to be cleared before integrands are shifted.

Note 2: "Like" signs on acceleration and velocity = +1; unlike signs = -1.

Note 3: $2 \times \text{Desired Feedrate} \times \text{Max Integrand}$

$$\left. \frac{ds}{d\theta} \right|_{\max}$$

Figure C-1. SUMMARY OF THE INPUT DATA REQUIREMENTS FOR PARABOLIC INTERPOLATION.

DISTRIBUTION**Atomic Energy Commission - ALO**

Vespe, V. C.

Atomic Energy Commission - ORO

Hickman, H. D.

Zachry, D. S., Jr

Lawrence Livermore Laboratories

Eckard, R.

Jenson, J. A.

Spies, R. J.

Strange, F.

Technical Information Division

Los Alamos Scientific Laboratory

Stack, F. E.

Oak Ridge Gaseous Diffusion Plant

McLaren, R. A.

Wilcox, W. J., Jr

Winkel, R. A.

Oak Ridge Y-12 Plant

Alvey, H. E.

Bearden, L. E.

Bernander, N. K.

Bowers, G. L.

Burditt, R. B.

Burkhart, L. E.

Butturini, W. G.

Davenport, C. M. (5)

Denny, A.

Fouk, D. L.

Gritzner, V. B.

Hensley, C. E.

Jones, F. W.

Kahl, K. G.

Keith, Alvin

Kite, H. T.

Lundin, M. I.

Miskell, R. V.

Noey, J. L.

Oliphant, G. W.

Perry, A. E.

Phillips, L. R.

Smith, D. N.

Smith, H. F., Jr

Smith, R. D.

Stephens, A. E.

Stoner, H. H.

Thompson, C. H.

Tilson, F. V.

Tunnell, H. A.

Weathersby, W. E.

Webber, T. R.

Whitson, W. K.

Williams, T. L.

Wright, C. C.

Yaggi, W. J.

Y-12 Central Files (5)

Y-12 Central Files (master copy)

Y-12 Central Files (route)

Y-12 Central Files (Y-12HC)

Paducah Gaseous Diffusion Plant

Millican, R.

Sandia - Albuquerque

Gardner, W. A.

Mail Service Section

Sandia - Livermore

Davies, L. E.

Library

In addition, this report is distributed in accordance with the category UC-32, **Mathematics and Computers**, as given in the *USAEC Standard Distribution Lists for Unclassified Scientific and Technical Reports*, TID-4500.