



# NATIONAL RADIO ASTRONOMY OBSERVATORY

ELECTRONICS DIVISION TECHNICAL NOTE NO. 116

TITLE: Lathe Table APPLESOFT Program

AUTHOR(S): Roger D. Norrod

DATE: March 16, 1983

## DISTRIBUTION:

### GB

GB Library  
R. Fisher  
R. Lacasse  
W. Brundage  
R. Weimer  
D. Schiebel  
C. Moore  
C. Brockway  
J. Coe  
G. Behrens  
R. Mauzy  
R. Norrod  
F. Crews

### CV

CV Library  
H. Hvatum  
M. Balister  
S. Weinreb  
C. Burgess

### TU

Library Downtown  
Library Mountain  
J. Payne

### VLA

VLA Library  
P. Napier  
J. Campbell



LATHE TABLE APPLESOFT PROGRAM

Roger D. Norrod

March 15, 1983

The following is a description of an Applesoft BASIC program which is intended to produce a table of data describing a part to be machined on a lathe equipped with a numerical readout. The program is set for the conventions followed by the Green Bank lathe readout; that is, the carriage (Z) readout increases positive toward the tailstock and the crossfeed (X) readout increases positive toward the operator.

This program description consists of two sections. The first is intended for someone who merely wants to use the program to produce a table. The second section is intended for a programmer who wishes to describe a new part or modify the existing program.

1.0 Using the Program

With the APPLE off, insert the proper disk into the disk drive and close the door. Turn the APPLE power on. The video display will soon display a list of programs that are on the disk and then a blinking block cursor. Type RUN LATHE TABLE and Return or the disk may automatically load and run the proper program. A menu is then displayed. To begin the process of producing a table, the user selects "T".

The program then asks the user several questions about how the piece is mounted in the lathe and how the table is to be organized. Most questions have only two acceptable answers and the choices are highlighted on the screen. In addition, most questions have default answers that are displayed between slashes. If the user presses only Return in response to a question, then the default answer is used. An entry of "QQ" will cause the program to return to the menu. The questions asked are:

1. If the machinist will start at reference point 1 or 2. A drawing of the part being machined should indicate two reference points on the Z axis, normally at the endpoints of the described curve. The Z axis will be zeroed at the point selected here, so the user will usually enter the point where the cut begins.
2. If the carriage will move toward the headstock or tailstock during the cut.
3. If the table should have a constant X step size or a constant Z step.
4. The tool radius.
5. The size of the step. The program will only accept inputs here greater than zero.
6. The amount of any oversize, such as would be required for roughing cuts, etc. No checks are made on this input, so the user should be careful to enter the proper sign for this offset. Note that the oversize is a diameter measurement, not a radius.

7. The diameter at which the X axis is zeroed. An entry of zero here implies that the X axis is zeroed at the lathe axis of rotation.

The program then displays the ranges of Z and X (including the effect of the entry in 7 above but not including the oversize or tool radius offsets), and asks the user if he wants the total table or a portion only. If the partial option is selected, the program asks for the starting and ending values. Selecting the starting and ending entries may require some thought since the entries do not include oversize or tool radius offsets, while displayed table values do include the offsets.

The program then asks the user to wait and calculates the data points. This may take several minutes, depending on the number of points requested. Because of memory size restrictions, the computer can only hold 500 points at one time. If the user has requested more than 500, the program will calculate the first 500 and alert the user that the memory is full.

Upon completion of the calculations, the display menu is presented. Options here are:

- (P) To print the table. The printout may be terminated at any time by pressing Q. Upon completion of printing, the program returns to the display menu.
- (D) To display the table. The data is displayed on the video screen and the I/O paddle buttons may be used to scroll the screen. Upon pressing Q at the keyboard, the program returns to the display menu.
- (R) To resume calculations. If memory is full, the program will resume calculations and then return to the display menu. If calculations are complete, the program will return to the main menu, ready to make a new table.
- (E) To end the program.

The program saves on disk the entries made by the user so that these become new default values the next time the program is run. Figure 1 shows how the tool radius should be accounted for when zeroing the Z and X axis.

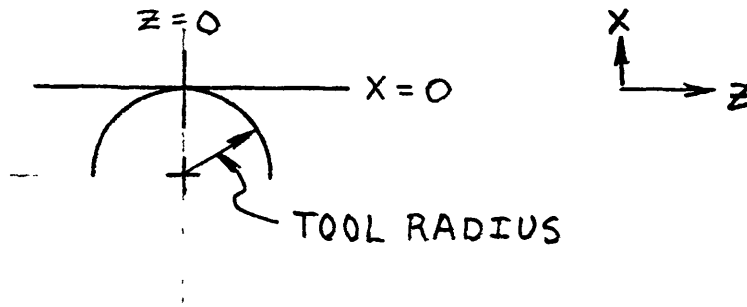


Figure 1: Zeroing the axes.

## 2.0 Programming Notes

The programmer should first read section 1.0. The program is organized as a group of subroutines. Remarks have been used extensively to make the program self-documenting. A program listing and a variable cross reference index are attached for an example program that has been used to machine a L-band OMT mandrel. For illustration, a drawing of the mandrel is also attached.

The axis along the bed of the lathe is the Z axis; the crossfeed moves along the X axis. Variables Z and X are the positions displayed to the user and agree in magnitude and sign with the digital lathe readout. Program variables RZ and RX are variables in equations that the programmer uses to describe the part. The program converts the variables RZ and RX to Z and X, automatically assigning the correct sign and magnitude.

In order to describe a new part, the programmer need only modify four sub-routines:

- 500-599 This routine calculates RX given a value of RZ, where RX is the radius of the part and RZ is the position along the length of the part.
- 600-699 This routine calculates the slope at a point RZ, RX and assigns it to the variable SL.
- 700-899 This routine calculates RZ given a value of RX. This routine is used if the user requests constant steps in the X direction.
- 900-999 In this routine, the user defines certain variables required by the program. These are:
  - UNIT%        Ø for units of inches  
              1 for millimeters
  - TI\$           A string used as a title for the program.
  - U1\$,U2\$       Optional strings printed as headers for the data table.
  - Z1,Z2         Real variables defining the RZ endpoints of the curve. These will usually correspond to the values of RZ at Reference Points 1 and 2, respectively. It is recommended that Z1 < Z2.
  - X1,X2         Real variables defining the values of RX at Z1 and Z2. These are used if the user requests constant X steps and only make sense if the described curve is monotonic. It is recommended that X1 < X2.
  - FG            +1 for outside cut, -1 for inside cut. This flag is used when calculating offsets due to oversizes or tool radii.

Following is a routine by routine description of the program:

- 1-29 Loads a "PRINT USING" utility, dimensions the arrays, calls the user routine 900, displays a screen heading, loads default values from disk (routine 1800), and jumps to line 100 to display the main menu.
- 30-99 Subroutine that displays five lines of the data array on the screen.
- 100-199 The main menu. Options and destinations are:  
 (T) - Make new table → 200  
 (E) - End program → APPLESOFT.
- 200-219 Directs flow of the process in calculating and displaying a table. Sets up the table (2000), gets user inputs (1000), saves user inputs to disk (1830), branches to the proper calculation routine (220 or 300), and returns to the main menu (100).
- 220-299 Subroutine called by 200 to calculate data array with constant RZ steps. Calculates 500 points or the entire array (whichever is less), calls the display menu (400), and then completes the table if required before returning to 200.
- 300-349 Similar to 220 except for constant RX steps.
- 350-399 Stores calculated points in the data array. Data stored is Z, X, and the increments of Z and X. Called by 220 or 330.
- 400-449 The display menu. Options and destinations are:  
 (D) - Screen display → Subroutine 30  
 (R) - Resume calculations → Returns to calling routine  
 (P) - Prints hardcopy → Subroutine 9100  
 (E) - Ends program → APPLESOFT
- The display menu is called by 220 or 330.
- 460-479 Transforms RZ, RX to Z,X. Adds offsets and converts the radius RX to the diameter X. Flips the Z axis if the Z axis is zeroed at Reference Point 2.
- 490-499 Calculates tool radius offsets needed by 460.
- 1000-1749 Prompts and accepts user inputs for configuring the table.
- 1800-1829 Three routines that read and write the user inputs to  
 1830-1849 disk for use as defaults. If an attempt is made to read a  
 1880-1899 non-existent disk file, then "default" default values are written to the disk. The name of the disk default file is TI\$+".DEF".

- 2000-2099 Routines to set up the table. Defines the proper rounding on units chosen, defines column headings, and defines printer header strings.
- 5300-5340 Printer initialization routine.
- 9100-9400 Routine to dump the data array to the printer.

Please report any bugs or problems to the author. It is recommended that anyone who modifies this program to describe a new part should generate a dedicated disk containing only his version, clearly identified. The disk must contain a copy of the "PU&F.0" binary program. I have found it helpful to have the greeting (HELLO) program run the proper "LATHE TABLE" program so that the user need not type the program name at each power-up.

RDN/cjd

PROGRAM LENGTH= 11342 BYTES                   VARIABLES= 35 BYTES  
 FREE MEMORY= 20878 BYTES  
 START=2049 LOMEM=13391 FREE=13426 STRING=34304 HIMEM=34304

```

1  REM  LATHE TABLE VERSION 15MAR83

2  PRINT CHR$(4)"MAXFILES3":ADR = PEEK(115) + 256 * PEEK(116)
  - 533: PRINT CHR$(4)"BLOAD PU&F.0,A":ADR
3  FOR I = 1 TO 11: READ D1,D2:D1 = D1 + ADR:D2 = D2 + ADR: POKE D1
  + 1,D2 / 256: POKE D1,D2 - 256 * PEEK(D1 + 1): NEXT : DATA 3,16,17,19,
26,72,47,72,64,118,68,117,117,130,125,382,127,427,129,485,371,150
4  CALL ADR: REM  INVOKE PRINT USING (REF CALL A.P.P.L.E. IN DEPTH
  UOL 1)

5  REM  BEGIN
6  ND = 501: DIM UK(4,ND),U$(4)
8  REM  DISPLAY HEADING
10 TEXT : HOME
11 GOSUB 900: REM  READ PROGRAMMER INPUTS
12 A$ = TI$ + " PROGRAM":B$ = "NRAO-15MAR83"
14 PRINT TAB( 20 - LEN(A$) / 2);A$: PRINT TAB( 20 - LEN(B$)
  / 2);B$
16 U = 2: UTAB U + 2: POKE 34,U: REM  SCREEN WINDOW

18 REM  INITIALIZE CONSTANTS
20 D$ = CHR$(4): REM  D$=CTRL-D
22 REM  DEFAULT VALUES
24 GOSUB 1800
26 GOTO 100: REM  DISPLAY MENU

30 REM  TABLE VIDEO DISPLAY
31 HOME :A$ = "PRESS Q TO EXIT": HTAB( 20 - LEN(A$) / 2): INVERSE
  : PRINT A$: NORMAL
32 U = 4: UTAB U + 1
34 PRINT TAB( 4);U$(1); TAB( 14);U$(2); TAB( 24);U$(3); TAB( 34);U$(4)
36 POKE 34,U: REM  SCREEN WINDOW
38 UTAB 12: & RPT PRINT > "-",40
39 UTAB 14: & RPT PRINT > "-",40
44 N1 = 127:N2 = 128:N3 = - 4:N4 = - 6:N5 = 4:N6 = 0:N7 = 7:N8 =
  3:N9 = 1
45 A4% = - 16384: REM  KEY BUFFER ADDR
46 J% = 1:A1% = - 16287:A2% = - 16286: REM  PB0 AND PB1
48 GOSUB 80: REM  DISPLAY FIRST PAGE
50 WT% = 256
51 KY = PEEK(A4%): IF KY = 209 OR KY = 81 THEN POKE - 16368,0:U
  = 2: POKE 34,U: RETURN : REM  RETURN IF Q PRESSED
52 IF PEEK(A1%) < N2 THEN 51: REM  WAIT FOR BUTTON
56 SC% = N3: IF PEEK(A2%) > N1 THEN SC% = N4
58 J% = J% + SC%
60 IF J% < N9 THEN J% = N9
62 IF J% > IZ THEN J% = IZ

```



```

64 GOSUB 80
66 FOR WT = N6 TO NT%: NEXT WT: REM WAIT
68 IF PEEK (A1%) > N1 THEN NT% = NT% / N5: GOTO 56: REM BUTTON STILL
  DOWN
70 GOTO 50
80 REM DISPLAY PAGE
82 FOR LN = N6 TO N5
84 UTAB N7 + LN * N8: & CLEAR TO EOL
88 & PRINT USNG > FR$:UK(1,J%),UK(2,J%),UK(3,J%),UK(4,J%)
92 J% = J% + 1
94 IF J% > IZ THEN LN = 5: & CLEAR TO EOP
96 NEXT LN
98 PRINT CHR$(0): RETURN : REM

```

```

100 HOME : UTAB U + 2: REM MENU DISPLAY
105 PRINT "PRESS (T) TO MAKE NEW TABLE"
110 HTAB 7: PRINT "(E) TO END PROGRAM"
180 PRINT : & CLEAR TO EOL: INPUT "SELECTION? ";P$
185 IF P$ = "T" THEN GOTO 200
187 IF P$ = "E" THEN TEXT : HOME : END
199 & BELL: GOTO 100: REM FAULTY ENTRY

```

```

200 REM MAIN PROGRAM
202 CH = 1: REM INITIALIZE HEADING FLAG
208 GOSUB 2000: REM SET UP TABLE
210 GOSUB 1000: REM GET USER INPUTS
211 U3$ = U3$ + STR$(FN R4(TC)) + ", OVERSIZE DIA = " + STR$(
  FN R4(OV))
212 U4$ = U4$ + STR$(RF) + ", TOOL RAD = " + STR$(FN R4(TR)):
  REM HEADER INFO
214 HOME : UTAB 11:A$ = "PLEASE WAIT": PRINT TAB( 20 - LEN (A$)
  / 2);A$
216 GOSUB 1830: REM SAVE DEFAULT VALUES
217 I = FRE (0): REM GARBAGE COLLECTION
218 ON BR GOSUB 220,300: REM BRANCH FOR CONSTANT Z STEP OR X STEP
219 GOTO 100: REM DISPLAY MENU

```

```

220 REM CONSTANT Z STEP
222 RZ = SZ: REM INITIALIZE RZ
224 GOSUB 500: GOSUB 460: REM CALC INITIAL RX AND Z,X
225 ZO = Z:XO = X: REM INITIALIZE OLD POINTS
230 I = 0: REM BEGIN COUNT
231 IF DD < 0 AND RZ < EZ THEN RZ = EZ
232 IF DD > 0 AND RZ > EZ THEN RZ = EZ
237 GOSUB 500: REM CALC RX(RZ)
238 GOSUB 460: REM CALC Z,X
240 GOSUB 350: REM SAVE DATA
245 RZ = RZ + DD
250 IF I = NO - 1 THEN CP = 0: GOTO 265: REM ARRAY FULL
253 IF DD < 0 AND RZ > EZ + DD + 1E - 5 THEN 232
255 IF DD > 0 AND RZ < EZ + DD - 1E - 5 THEN 232
260 CP = 1: REM END OF TABLE FLAG
265 IZ = I
270 GOSUB 400: REM PROMPT USER FOR TYPE DISPLAY
280 IF NOT CP THEN 230
285 RETURN : REM TABLE COMPLETE

```

```

300 REM ROUTINE FOR CONSTANT X STEP
302 RX = SX: REM INITIALIZE RX
303 GOSUB 700: GOSUB 460: REM CALC RZ AND Z,X
304 ZO = Z: XO = X: REM INITIALIZE OLD DATA
308 I = 0: REM BEGIN COUNT
310 REM BEGIN LOOP
311 IF DD < 0 AND RX < EX THEN RX = EX
312 IF DD > 0 AND RX > EX THEN RX = EX
316 GOSUB 700: GOSUB 460: REM CALC RZ AND Z X
318 GOSUB 350: REM STORE DATA
320 RX = RX + DD
322 IF I = ND - 1 THEN CP = 0: GOTO 330: REM ARRAY FULL
323 IF DD < 0 AND RX > EX + DD + 1E - 5 THEN 310
325 IF DD > 0 AND RX < EX + DD - 1E - 5 THEN 310
327 CP = 1: REM END OF TABLE
330 IZ = I
332 GOSUB 400: REM PROMPT USER FOR TYPE DISPLAY
336 IF NOT CP THEN 308
340 RETURN : REM TABLE COMPLETE

350 REM STORE DATA IN ARRAY
355 DZ = Z - ZO: ZO = Z
360 DX = X - XO: XO = X: REM CALC STEPS
365 I = I + 1
370 U(1,I) = FN R4(Z * ZS): U(2,I) = FN R4(DZ * ZS)
375 U(3,I) = FN R4(X - TC): U(4,I) = FN R4(DX): REM TC IS CENTER
OFFSET
380 RETURN : REM

400 REM SECONDARY MENU FOR DISPLAY TYPE
405 HOME
410 UTAB U + 2: PRINT "NUMBER POINTS IS "; IZ
412 IF NOT CP THEN UTAB U + 3: A$ = "MEMORY FULL - CALCULATIONS INCOMPLETE": INVERSE : PRINT A$: NORMAL
413 IF CP THEN UTAB U + 3: A$ = "CALCULATIONS COMPLETE": PRINT A$
415 UTAB U + 5: PRINT "PRESS (P) TO PRINT TABLE"
418 HTAB 7: PRINT "(D) TO DISPLAY TABLE"
420 HTAB 7: PRINT "(R) TO RESUME CALCULATIONS"
422 HTAB 7: PRINT "(E) TO END PROGRAM"
425 PRINT : & CLEAR TO EOL: INPUT "SELECTION? "; P$
428 IF P$ = "D" THEN GOSUB 30: GOTO 405: REM DISPLAY TABLE
430 IF P$ = "R" THEN 445: REM CALC REST OF POINTS OR RETURN TO MENU
435 IF P$ = "P" THEN GOSUB 910: GOTO 405: REM PRINT TABLE
437 IF P$ = "E" THEN POP : POP : TEXT : HOME : END : REM END PROGRAM
440 & BELL: GOTO 415: REM FAULTY INPUT
445 IF CP THEN RETURN : REM IF TABLE FINISHED, DISPLAY MENU1
447 HOME : A$ = "PLEASE WAIT": B$ = "CALCULATING MORE POINTS"
448 UTAB 10: PRINT TAB( 20 - LEN (A$) / 2); A$: PRINT : PRINT TAB(
20 - LEN (B$) / 2); B$
449 RETURN : REM ELSE CALCULATE REST OF TABLE

460 REM TRANSFORMS RZ,RX TO Z,X
462 GOSUB 600: GOSUB 490: REM CALC SLOPE AND OFFSETS
464 X = 2 * (RX + FX - TR): REM ADD OFFSETS AND CONVERT TO DIA
466 Z = RZ + FZ
470 REM CHANGE Z ZERO TO REF PT 2
472 IF RF = 2 THEN Z = Z2 - Z
479 RETURN : REM

```

```

490 REM CALCULATES FZ AND FX, TOOL RADIUS OFFSETS
492 AG = ATN (SL)
494 FZ = - FG * (TR + OU / 2) * SIN (AG)
496 FX = FG * (TR + OU / 2) * COS (AG)
499 RETURN : REM

```

```

500 REM ROUTINE THAT CALCULATES RX GIVEN RZ
520 IF RZ < Z1 THEN RX = X1: RETURN
530 IF RZ > 12.05 THEN RX = X2: RETURN
540 RX = 1.4810 + 0.86725 * (1 + SIN (.260713083 * (RZ - 6.025)))
599 RETURN : REM

```

```

600 REM ROUTINE TO CALCULATE SLOPE AT POINT RZ,RX
610 IF RZ < 0 THEN SL = 0: RETURN
615 IF RZ > 12.05 THEN SL = 0: RETURN
620 SL = 0.22610342 * COS (.260713083 * (RZ - 6.025))
699 RETURN : REM

```

```

700 REM ROUTINE TO CALCULATE RZ GIVEN RX
712 IF RX < X1 THEN RZ = Z1: RETURN
714 IF RX > X2 THEN RZ = Z2: RETURN
720 R1 = 1.15307 * RX - 2.7076967
725 IF R1 > = 1.0 THEN C1 = 1.5707963: GOTO 740
728 IF R1 < = - 1.0 THEN C1 = - 1.5707963: GOTO 740
730 C1 = ATN (R1 / SQR (- R1 * R1 + 1)): REM ARC SINE
740 RZ = 3.8356341 * C1 + 6.025
899 RETURN : REM

```

```

900 REM ROUTINE TO SET UP TABLE PARAMETERS
903 UNIT% = 0: REM 0 FOR INCHES, 1 FOR MILLIMETERS
905 FG = 1: REM +1 FOR OUTSIDE CUT, -1 FOR INSIDE
910 TI$ = "LOMT MANDREL"
914 UI$ = "L-BAND OMT MANDREL"
920 REM EXTREMES OF Z AND X
922 Z1 = 0:Z2 = 14.1000
924 X1 = 1.4810:X2 = 3.2155
999 RETURN : REM

```

```

1000 REM ROUTINE TO GET USER INPUTS
1005 HOME : UTAB U + 1:A$ = "ENTER 00 TO EXIT"
1007 HTAB 20 - LEN (A$) / 2: INVERSE : PRINT A$: NORMAL
1008 U = 3: POKE 34,U: REM

```

```

1010 UTAB U + 2: PRINT "START AT REF POINT ";
1020 INVERSE : PRINT "1";: NORMAL
1030 PRINT " OR ";
1040 INVERSE : PRINT "2";: NORMAL
1050 PRINT " ";:OL$ = STR$ (RF): GOSUB 1700: REM DISPLAY OLD VALUE
1052 POKE 49168,0: INPUT UI$: GOSUB 1650: REM CLR KEYS, ACCEPT INPUT,
LOOK FOR "00" INPUT OR JUST RETURN FOR OLD VALUE
1055 UI$ = LEFT$ (UI$,1)
1060 IF UI$ = "1" THEN RF = 1: PRINT "--> ";RF: GOTO 1100
1070 IF UI$ = "2" THEN RF = 2: PRINT "--> ";RF: GOTO 1100

```

```

1080 & BELL: GOTO 1010: REM FAULTY INPUT

1100 UTAB U + 5: PRINT "CARRIAGE MOVES TOWARD ";
1110 INVERSE : PRINT "H": NORMAL
1120 PRINT "EAD OR ";
1130 INVERSE : PRINT "T": NORMAL
1140 PRINT "AIL      STOCK";
1142 IF ZS < 0 THEN OL$ = "H"
1144 IF ZS > 0 THEN OL$ = "T"
1146 GOSUB 1700
1148 INPUT UI$: GOSUB 1650
1150 UI$ = LEFT$(UI$,1)
1160 IF UI$ = "H" THEN ZS = - 1.0: PRINT "--> ";UI$: GOTO 1200
1170 IF UI$ = "T" THEN ZS = 1.0: PRINT "--> ";UI$: GOTO 1200
1180 & BELL: GOTO 1100: REM FAULTY INPUT

1200 UTAB U + 8: PRINT "CONSTANT ";
1210 INVERSE : PRINT "Z": NORMAL
1220 PRINT " STEP OR CONSTANT ";
1230 INVERSE : PRINT "X": NORMAL
1240 PRINT " STEP";
1242 IF BR = 1 THEN OL$ = "Z"
1243 IF BR = 2 THEN OL$ = "X"
1244 GOSUB 1700
1245 INPUT UI$: GOSUB 1650
1247 UI$ = LEFT$(UI$,1)
1250 IF UI$ = "Z" THEN BR = 1: PRINT "--> ";UI$: GOTO 1280
1260 IF UI$ = "X" THEN BR = 2: PRINT "--> ";UI$: GOTO 1280
1270 & BELL: GOTO 1200: REM FAULTY INPUT

1280 UTAB U + 11: PRINT "TOOL RADIUS";UNIT$;
1282 OL$ = STR$(TR): GOSUB 1700
1284 INPUT UI$: GOSUB 1650
1286 TR = VAL (UI$): PRINT "--> "; FN R4(TR)
1289 REM

1300 UTAB U + 14: PRINT "SIZE OF STEP";UNIT$;
1302 OL$ = STR$( ABS (DD))
1303 IF BR = 2 THEN OL$ = STR$( FN R4( ABS (DD) * 2))
1304 GOSUB 1700
1310 INPUT UI$: GOSUB 1650
1320 DD = VAL (UI$)
1325 IF DD < = 0 THEN & BELL: GOTO 1300: REM FAULTY INPUT
1327 PRINT "--> "; FN R4(DD): REM

1330 UTAB U + 17: PRINT "X AXIS OVERSIZE DIA";UNIT$;
1332 OL$ = STR$(OU): GOSUB 1700
1334 INPUT UI$: GOSUB 1650
1336 OU = VAL (UI$): PRINT "--> "; FN R4(OU)
1339 REM

1340 UTAB U + 19: PRINT "ABSOLUTE X AXIS ZEROED AT ";UNIT$;
1342 OL$ = STR$(TC): GOSUB 1700
1344 INPUT UI$: GOSUB 1650
1346 TC = VAL (UI$): PRINT "--> "; FN R4(TC)

```

```

1343 REM

1350 REM SETUP TABLE RANGES
1351 IF RF = 1 THEN SX = X1:EX = X2:SZ = Z1:EZ = Z2
1352 IF RF = 2 THEN SX = X2:EX = X1:SZ = Z2:EZ = Z1
1353 IF BR = 2 THEN DD = DD / 2: REM CONVERT DIA TO RADIUS
1354 IF SX > EX AND BR = 2 THEN DD = - DD
1355 IF SZ > EZ AND BR = 1 THEN DD = - DD
1356 HOME : UTAB U + 1: FLASH : PRINT "RANGES DO NOT INCLUDE TOOL
RADIUS OFFSET": NORMAL
1358 PRINT "RANGE OF Z ": PRINT FN R4(Z1), FN R4(Z2)
1360 UTAB U + 5: PRINT "RANGE OF X :"
1362 PRINT FN R4(2 * SX - TC), FN R4(2 * EX - TC)
1364 UTAB U + 8: INVERSE : PRINT "P": NORMAL
1366 PRINT "ARTIAL OR "
1368 INVERSE : PRINT "T": NORMAL
1370 PRINT "OTAL TABLE":
1372 CALL - 868: INPUT UI$: GOSUB 1650
1374 UI$ = LEFT$(UI$,1)
1376 IF UI$ = "T" THEN PRINT "--> ";UI$: GOTO 1600
1378 IF UI$ = "P" THEN PRINT "--> ";UI$: GOTO 1400
1380 & BELL: GOTO 1364: REM FAULTY INPUT

1400 REM PARTIAL TABLE
1420 ON BR GOTO 1430,1500: REM BRANCH ON CONSTANT Z OR X STEPS

1430 UTAB U + 12: PRINT "STARTING Z";UNIT$:
1432 OL$ = STR$(Z1): GOSUB 1700
1440 INPUT UI$: GOSUB 1650
1450 Z3 = VAL(UI$): PRINT "--> "; FN R4(Z3)
1460 UTAB U + 14: PRINT "ENDING Z";UNIT$:
1462 OL$ = STR$(Z2): GOSUB 1700
1470 INPUT UI$: GOSUB 1650
1480 Z4 = VAL(UI$): PRINT "--> "; FN R4(Z4)
1490 IF (Z4 < = Z3) THEN & BELL: UTAB U + 10: PRINT "ERROR IN ENDPPOINTS
": GOTO 1430
1495 IF RF = 1 THEN SZ = Z3:EZ = Z4
1497 IF RF = 2 THEN SZ = Z2 - Z3:EZ = Z2 - Z4
1499 GOTO 1600: REM CONTINUE

1500 UTAB U + 12: PRINT "STARTING X";UNIT$:
1505 OL$ = STR$(2 * SX - TC): GOSUB 1700
1510 INPUT UI$: GOSUB 1650
1520 X3 = VAL(UI$) + TC: PRINT "--> "; FN R4(X3): REM SUBTRACT TOOL
OFFSET
1530 UTAB U + 14: PRINT "ENDING X";UNIT$:
1535 OL$ = STR$(2 * EX - TC): GOSUB 1700
1540 INPUT UI$: GOSUB 1650
1550 X4 = VAL(UI$) + TC: PRINT "--> "; FN R4(X4)
1560 IF (DD < 0 AND X3 < X4) OR (DD > 0 AND X3 > X4) THEN & BELL:
UTAB U + 10: PRINT "ERROR IN ENDPPOINTS": GOTO 1500
1565 SX = X3 / 2:EX = X4 / 2
1575 REM CONTINUE

1600 U = 2: POKE 34,U: HOME : REM SCREEN WINDOW
1610 RETURN : REM END USER INPUT

```

```

1650 REM EXIT ROUTINE WHEN "00" INPUT OR USE OLD VALUE IF NULL STRG
1655 IF LEFT$(UI$,2) = "00" THEN POP : POP : U = 2: POKE 34,U: GOTO
100: REM DISPLAY MENU
1657 IF UI$ = "" THEN UI$ = OL$
1658 % CLEAR TO EOL
1660 RETURN : REM

```

```

1700 REM DISPLAY OLD INPUT
1710 PRINT "/" + OL$ + "/"
1715 % CLEAR TO EOL
1749 RETURN : REM

```

```

1800 REM LOAD DEFAULT VALUES FROM DISK
1803 ONERR GOTO 1880
1806 PRINT CHR$(0):TD$ = TI$ + ".DEF"
1809 PRINT D$;"OPEN";TD$
1812 PRINT D$;"READ";TD$
1815 INPUT RF,ZS,BR,DD,OU,TC,TR
1818 PRINT D$;"CLOSE";TD$
1820 POKE 216,0: REM CLEAR ERR FLG
1829 RETURN : REM

```

```

1830 REM WRITES CURRENT DEFAULT VALUES TO DISK
1832 PRINT CHR$(0):TD$ = TI$ + ".DEF":CM$ = ","
1834 PRINT D$;"OPEN";TD$
1836 PRINT D$;"DELETE";TD$
1838 PRINT D$;"OPEN";TD$
1839 PRINT D$;"WRITE";TD$
1840 PRINT RF;CM$;ZS;CM$;BR;CM$;DD;CM$;OU;CM$;TC;CM$;TR
1845 PRINT D$;"CLOSE";TD$
1849 RETURN : REM

```

```

1880 REM IF NO DEFAULT FILE, MAKE ONE
1882 PRINT CHR$(0): PRINT D$;"CLOSE"
1883 RF = 1:ZS = 1.0:BR = 1:DD = 0:OU = 0:TC = 0:TR = 0
1886 GOSUB 1830
1888 GOTO 22
1899 REM

```

```

2000 REM ROUTINE TO SET UP TABLE
2010 ON UNIT% + 1 GOSUB 2090,2090
2020 U$(1) = "Z" + UNIT$: REM COLUMN HEADINGS
2025 U$(2) = "DZ" + UNIT$
2030 U$(3) = "X" + UNIT$
2035 U$(4) = "DX" + UNIT$
2040 NC = 4: REM FOUR COLUMNS
2045 U3$ = "CROSSFEED X READOUT ZEROED AT DIA= "
2050 U4$ = "CARRIAGE Z READOUT ZEROED AT REF POINT "
2079 RETURN : REM

```

```

2080 REM INCHES
2082 UNIT$ = "(IN)"
2084 DEF FN R4(X) = INT ((X + 0.00025) * 2000) / 2000: REM ROUNDING

```

FUNCTION

2085 FR\$ = " 0.0000": REM FORMAT STRING  
2086 RETURN : REM

2090 REM MILLIMETERS

2092 UNIT\$ = "(MM)"

2094 DEF FN R4(X) = INT ((X + 0.005) \* 100) / 100: REM ROUNDING  
FUNCTION

2095 FR\$ = " 0.00": REM FORMAT STRING

2096 RETURN : REM

5300 REM TURNS ON TRENDCOM PRINTER

5305 D\$ = CHR\$(4): REM D\$=CTRL D

5307 PRINT CHR\$(0)

5310 PRINT D\$;"PR# 1": PRINT CHR\$(0)

5320 POKE 1913,0: POKE 1785,79: REM MARGIN

5330 POKE 1657,80: REM LINE LENGTH

5340 RETURN : REM

9100 REM DUMPS TABLE TO PRINTER

9110 HOME : VTAB 11:A\$ = "PRINTING, PRESS 0 TO TERMINATE": PRINT  
TAB( 20 - LEN (A\$) / 2);A\$

9120 FP\$ = " " + FR\$: REM ADD 8 SPACES TO FORMAT

9200 REM PRINTS TABLE OF CALCULATED DATA

9202 GOSUB 5300: REM TURN ON PRINTER

9203 I\$ = CHR\$(9): REM I\$=CTRL I

9205 PRINT I\$;"N": REM DISABLE VIDEO

9207 POKE 33,80: REM SET APPLESOFT LINE LENGTH

9208 IF NOT CH THEN 9255

9209 PRINT TAB( 40 - LEN (U1\$) / 2);U1\$: PRINT : REM PRINT HEADING

9210 PRINT TAB( 40 - LEN (U2\$) / 2);U2\$

9212 PRINT TAB( 40 - LEN (U3\$) / 2);U3\$

9214 PRINT TAB( 40 - LEN (U4\$) / 2);U4\$: PRINT : REM PRINT HEADING

9216 REM PRINT COLUMNAR HEADINGS

9220 FOR J = 1 TO NC: PRINT TAB( 5 + (2 \* J - 1) \* 9 - LEN (U\$(J))  
/ 2);U\$(J);: NEXT J

9225 PRINT CHR\$(0):CH = 0

9230 GOSUB 9300

9255 REM PRINT DATA

9260 FOR J = 1 TO IZ

9275 & PRINT USING > FP\$;U(1,J),U(2,J),U(3,J),U(4,J)

9282 KY = PEEK ( - 16384): IF KY = 209 OR KY = 81 THEN J = IZ + 1:  
REM "0" TO TERMINATE PRINT.

9283 POKE - 16368,0: PRINT

9285 K = K + 1: IF K > = 5 THEN GOSUB 9300

9287 NEXT J

9289 PRINT : PRINT : PRINT

9290 PRINT D\$;"PR#0": REM TURN PRINTER OFF

9292 POKE 33,40: PRINT I\$;"I": REM RESTORE VIDEO

9295 RETURN : REM

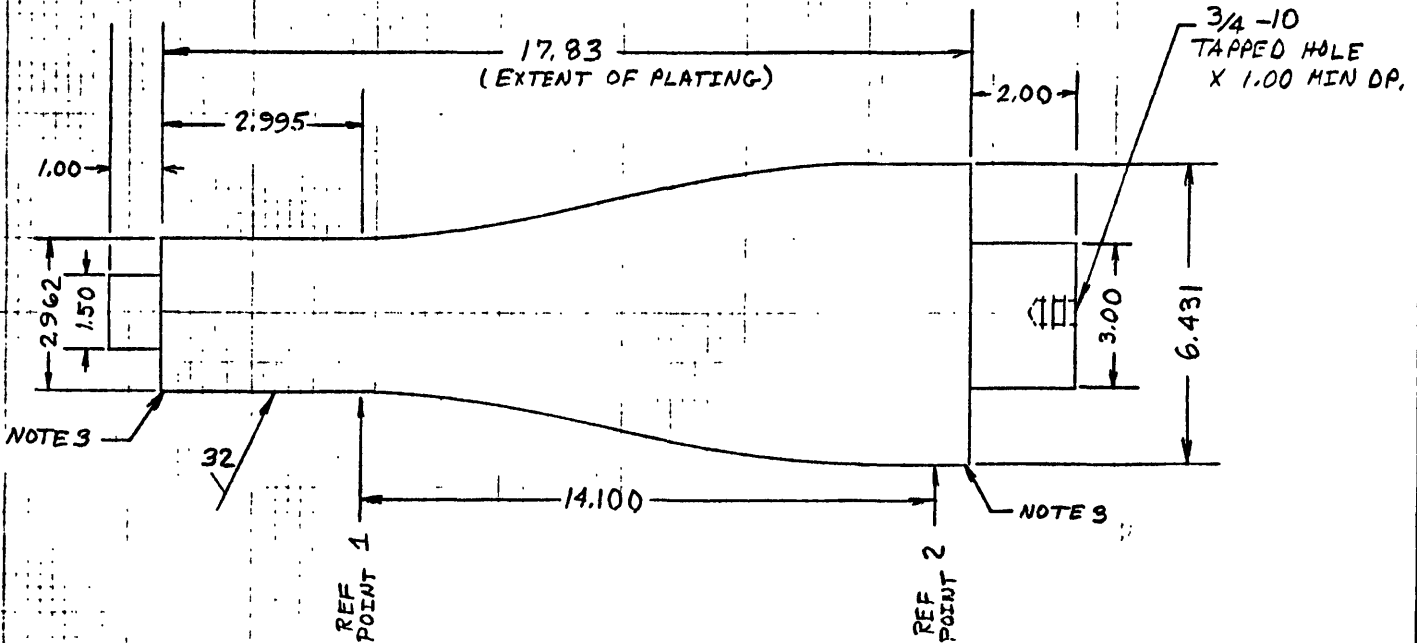
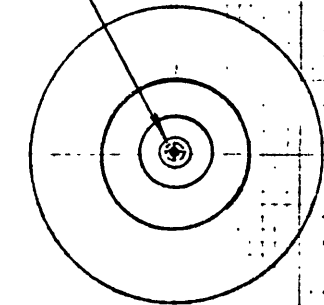
9300 FOR II = 1 TO 20 \* NC

9310 PRINT "-";

9320 NEXT II

9330 K = 0: PRINT CHR\$(0): RETURN

1/4-20 TH03 X .50 MIN DP.  
60° CENTER



NOTE 3

32

REF POINT 1

REF POINT 2

NOTE 3

3. PLATING NEED NOT WRAP AROUND CORNERS.  
OK TO MASK SURFACE UP TO 0.20 INCHES FROM CORNERS.

2. MANDREL MATERIAL AL6061-T6

NOTE: 1. ELECTROFORM COPPER ABOUT MANDREL  
TO MINIMUM THICKNESS OF 0.075.  
MASK ENDS.

FINISH:  
64 UNLESS NOTED

TOL:  
.XX = ±.020  
.XXX = ±.003

DIMS IN INCHES

NRAO GREEN BANK, WV		
SCALE: NTS	APPROVED BY:	DRAWN BY RON
DATE: 30 DEC 82		REVISED
OMT SHELL PROTOTYPE MANDREL L BAND OMT		
2.627-215-002		DRAWING NUMBER



PROGRAM: LOMT MANOREL

A# MISC USE  
12 14 31 214 412 413 447 448 1005 1007 9110

A1% ADDR OF PBO  
46 52 68

A2% ADDR OF PB1  
46 56

A4% ADDR OF KEYBOARD BUFFER  
45 51

ADR ADDR OF PU&F  
3 4 2

AG SLOPE OF CURVE (ANGULAR)  
492 494 496

B# MISC USE  
14 447 448 12

BR = 1 for constant  $z$  step, = 2 for constant X  
218 1242 1243 1250 1260 1303 1353 1354 1355  
1420 1815 1840 1883

C1 Temp Storage  
725 728 730 740

CH = 0 if printer header printed, = 1 otherwise  
202 9200 9225

CM# " , "  
1832 1840

CP = 0 if memory full, = 1 if calculations finished  
250 260 280 322 327 336 412 413 445

D# = CTRL-D  
20 1809 1812 1818 1834 1836 1838 1839 1845 1882  
5305 5310 9290

D1 Temp Storage  
3

D2 Temp Storage  
3

DD RZ OR RX STEP  
271 232 245 253 255 311 312 320 323 325 1302  
1303 1320 1325 1327 1353 1354 1355 1560 1815 1840  
1883

DX CHANGE IN X

360 375

DZ *CHANGE IN Z*  
355 370

E  
~~253 232 323 325~~

EX *ENDING X*  
311 312 323 325 1351 1352 1354 1362 1535 1565

EZ *ENDING Z*  
253 232 1351 1352 255 1355 1495 1497 231

FG = +1 if outside cut, -1 if inside cut  
464 496 905

FH R40 *ROUNDING Function*  
212 211 370 375 1296 1303 1327 1336 1346 1358  
1362 1450 1480 1520 1550 2084 2094

FP# *PRINTER Format string*  
9120 9275

FR# *Screen Format string*  
88 2085 2095 9120

FX *X Offset*  
464 496

FZ *Z Offset*  
466 494

I *Counter Index*  
370 375 365 3 217 322 265 250 230 308 330

I# *CTRL-I*  
9203 9205 9292

II *COUNTER Index*  
9300 9320

IZ *Number lines in data array*  
265 62 94 330 9260 9292 410

J *Printer routine counter index*  
9220 9260 9275 9292 9287

J# *Screen display counter*  
58 46 60 88 92 94 62

K *Printer lines counter*  
9285 9330

KY *Keypress Value*  
51 9282

LN *Screen display index*  
84 94 96 82

N1 = 127  
 56 68 44  
 N2 = 128  
 52 44  
 N3 = -4  
 56 44  
 N4 = -6  
 56 44  
 N5 = 4  
 44 68 82  
 N6 =  $\emptyset$   
 44 82 66  
 N7 = 7  
 44 84  
 N8 = 3  
 84 44  
 N9 = 1  
 44 68  
 NC *Number columns printed*  
 2040 9220 9300  
 ND *Dimension of arrays*  
 322 258 6  
 OL# *Old or default user input string*  
 1050 1142 1144 1242 1243 1282 1302 1303 1332  
 1342 1432 1462 1505 1535 1657 1710  
 OU *Oversize diameter value*  
 1332 1336 211 494 1815 1840 1883 496  
 P# *User menu input*  
 185 187 430 435 437 425 180 428  
 R1 *Temp Storage*  
 720 725 728 730  
 RF *Starting Reference Point*  
 1870 212 1351 1352 1060 1495 1497 1815 1840  
 1883 472 1050  
 RX *"Real" X*  
 720 302 520 323 712 714 530 325 312 320 540  
 464 311  
 RZ *"Real" Z*  
 245 222 620 253 255 714 740 540 520 231 615  
 530 610 232 712 466  
 SC# *Direction of screen scroll*  
 58 56

SL *Slope of curve*

610 615 492 620

SM *Starting X*

1351 1352 302 1354 1362 1505 1565

SN *Starting Z*

1351 1352 1355 1495 1497 222

TC *Diameter of X axis zero point*

211 1342 1362 1346 1505 1523 1535 1550 1815  
1840 1883 375

TD# *Default disk file name*

1806 1809 1812 1818 1832 1834 1836 1838 1839  
1845

TI# *User defined program name*

910 1806 1832 12

TR *Tool Radius*

496 494 1815 1286 1840 212 1883 464 1282

U# *column Headings*

6 34 2020 2025 2030 2035 9220

UK *Data Array*

88 370 9275 6 375

U1#

914 9209

U2#

9210

U3#

211 2045 9212

U4#

212 2050 9214

} *Printer Heading strings*

UI# *User input String*

1470 1480 1245 1344 1148 1247 1510 1520 1250  
1070 1540 1550 1260 1655 1657 1320 1286 1150 1346  
1336 1055 1378 1334 1060 1450 1160 1372 1374 1284  
1440 1376 1170 1310 1052

UNIT# *"(IN)" or "(MM)"*

1300 1500 1460 2020 2025 2030 2035 2082 2092  
1280 1530 1340 1430 1330

UNIT% =  $\emptyset$  if inch units, 1 if mm

2010 903

U *Top of screen window*

1300 51 1500 1356 1280 1490 1005 413 1655 1364  
36 1100 1560 1200 16 1340 100 32 410 1530 1460  
1360 1008 415 1430 1010 1600 1330 412

HT *Wait counter*

66

WT% = 256

58 68 66

X = Cross feed axis

375 2084 225 360 2094 464 304

X1 = Ref Pt 1 RX coordinate

924 1352 712 520 1351

X2 = Ref Pt 2 RX coordinate

530 1351 1352 714 924

X3 Temp Storage

1560 1565 1520

X4 Temp Storage

1560 1550 1565

X0 Previous X value

225 360 304

Z Lathe Bed Axis

370 355 472 304 466 225

Z1 Ref Pt 1 RZ Coordinate

1432 1351 1352 520 922 1358 712

Z2 Ref Pt 2 RZ Coordinate

1351 1358 1462 472 1497 714 922 1352

Z3 Temp Storage

1495 1450 1497 1490

Z4 Temp Storage

1497 1495 1490 1480

Z0 Previous Z value

304 355 225

Z5 = -1 if move toward headstock, +1 if move toward tailstock

1160 1170 1142 1883 1840 370 1144 1815

