RECOMP II USERS' PROGRAM NO. 1158

PROGRAM TITLE:	FLOATING POINT TO PLOTTER INCREMENT CONVERSION SUBROUTINE
PROGRAM CLASSIFICATION:	Subroutine
AUTHOR:	R. Doyle
PURPOSE:	Given the coordinates of a point (X, Y) as a pair of floating point numbers, compute the number of x and y plotter increments necessary to move the pen from its present position to the given point. This subroutine is to be used in conjunction with the "Line Plotter", "Point Plotter", and "Plot X then Y" subroutines.
DATE:	28 November 1962

Published by

RECOMP Users' Library

at

AUTONETICS INDUSTRIAL PRODUCTS

A DIVISION OF NORTH AMERICAN AVIATION, INC.

3400 East 70th Street, Long Beach 5, California

Program Title: Floating Point to Plotter Increment Conversion Subroutine

1. Purpose: Given the coordinates of a point (X, Y) as a pair of floating point numbers, compute the number of x and y plotter increments necessary to move the pen from its present position to the given point. This subroutine is to be used in conjunction with the "Line Plotter," "Point Plotter," and "Plot X then Y" subroutines.

2. Restrictions: The coordinates of the point must be normalized floating point numbers, and should be consistent with scale factor and available plotting space.

3. Method

3.1 Let

X = the value of the x coordinate of the point to which the pen is to be moved

X_s = scale factor, defined to be the number of units per inch, for the x coordinate

x = the value of the x coordinate of the point to which the pen is to be moved in units of plotter increments (1/100 inch)

x_p = the value of the x coordinate of the present position of the pen in units of plotter increments

Y, Y_s, y, and y_p are similarly defined

3.2 The floating point value X is divided by $X_s/100$, and converted to a fixed point rounded integer to give x; y is similarly computed. The differences

$$\triangle X = x - x_p$$

$$\triangle Y = y - y_p$$

are the required number of plotter increments. Finally, x and y replace x_p and y_p respectively.

3.3 The subroutine must be initialized by supplying the two scale factors, X_s and Y_s, and the initial position of the pen.

4. Use: This subroutine is intended to convert floating point data to the form required by the "Line Plotter," Point Plotter," and "Plot X then Y" subroutines. Thus, calling this routine will be followed by calling one of the above mentioned routines.

4.1 Initialization:

Before plotting, the pen should be positioned manually and the following values supplied to the subroutine. This is to be accomplished by storing them in the V-loop and transferring to $L_o + 30$.

7770 X_o = x coordinate of the initial pen position
72 Y_o = y " " " " " "
74 X_s = x scale factor in units/inch
76 Y_s = y " " " "

X, Y (and X_o, Y_o) are given in arbitrary units of the users choice; the scale factors X_s, Y_s are the desired number of these arbitrary units per inch of pen motion.

4.2 Calling Sequence: Transfer to the origin of the subroutine (L_o) with the following word in the accumulator:

+ 00 XXXX0 + 00 YYYY 0

where XXXX = location of floating point X coordinate
YYYY = " " " " Y "

Return to the next location with Δ X in the A-register and Δ Y in the R-register as fixed point integers at b = 39. Δ X and Δ Y are the number of plotter increments necessary to move the pen from its present position, to the position specified by the coordinate values X and Y. Note that the output of this subroutine in the exact form required by "Line Plotter," "Point Plotter," and "Plot X then Y" subroutines.

4.3 Applications

4.3.1 In what follows it is assumed that initialization has been accomplished in accordance with 4.1.

Let

L_o = origin of "Floating Point to Plotter Increment Conversion" subroutine

LP = origin of "Line Plotter" subroutine

PP = origin of "Point Plotter" subroutine

PXY = origin of"Plot X then Y"subroutine

KW = + 00 XXXX 0 + 00 YYYY 0

4.3.2 To plot straight line from present pen position to point (X,Y):

```
CLA  KW
TRA  Lo
TRA  LP
RETURN
```

4.3.3 To move the pen, in the raised position, to the point (X,Y):

```
CLA  KW
TRA  L_o
TRA  PP
RETURN
```

4.3.4 To plot a straight line from present position (X_p, Y_p) to point (X,Y_p), followed by a straight line to (X,Y); i.e., to plot a step in a bar graph:

```
CLA  KW
TRA  L_o
TRA  PXY
RETURN
```

4.4 Reset:

At anytime the pen may be returned to the initial position (X_o, Y_o) by

```
CLA  L_o + 56
TRA  L_o
TRA  PP
RETURN
```

5. Coding Information:

5.1 Locations used:

This routine occupies 57_8 locations (i.e., L_o to $L_o + 56$). It destroys both loops and all registers. All locations are used and none are erasable.

5.2 Constants

$L_o + 42, 43$ 0.5 (floating point)

$L_o + 54, 55$ 100.0 (floating point)

5.3 Variables

$L_o + 40$	X_p	(fixed point, binary scale of 39)
41	Y_p	(" " " " " ")
$L_o + 44,45$	$X_s/100$	(floating point)
$+ 46,47$	$Y_s/100$	(" ")
$+ 50,51$	X_o	(floating point)
$+ 52,53$	Y_o	(" ")

5.4 This subroutine is relocatable by method of AN-076

6. Remark

Sometimes it is desired to make a decision based on the value of the x and/or y coordinate in units of plotter increments. For example, if one is plotting a function $Y(x) > 0$ such that $Y \rightarrow 0$ as $x \rightarrow \infty$, he may wish to terminate the plot when Y becomes less than the resolution of the plotter, i.e., when Y = zero plotter increments. For this purpose it is to be noted that upon return from this subroutine the value of x and y, in units of plotter increments, is contained in locations $L_o + 40$ and $L_o + 41$ respectively (fixed point at b = 39).

```
0000.0
 + CTL 0000.0  + SAX 7760.0
 + CTV 0040.0  + TRA 7763.0
 + XAR 0000.0  + TRA 0000.1
 + ADD 7762.0  + STO 0027.0
 + CLA 7760.0  + STA 0016.1
 + STA 7766.0  + 70  0000.0
 + FCA 0000.0  + FDV 7774.0
 + CTL 0010.0  + TRA 7760.0

0010.0
 + TPL 7761.1  + FSB 7772.0
 + TRA 7762.0  + FAD 7772.0
 + XAR 0000.0  + ALS 0001.0
 + SUB 7764.0  + STA 7764.1
 + XAR 0000.0  + ARS 0047.0
 + STO 0040.0  + SUB 7770.0
 + STO 7770.0  + FCA 1252.0
 + CTL 0020.0  + TRA 7760.0

0020.0
 + FDV 7776.0  + TPL 7762.0
 + FSB 7772.0  + TRA 7762.1
 + FAD 7772.0  + XAR 0000.0
 + ALS 0001.0  + SUB 7765.0
 + STA 7765.1  + CLA 7770.0
 + XAR 0000.0  + ARS 0047.0
 + STO 0041.0  + SUB 7771.0
 + XAR 0000.0  + TRA 3057.0

0030.0
 + SAX 7760.0  + ADD 0002.0
 + STA 0037.1  + FCA 7770.0
 + FST 0050.0  + FCA 7772.0
 + FST 0052.0  + FCA 7774.0
 + FDV 0054.0  + FST 0044.0
 + FCA 7776.0  + FDV 0054.0
 + FST 0046.0  + CLA 0056.0
 + TRA 0000.0  + TRA 3041.0

0040.0
 - CLA 0000.0  - CLA 0205.1
 - CLA 0000.0  - CLA 0310.0
 + ARS 0000.0  - CLA 0000.0
 + CLA 0000.0  - CLA 0000.0
 + 75  3412.0  - 75  3412.0
 - CLA 0000.0  - CLA 0003.1
 + TZE 7534.0  - TZE 7534.0
 - CLA 0000.0  - CLA 0003.1

0050.0
 - ARS 0000.0  - CLA 0000.0
 + CLA 0000.0  - CLA 0001.0
 - ARS 0000.0  - CLA 0000.0
 + CLA 0000.0  - CLA 0001.0
 + 62  0000.0  - CLA 0000.0
 + CLA 0000.0  - CLA 0003.1
 + CLA 0050.0  + CLA 0052.0
```