MASSACHUSETTS INSTITUTE OF TECHNOLOGY

PROJECT MAC

Artificial Intelligence Project
Program Memo. No. [redacted] 112

Memorandum-MAC-M-334
October 1966

CHAR PLOT

Donald Sorillo

CHAR PLOT is a routine which enables one to use the Calcomp plotter as an output typewriter.

CHAR PLOT

This program is stored as CHPLOT BIN [English CHAR PLOT]. In use a code, representing a character or command as defined in Appendix I, is placed into accummulator C. Upon calling the routine the plotter will, either print a character, or set itself into one of several modes.

CHARACTER MODE

The input to the routine is a word whose 8 low order bits contain a code and whose sign bit must be Ø. The routine is entered by

```
MOVE C, [WORD]
PUSHJ P, PLOTC.
```

A word = 0, stops everything and initiates the system.

NOTE: The program starts off in lower case mode. While it is in this mode any attempt to issue a lower-case code causes the computer to hang up. It is suggested that the first call be used to set the routine to upper case and the 8th bit in the code used to shift between upper and lower cases.

The symbols P,C, and CRKCHN are global and user-defined. Other symbols are:

PLOTC	Normal entry point.
UCTAB	Beginning of upper case table.
LCTAB	Beginning of lower case table.
CLNGTH	Routine which returns length of the character which was its argument in Acc. C.
x y	Current position of pen. At initialization x = y = Ø.

LBUFF LWBUFF	Each 1000_8 words. Can be reduced to one word to gain more storage for other routines.
CRKBRK	Location for return from interrupt.

CONTROL MODE

The input to the routine is a word whose sign bit is a one. It is entered by:

```
MOVE C, [WORD]
PUSHJ P, PLOTC.
```

The high-order bits of the word are decoded as follows:

2*	
10	xy Format mode
110	Define CHAR/CHAR SET mode
1110	Go to eff. address at process time
1111	Half-word mode.

XY Format Mode

Bits:

2	16	2	16
10	x	code	y

The x and y words are in 2's complement. The 2-bit code is as follows:

00 set x and y, i.e., tell program the x and y coordinates of the plotter pen. This does not move the plotter pen.

01 Increment x,y with scale--move the plotter in an x and y direction the number of units equal to the product of the specified number times any scale factor.

10 Increment x,y without scale--move the plotter in an x and y direction the number of units specified.

11 go to x,y--position plotter at the given x and y coordinates.

Define CHAR/CHAR SET mode

Bits:

3	6	1	8	18
110	Pointer	0	Character	address

Used in conjuction with byte manipulation instructions. The address and pointer refer to a word and position within the word of the first 4-bit byte. [The Byte length is set to 4 automatically].

The 8-bit octal code for the desired character is in the 'character' part of the word. The word or words containing the bytes are set up as follows:

|Δx|Δy| | |...|Ø|

Δx and Δy may be zero. The first Ø after Δx and Δy terminates the string.

The codes for bytes are:

code	action
2	Pen up
4	Pen down
5	Dot down--prints a dot a current point position.
10	Right

11 Up

12 Left

13 Down

14 Up Right

15 Up Left

16 Down Left

17 Down Right

This is used to specify a new character. Once done, the character may be called by the programmer in the normal manner.

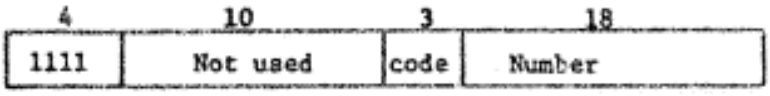
The CHAR SET feature enables the use to define a complete set of characters. The 8-bits for 'character' must be Ø to use this mode. The address part is that location which is the beginning of the character table.

Once entered the new set is the only one addressable. To return to the old set use a DEFINE CHAR SET with UCTAB as the address.

N.B.: If lower case is not defined in the new set, any code with a 2^8 bit set to one will cause the program to hang up.

HALF-WORD MODE

Bits:



In all of the below, the function is set to NUMBER. The 3-bit code is:

0 set scale

1 set horizontal tab. This sets tabs at n, 2n, 3n, etc.

2 set line feed [single space = 9 units]

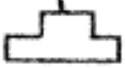
3 set vertical tab. Sets tabs at n, 2n, 3n, etc.

4 set x
5 set y
6 go to x
7 go to y

} number must be in 2's complement

APPENDIX I

CHARACTER SET (For lower case, add 200_8)

Octal Code	
0	illegal character
1	pen down
2	subscript
3	pen up
4	superscript
5	leave mode without pen controls
6	enter mode with no pen controls executed
7	 (Bell; control G)
10	set scale to zero
11	horizontal tab
12	line feed
13	vertical tab (will not go past a page boundary)
14	form feed (pages are 2000_8 units long)
15	carriage return
16	shift into upper case
17	shift into lower case
20	add 2^0 to scale [for lower case subtract]
21	add 2^1 to scale
22	add 2^2 to scale
23	add 2^3 to scale
24	add 2^4 to scale

25	add 2^5 to scale		
26	add 2^6 to scale		
27	add 2^7 to scale		
30	direct along x-axis	L	These codes affect the position of the apparent axes
31	direct along y-axis	⅃	
32	upside down x-axis	⅂	
33	upside down y-axis	F	
34	backwards x-axis	⅁	
35	backwards y-axis	Ⅎ	
36	upside down and backwards x-axis	⅃	
37	upside down and backwards y-axis	⅂	
	upper case		lower case (if no entry, same same as upper case)
40	4 unit space		3 unit space
41	!		
42	"		
43	#		
44	$		¢
45	%		
46	&		
47	'(apostrophe)		
50	(		
51	)		
52	* (asterisk)		* (superscript asterisk)
53	+		

54	,	~ (tilde)
55	-	
56	. (period)	
57	/	
60	ø	
61	1	
62	2	
63	3	
64	4	
65	5	
66	6	
67	7	
70	8	
71	9	
72	:	:
73	;	;
74	<	^
75	_	¥
76	>	`
77	?	?
100	@	@
101	A	a
102	B	b

103	C	c
104	D	d
105	E	e
106	F	f
107	G	g
110	H	h
111	I	i
112	J	j
113	K	k
114	L	l
115	M	m
116	N	n
117	O	o
120	P	p
121	Q	q
122	R	r
123	S	s
124	T	t
125	U	u
126	V	v
127	W	w
130	X	x
131	Y	y
132	Z	z

133	[	[
134	\	/ (no horizontal spacing)
135	]	]
136	↑	↑
137	←	←
140–157	Not used	
160	Right one relative unit	Right one absolute unit
161	Up one relative unit	Up one absolute unit
162	Left one relative unit	Left one absolute unit
163	Down one relative unit	Down one absolute unit
164	Up right one relative unit	Up right one absolute unit
165	Up left one relative unit	Up left one absolute unit
166	Down left one relative unit	Down left one absolute unit
167	Down right relative unit	Down right one absolute unit
170	. (dot)	. (dot)
171–177	Not used	