

1 PPPPPPPPPPPPPP	UUU	UUU	LLL	SSSSSSSSSSSSSS	AAA	RRR	5	
2 PPPPPFFFFFFFFFF	UUU	UUU	LLL	SSSSSSSSSSSSSS	AAA	RRR	6	
3 PPPPPFFFFFFFFFF	UUU	UUU	LLL	SSSSSSSSSSSSSS	AAA	RRR	7	
4 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	8
5 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	9
6 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	10
7 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	11
8 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	12
9 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	13
10 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	14
11 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	15
12 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	16
13 PPP	PPP	UUU	UUU	LLL	SSS	AAA	RRR	17
14 PPPPPPPPPPPPP	UUU	UUU	LLL	SSSSSSSSSS	AAA	RRR	18	
15 PPPPPFFFFFFFFFF	UUU	UUU	LLL	SSSSSSSSSS	AAA	RRR	19	
16 PPPPPFFFFFFFFFF	UUU	UUU	LLL	SSSSSSSSSS	AAA	RRR	20	
17 PPP	UUU	UUU	LLL	SSS	AAA	RRR	21	
18 PPP	UUU	UUU	LLL	SSS	AAA	RRR	22	
19 PPP	UUU	UUU	LLL	SSS	AAA	RRR	23	
20 PPP	UUU	UUU	LLL	SSS	AAA	RRR	24	
21 PPP	UUU	UUU	LLL	SSS	AAA	RRR	25	
22 PPP	UUU	UUU	LLL	SSS	AAA	RRR	26	
23 PPP	UUUUUUUUUUUUUUUU	LLL	LLL	SSSSSSSSSSSS	AAA	RRR	27	
24 PPP	UUUUUUUUUUUUUUUU	LLL	LLL	SSSSSSSSSSSS	AAA	RRR	28	
25 PPP	UUUUUUUUUUUUUUUU	LLL	LLL	SSSSSSSSSSSS	AAA	RRR	29	
26							29	
27							30	
28							31	
29							32	
30							33	
31							34	
32							35	
33							36	
34							37	
35							38	
36							39	
37							40	
38							41	
39							42	
40							43	
41							44	
42							45	
43							46	
44							47	
45							48	
46							49	
47							50	
48							51	
49							52	
50							53	
51							54	
52							55	
53							56	
54	LPTSPL VERSION 6(344) RUNNING ON LPT000						57	
55	*START* USER JNO PROJECTS 13000,30001 JOB PULSAR SEQ. 17951 DATE 12-APR-79 14:30:48 MONITOR ALBUQUERQUE SCHOOLS 603 *START*						58	
56	REQUEST CREATED: 12-APR-79 14:25:53						59	
57	FILE: DSK80:PULSAR(13000,30001) CREATED: 12-APR-79 14:15:00 PRINTED: 12-APR-79 14:31:00						60	
58	QUEUE SWITCHES: /FILE:ASCII /COPIES:1 /SPACING:1 /LTHIT:155 /FORMS:NORMAL						61	
59	FILE WILL BE DELETED AFTER PRINTING						62	
60							63	
61							64	
62							65	
63							66	
64							67	
65							68	
66							69	
67							70	

1
2
3
4
5 [PULSAR PROGRAM LOADER - PULSAR]
6 [START OF ASSEMBLY]
7 [END OF PASS 1]
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100



PULSAR - LANGUAGE FOR THE BEBOTHERMENT OF PROGRAMMERS. MACRO %52(551) 14:07 12-APR-79 PAGE 1
PULSAR THR 5-APR-79 13:59

THIS IS THE DEFINITION PART OF PULSAR.

TITLE PULSAR - LANGUAGE FOR THE BEBOTHERMENT OF PROGRAMMERS.
;THIS PROGRAM LOADS PULSAR PROGRAMS INTO CORE.
SUBTTL THIS IS THE DEFINITION PART OF PULSAR
IF1,<PRINTX [PULSAR PROGRAM LOADER = PULSAR]>

;WRITTEN BY SCOTT HALBERT. ALL RIGHTS FOR SOFTWARE RESERVED.
;THE NAME PULSAR IS NOT AN ACRONYM NOR IS IT IN ANYWAY ALLUSIVE TO
;THE FUNCTION OF THE PROGRAM. I CHOSE IT BECAUSE IT SOUNDED NEAT,
;AND BECAUSE IT IS AN ALLUSION TO THE CELESTIAL BODY. THIS LANGUAGE
;IS AN INTERPRETER. THIS IS BECAUSE MY RELIGIOUS AFFLICTIONS DO
;NOT ALLOW ME TO TOUCH COMPLIERS WITHOUT WEARING A RADIATION SUIT.
;THIS LANGUAGE WILL PROBABLY CAUSE INSANITY IN PEOPLE WHO ATTEMPT TO
;USE IT WITHOUT PROPER PSYCHOLOGICAL PREPARATION.
;WARNING: OBJECT CODE CREATED BY THIS PROGRAM COULD GO OFF THE
;DEEP END AT ANY TIME. YOU MAY HAVE TO CALL A PULSAR PSYCHOLOGIST.
;DO NOT CALL ONE WHO HAS NOT BEEN CERTIFIED INSANE.

;THIS PROGRAM IS BEING DEVELOPED FOR THE CLASS OF INDIVIDUAL
;COMPUTER PROJECTS UNDER ALEX PERALTA AT THE CAREER ENRICHMENT
;CENTER, ALBUQUERQUE NEW MEXICO. CONSTRUCTION OF THE PROGRAM WAS
;BEGIN ON APPROX. FEB. 7, 1979. ALL RIGHTS TO THE PROGRAM AND THE
;RESULTING OBJECT CODE ARE THE PROPERTY OF SCOTT HALBERT.
;PROGRAM BECAME OPERATIONAL FOR THE FIRST TIME ON MARCH 13, 1979.

SEARCH QUDSYM ;CALL UP SYMBOLS FROM SYSTEM.

000005 PULWHO=S ;WHO DUN IT
000001 PULMAJ=1 ;MAJOR VERSION NUMBER
000002 PULMIN=2 ;MINOR VERSION NUMBER
000334 PULEDT=334 ;EDIT NUMBER

500201 000334 %%PULS=<PULWHO>B2!<PULMIN>B11!<PULMAJ>B17+PULEDT ;THIS IS WHAT IT IS

000001 A=1 ;ACCUMULATORS
000002 B=2
000003 C=3
000004 D=4
000005 E=5
000006 F=6
000007 G=7
000016 R=16
000017 F=17

;FLAGS AND OTHER VALUES

000001 YLOG=1 ;LOGICAL VARIABLE
000002 XINT=2 ;INTEGER VARIABLE
000003 ZSTR=3 ;STRING VARIABLE
000004 %REA=4 ;REAL
000005 ZPROC=5 ;THIS IS THE VALUE FOR A PROCEDURE

400006 000000 ZPERM=1B0 ;VARIABLE OR PROC IS PERMANENTLY LOCKED
200000 600000 ZLOCK=1B1 ;VAR OR PROC IS SIMPLY LOCKED
100000 000000 ZCHANGE=1B2 ;VAR OR PROC MAY NOT BE REDEFINED

PULSAR = LANGUAGE FOR THE BEEFOTHERMENT OF PROGRAMMERS. MACRO %52(551) 14:07 12-APR-79 PAGE 1-1
PULSAR THR 5-APR-79 13:59
THIS IS THE DEFINITION PART OF PULSAR

```
56      040000 000000  XNOBOU=1B3 ;VAR CANNOT BE BOUND
57      777777  XENDB=777777 ;END OF LIST
58      003000 003000  %XPUL=<3000>;3000>
59      000640  BLK57=<-D256/205>+1 ;THIS IS THE LENGTH OF THE TEMPORARY BUFFERS
60      000100  SIZZ=100 ;THIS IS PRESENT BUFFER SIZE
61      000036  HAZRTG=.RBTIM+1 ;WORDS FOR EXT RIHS
62      000200  BUFG=200 ;THIS IS THE BUFFER SIZE (BLOCK)
63      000002  KILLVL=2 ;WIPE OUT LISTING LEVEL
64      ;0 ABSOLUTELY EVERYTHING WILL OUTPUT
65      ;1 EVERYTHING WILL BE SALLED
66      ;2 LITERAL LISTING WILL BE WIPE OUT
67
68      ;DEFINIE KILL(LEVEL)<
69      ;FILE <LEVEL-KILLVL>,<XLIST>>
70      ;DEFINIE UNKILL.<ALL
71      ;LIST
72      ;IFN KILLVL,<SALL>
73      ;IFN KILLVL,<SALL>
74
75      ;IFN KILLVL,<SALL> ;KILL THE LISTING OF THINGS IF KILLVL IS NOT 0
76
77
78
79
80      260740 000000  OPDEF CALL[PUSHJ P,] ;DEFINIE THIS
81      263740 000000  OPDEF RETURN[POPJ P,] ;AND THIS
82      254000 4000731 OPDEF SKPRET[IJRST POP11] ;THIS
83      261740 000000  OPDEF SAVE[PUSH P,]
84      262740 000000  OPDEF REST[POP P,]
```

PULSAR - LANGUAGE FOR THE BETTERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 2
PULSAR THR S-APR-79 13:59 THIS IS THE START OF THE PROGRAM AND SETUP

SUBTTL THIS IS THE START OF THE PROGRAM AND SETUP

LOC 137 :LOC TO .JBVER
22PULS :SET IT UP

LOC 125 :LOC TO .JBAPR
JRST ALCOR.

TWOSEG
RELOC 400K :GO BACK TO RELOCATABLE

PULSAR: JRST +2 :THIS IS NORMAL ENTRY

% INDIRECT FILES NOT SUPPORTED AT THIS TIME.

RESET :SCREAM AT HIM
ARRAY PDL[1000] :WIPE EVERYTHING
MOVE P,[1000 1000,PDL] :MAKE AN ARRAY FOR PDL
MOVE A,[0,,TOPC] :SET UP THE PUSH-DOWN-LIST
CORE A :SET UP TO SCRUNCH CORE
GET IT
JRST SCRNC :BITCH.
CALL CURMES :GET THE MESSAGE
MOVEI A,AP.ILM!AP.REN :SET UP FOR INTERRUPTS
APNENB A, :SET THEM
SETZH FRCFLG :SET THE PROCEDURE FLAG
SETZM GMORE :ALLOW COMMENTS FOR NOW
SETZM LISTT :SET ALSO FOR LISTING
SETZM TOPF :ZERO THE TOP LEVEL

PULSAR = LANGUAGE FOR THE BEROOTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 3
PULSAR THR 5-APR-79 13:59 THIS IS THE PLACE WHERE THE ORIGINAL FILE IS INPUT

SUBTTL THIS IS THE PLACE WHERE THE ORIGINAL FILE IS INPUT

112
113
114 400016* 201 16 S 00 000515* DOFIL: MOVET R,FILEI_ ;GET THE ADDRESS OF THE LOOKUP BLOCK
115 OUTSTR LASC1Z= ;
116 400017* 051 05 S 00 401374* > F3 CALL IFILE ;PROMPT HIM
117 400020* 260 17 S 00 400050* CLRFI IFILE ;CALL THE INPUT ROUTINE
118 400021* 051 11 S 00 4000194* OPEN A,DEV ;FOL ALL ATTEMPTS TO CONFUSE US
119 400022* 050 01 S 00 4000000 OPEN A,DEV ;OPEN THE DEVICE
120 400023* 254 00 S 00 400150* JRST DVNAVL ;CAN'T. GO COMPLAIN
121 400024* 200 01 S 00 401375 MOVE A,[1B0+TBUF+11] ;SET UP THE BUFFER HEADER
122 400025* 202 01 S 00 4000530 MOVEM A,TB ;STICK IT IN THERE
123 400026* 206 01 S 00 401376 MOVE A,[POINT 7,J] ;AND THE BYTE POINTER FOR IT
124 400027* 202 01 S 00 4000531 MOVEM A,TB+1 ;AND STORE THAT THERE
125 400030* 076 01 S 00 000515* LOOKUP A,FILEI ;AND LOOKUP THE FILE
126 400031* 254 00 S 00 400152* JRST FLNAVL ;COMPLAIN
127 400032* 260 17 S 00 4000523 CALL HEDOU ;OUTPUT THE HEADER
128 400033* 424 05 S 00 4000534 SETZM LINE# ;TO ONE
129 400034* 206 03 S 00 401377 MOVE C,[POINT 7,BUFF1] ;SET UP A POINTER
130 400035* 206 03 S 00 4001546 SETZM PLACE ;ZERO THE COLUMNS
131 400036* 260 17 S 00 4000515 CALL INCHR ;GET A CHARACTER
132 400037* 305 01 S 00 000615 CAIN A,15 ;SEE IF A CHARRETURN
133 CALL [CALL INCHR YES, GO WASTE THINGS
134 SETZ A, ;ZERO THE LAST BYTE
135 400040* 260 17 S 00 401400* RETURN] ;AND GO BACK
136 400041* 136 01 S 00 000003 IDPB A,C ;STICK IT IN THE POINTER
137 400042* 326 01 S 00 400036* JUMPN A,LPY+2 ;SEE IF IT IS ZERO
138 400043* 260 17 S 00 401123* CALL SLOPRC ;GO LOAD THE PROCEDURE
139 400044* 254 00 S 00 400034* JRST LPY ;LOOP BACK
140 400045* 262 17 S 00 4000000 RES ;SAVE THING
141 400046* 260 17 S 00 401260* CALL DOAEND ;PUT AN END STATEMENT HERE
142 400047* 254 00 S 00 4000560 JRST ENDR ;
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168

```

143
144
145 400050 515 01 0 16 00000000
146 400051 541 01 0 16 00000001
147 400052 402 01 0 16 00000016
148 400053 251 01 0 16 00000004
149 400054 201 01 0 00 00000004
150 400055 202 01 0 16 00000000
151 400056 047 01 0 00 00000024
152 400057 255 00 0 00 00000000
153 400058 222 01 0 16 00000001
154 400059 298 01 0 00 401403
155 400060 262 01 0 00 00000000
156 400061 260 01 0 00 401404
157 400062 262 01 0 16 00000002
158 400063 202 01 0 00 401405
159 400064 262 01 0 16 00000003
160 400065 260 17 0 00 400161
161 400066 312 02 0 00 00000000
162 400067 254 02 0 00 400101
163 400068 047 01 0 00 00000000
164 400069 254 00 0 00 400136
165 400070 202 01 0 00 00000005
166 400071 047 01 0 00 00000055
167 400072 200 01 0 16 00000001
168 400073 202 01 0 16 00000001
169 400074 260 17 0 00 400161
170 400075 322 01 0 00 400103
171 400076 293 01 0 16 00000002
172 400077 362 02 0 00 00000056
173 400078 250 00 0 00 00000000
174 400079 260 17 0 00 400161
175 400080 512 01 0 16 00000003
176 400081 302 02 0 00 0000133
177 400082 254 00 0 20 400121
178 400083 260 17 0 00 400203
179 400084 362 02 0 00 00000054
180 400085 254 00 0 00 0000140
181 400086 516 01 0 16 00000001
182 400087 260 17 0 00 400203
183 400088 542 01 0 16 00000001
184 400089 260 17 0 00 400161
185 400090 326 01 0 00 400142
186 400091 302 02 0 00 00000074
187 400092 254 00 0 00 400132
188 400093 260 17 0 00 400203
189 400094 301 01 0 00 00000000
190 400095 313 01 0 00 0000777
191 400096 254 00 0 00 400144
192 400097 137 01 0 00 4001406
193 400098 260 17 0 00 400161
194 400099 326 01 0 00 400142
195 400100 342 02 0 00 00000057
196 400101 254 00 0 00 400135
197

143
144
145 IFILE: HRL7J A,0(R) ;SET UP FOR ZEROING BLT
146 HRR1 A,1(R) ;KEEP SETTING
147 SETZM OR ;GET THE INITIAL ZERO
148 BLT A,,RBRPV(R) ;WIPE IT
149 MOVEI A,,RBRPV ;GET THE ADD OF THE PROT, ETC
150 MOVEM A,,RCNT(R) ;STICK IT IN THE COUNT
151 GETPPN A, ;GET HIS PPN FOR DEFAULT
152 JFCL ;NO OP
153 MOVEM A,,RHPPN(R) ;STICK IT IN
154 MOVE A,TSIXBIT =DSK ;MAKE 'DSK' DEFAULT
155 MOVEM A,DEVY ;STICK IT
156 MOVE A,TSIXBIT =PROG ;MAKE THAT DEFAULT
157 MOVE A,RRBNM(R) ;STICK THAT
158 MOVE A,TSIXBIT =PUL ;STICK THAT
159 MOVEM A,RBEXT(R) ;MAKE THAT DEF
160 CALL GET6 ;STICK THAT
161 GATE R,";" ;GET THE FIRST SIXBIT AND DELIM.
162 JRST GETFL1 ;SEE IF THERE IS A DEVICE
163 DEVNAM A, ;NO, GO GET THE FILE
164 JRST ERR1 ;GET ITS REAL NAME
165 DEVPNM A,DEVY ;NO SUCH DEVICK
166 MOVEM A,RRPPN(R) ;STORE THE REAL ONE
167 DEVPPN A, ;SEE ABOUT THE FPN
168 MOVE A,RRPPN(R) ;GET THE DEFAULT IF NOT DISK
169 MOVEM A,RRPPN(R) ;AND STORE ONE OF THE TWO
170 CALL GET6 ;GET ANOTHER SIX
171 CALL GET6 ;SEE IF IT IS ZERO
172 JRST GETFL2 ;STORE IT IF NOT
173 DEVPNM A,DEVY ;SEE IF IT IS A FILE
174 JRST GETFL3 ;NO IT IS NOT
175 CALL GET6 ;GET THE EXTENSIN
176 HLLZM A,,RHEXT(R) ;STORE IT
177 GETFL1: JUMPE A,GETFL2 ;SEE IF THAT IS A PPN
178 MOVEM A,RRBNH(R) ;NO, GO AWAY
179 GETFL2: CAIE R,";" ;SO, GET THE OCTAL
180 JRST GETFL3 ;IS HE DOING THIS RIGHT?
181 CALL GET6 ;NO, COMPLAIN
182 GETFL3: CAIE R,";" ;STORE IT IN THE PROJECT
183 JRST GETFL4 ;AND GET THE NEXT PART
184 CALL GET6 ;AND STICK IT IN THE PROGRAMMER
185 CALL GET6 ;GET ANOTHER WORD
186 JRST GETFL5 ;COMPLAIN THAT HE HAS STRANGE STUFF
187 JUMPN A,ERR3 ;SEE IF SAY ANYTHING ABOUT PROT.
188 CALL GET6 ;NO, GO SOMEWHERE ELSE
189 CALL GET6 ;GET THE PROT
190 CAIE A,0 ;SEE IF IT IS IN RANGE
191 JRST GETFL5 ;0<=777<=777
192 DBR A,POINT 9,,RBRPV(R),B1 ;IF OUT OF RANGE... SCREAM
193 CALL GET6 ;AND GET SOME MORE
194 JUMPN A,ERR3 ;IF ZERO COMPLAINT
195 CAIE R,";" ;SEE IF IT IS A SHITCH
196 JRST GETFL6 ;NO GO AWAY
197 OUTSTR LASCIZ -
```

2
3
4
5 PULSAR - LANGUAGE FOR THE REBOTHERMENT OF PROGRAMMERS MACRO X52(551) 14:07 12-APR-79 PAGE 4-1
6 PULSAR THR 5-APR-79 13:59
7 THIS IS THE FILE INPUTTER ROUTINE.

8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62

Moore Instruments Inc., Inc.
SPEEDBED WITH DYNBACK PATENT PENDING
FIRE THERMOA
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62

% SWITCHES ARE NOT IMPLEMENTED AT THIS TIME.
 1) GETFL6: RETURN ;COMPLAIN
 ERR1: OUTSTR [ASCIZ] ;THE TYPES THEREON WILL BE IGNORED
 ?SCREWY DEVICE. 1)
 2) ERR2: JRST PULSAR ;COMPLAIN
 ?SCREWY PPN. 1)
 3) ERR3: OUTSTR [ASCIZ] ;COMPLAIN
 ?SCREWY FILESPEC. 1)
 4) ERR4: OUTSTR [ASCIZ] ;COMPLAIN
 ?SCREWY PROTECTION. 1)
 5) JRST PULSAR ;COMPLAIN
 ERR5: OUTSTR [ASCIZ] ;COMPLAIN
 ?SCREWY SWITCH. 1)
 6) JRST PULSAR ;COMPLAIN
 DYNVAL: OUTSTR [ASCIZ] ;COMPLAIN
 2) DEVICE SPECIFIED IS NOT AVAILABLE.
 1) JRST PULSAR ;OUTPUT A COMPLAIN
 FLNAV: OUTSTR [ASCIZ] ;AND GO BACK
 ? LOOKUP ON FILE 1)
 CALL FILOU ;BITCH
 OUTSTR [ASCIZ] ;FAILING WITH A C-1 ;COMPLAIN
 2) HRR7 A,FILEI+,RREXT ;GET THE ERROR CODE
 CALL OCTOU ;OUTPUT IT
 OUTSTR [ASCIZ] ;ERROR CODE.
 3) ;BITCH SOME MORE
 JPST PULSAR ;GO AWAY
 SET2B A,B ;ZERO BOTH A AND B
 MOVE C,[POINT 6,A] ;SET A BUFFER
 MOVEI D,6 ;MAX OF 6
 INCHWL B ;GET A CHAR
 CAIN B," " ;IF SPACE IGNORE
 JRST -2 ;SO GO IGNORE
 GET6B: CAILE A,"Z" ;SEE IF IT IS IN RANGE
 RETURN ;NOPE
 CAIL B,"A" ;SEE AGAIN
 JRST GET6C ;OKAY GO DO IT
 CAIG B,"9" ;IS IT IN RANGE
 CAIGE B,"0" ;HUH?
 RETURN ;NOT IN RANGE
 GET6C: SUB1 B," " ;GET RID OF OFFSET
 SOSL D ;DECREMENT THE DOHICKEY
 IDPB B,C ;STICK THE BYTE
 INCHWL B ;GET ANOTHER CHAR
 JRST GET6B ;AND GO BACK
 4) SET2B A,B ;ZERO BOTH
 MOVEI D,10 ;SIU A 10 AS BASE
 INCHWL B ;GET A LINE
 CAIN B," " ;SEE IF A SPACE
 JRST -2 ;IF SO IGNORE
 CAIG B,"9" ;SEE IF IN RANGE

PULSAR - LANGUAGE FOR THE BROTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 4-2
PULSAR TIR 5-APR-79 13:59 THIS IS THE FILE INPUTTER ROUTINE.

```
253 400211* 365 02 0 00 000060      CAIGE   B,"0"           ;HUH IS IT
254 400212* 263 17 0 00 000000      RETURN  D               ;NO SO GO AWAY
255 400213* 375 20 0 00 000004      SOSGE   D               ;DECREMENT THE MAX
256 400214* 254 00 0 00 000000      JRST    +3              ;BACK UP
257 400215* 222 01 0 00 001473*     IMUL   A,[10]            ;STICK IT
258 400216* 271 01 0 00 000000      ADDI   A,"0"(B)        ;AND ADD IT IN
259 400217* 254 00 0 00 000205*     JRST    GET8+2          ;AND GO BACK THERE
```

PULSAR - LANGUAGE FOR THE BEBOTHERMENT OF PROGRAMMERS. MACRO Z52(551) 14:07 12-APR-79 PAGE 5
 PULSAR THR 5-APR-79 13:59 THESE ARE A GREAT MESS OF OUTPUT ROUTINES

262			SUBTTL THESE ARE A GREAT MESS OF OUTPUT ROUTINES
261	4002220*	261 17 0 00 0000001	ALCOR.: SAVE A ;SAVE A
262	4002221*	200 01 0 00 000000*	MOVE A, JBREL ;GET THE TOP OF LOW
263	4002222*	271 01 0 00 001000	ADDI A,IK ;ADD A LITTLE MORE
264	4002223*	553 01 0 00 0000001	HRRZS A ;WIPE THE LEFT
265	4002224*	047 01 0 00 0000111	CORE A ;REALOC
266	4002225*	250 00 0 00 0000465*	JRST COREPR ;GO BITCH
267	4002226*	200 01 0 00 000000*	MOVE A, JBTPC ;GET THE THING
268	4002227*	291 01 1 01 0002000	MOVEI A,PC(A) ;GET THE EFFECTIVE ADD
269	4002230*	202 01 0 00 0000522*	MOVEM A,ADDRS# ;SAVE IT
270	4002231*	260 17 0 00 0003111*	CALL CORNES ;FROHTER
271	4002232*	262 17 0 00 0000001	RES A
272	4002233*	370 00 0 00 0000226*	SOS JBTPC# ;DECREMENT
273	4002234*	254 00 1 00 000233*	JRST 0,JBTPC ;AND GO BACK
274	4002235*	261 17 0 00 0000002	VERDU: SAVE R ;SAVE R
275	4002236*	200 02 0 00 0000001	MOVE B,A ;PUT THE VER IN B
276	4002237*	135 01 0 00 001474*	LDB A, IPOINT B,B,111 ;GET THE MAJOR VERSION
277	4002240*	260 00 0 00 0000472*	CALL OCTOU ;AND OUTPUT IT
278	4002241*	135 01 0 00 001475*	LDB A, IPOINT S,B,171 ;GET THE MINOR VERSION
279	4002242*	322 01 0 00 0000245*	JUMPE A,+3 ;IF NOT THERE DON OUT IT
280	4002243*	271 01 0 00 0001000	ADDI A, #0 ;ADD THE OFFSET
281	4002244*	260 17 0 00 0000520*	CALL CHROU ;AND OUTPUT
282	4002245*	201 01 0 00 0000050	MOVEI A,"(" ;GET A PAREN
283	4002246*	250 17 0 00 0000520*	CALL CHROU ;AND OUTPUT
284	4002247*	550 01 0 00 0000002	HRRZ A,B ;GET THE EDIT #
285	4002248*	260 17 0 00 0000472*	CALL OCTOU ;AND OUTPUT IT
286	4002249*	261 01 0 00 0000051	MOVEI A,")" ;GET RIGHT PAREN
287	4002250*	260 17 0 00 0000520*	CALL CHROU ;OUTPUT IT
288	4002253*	135 01 0 00 001476*	LDB A, IPOINT 3,B,21 ;GET THE WHO NUM
289	4002254*	320 01 0 00 0000261*	JUMPE A,+5 ;IF ZERO IGNORE
290	4002255*	201 01 0 00 0000055	MOVEI A, #0 ;OUTPUT A =
291	4002256*	260 17 0 00 0000520*	CALL CHROU ;
292	4002257*	135 01 0 00 001476*	LDB A, IPOINT 3,B,21 ;GET IT AGAIN
293	4002260*	260 17 0 00 0000472*	CALL OCTOU ;AND OUTPUT IT
294	4002261*	200 01 0 00 0000002	MOVE A,B ;PUT B BACK IN A
295	4002262*	262 17 0 00 0000002	RES B ;AND GET OLD VALUE OF B
296	4002263*	263 17 0 00 0000000	RETURN ;AND RETURN
297	4002264*	274 01 0 00 001546*	FORAT: SUB A,PLACE ;GET THE PLACES REMAINING
298	4002265*	323 01 0 00 0001400*	JUMPE A,IRETURN ;IF <=0 THEN RETURN
299	4002266*	261 17 0 00 0000001	SAVE A ;SAVE A
300	4002267*	201 21 0 00 0000040*	MOVEI A, " " ;PUT IN A SPACE
301	4002270*	260 17 0 00 0000520*	CALL CHROU ;OUTPUT IT
302	4002271*	262 17 0 00 0000001	RES A ;GET BACK A
303	4002272*	367 01 0 00 0000266*	SOJG A,-4 ;AND LOOP UNTIL 0
304	4002273*	263 17 0 00 0000000	RETURN ;AND RETURN
305	4002274*	261 17 0 00 0000001	SAVE A ;SAVE A
306	4002275*	201 21 0 00 000133*	MOVEI A, " " ;OUTPUT A =
307	4002276*	260 17 0 00 0000520*	CALL CHROU ;
308	4002277*	554 01 0 00 000472*	HLRZ A,(P) ;GET A BACK
309	4002300*	260 17 0 00 000472*	CALL OCTOU ;OUTPUT THE LEFT
310	4002301*	201 01 0 00 0000054*	MOVEI A, " " ;OUT A ,
311	4002302*	260 17 0 00 0000520*	HRRZ A,(P) ;GET A
312	4002303*	550 01 0 00 0000000	CALL OCTOU ;AND OUTPUT THE RIGHT
313	4002304*	260 17 0 00 0000472*	MOVEI A,"1" ;OUTPUT A]
314	4002305*	201 01 0 00 0000135	

PULSAR - LANGUAGE FOR THE BEBOTHERMENT OF PROGRAMMERS. MACRO %52(551) 14:07 12-APR-79 PAGE 5-1
PULSAR THR 5-APR-79 13:59 THESE ARE A GREAT MESS OF OUTPUT ROUTINES

315 400306* 260 17 0 00 400520* CALL CHROU 8
316 400317* 262 17 0 00 000001 RES A 9
317 400310* 265 17 0 00 000000 RETURN 10
318 400311* 332 00 0 00 000541* CORMES: SKIPE NOCMS# 11
319 400312* 263 17 0 00 000000 RETURN 12
320 400313* 261 17 0 00 000001 SAVE A 13
321 400314* 051 03 0 00 401477* OUTSTR FASCIZ* 14
322 400315* 550 01 0 00 400221* (") HRRZ A,,JREL 15
323 400316* 242 01 0 00 777765 LSH A,-12 16
324 400317* 271 01 0 00 000001 ADDI A,1 17
325 400318* 260 17 0 00 400467* CALL DECOU 18
326 400320* 260 17 0 00 400467* OUTSTR [ASCIZ =K] 19
327 400321* 051 03 0 00 401500* (") OUT THE EFFECTIVE ADD 20
328 400322* 262 17 0 00 000001 RES A 21
329 400323* 263 17 0 00 000000 RETURN 22
330 400324* 261 17 0 00 000001 PRDATE: SAVE A 23
331 400324* 261 17 0 00 000001 SAVE R 24
332 400325* 261 17 0 00 000002 SAVE C 25
333 400326* 261 17 0 00 000003 MOVE B,A 26
334 400327* 260 02 0 00 000001 IDIV B,[=D372] 27
335 400328* 238 02 0 00 401501* ADD B,[=D641] 28
336 400331* 270 02 0 00 401502* MOVM B,YEAR# 29
337 400332* 261 02 0 00 000002 ITO THAT 30
338 400333* 260 02 0 00 000003 DIV B,C 31
339 400334* 230 02 0 00 401503* IDIV B,[=D31] 32
340 400335* 350 00 0 00 000003 AOS C 33
341 400336* 260 01 0 00 000005 MOVE A,C 34
342 400337* 260 17 0 00 400467* CALL DECOU 35
343 400340* 201 03 0 00 401504* MOVEI C,[ASCIZ="JAN--FEB--MAR--APR--MAY--JUN--JUL--AUG--SEP--OCT--NOV--DEC="] 36
344 400341* 270 03 0 00 000002 ADD C,R 37
345 400342* 260 01 0 00 000000 MOVE A,(C) 38
346 400343* 261 17 0 00 000013 SAVE 13 39
347 400344* 261 17 0 00 000014 SAVE 14 40
348 400345* 400 14 0 00 000000 SETZ 14 41
349 400346* 260 13 0 00 000001 MOVE 13,A 42
350 400347* 220 16 0 00 000015 MOVT R,13 43
351 400350* 260 17 0 00 400451* CALL STR 44
352 400351* 262 17 0 00 000014 RES 14 45
353 400352* 262 17 0 00 000013 RES 13 46
354 400353* 260 01 0 00 001556* MOVE A,YEAR 47
355 400354* 260 17 0 00 400467* CALL DECOU 48
356 400355* 260 17 0 00 000003 RES C 49
357 400356* 260 17 0 00 000002 RES B 50
358 400357* 262 17 0 00 000001 RES A 51
359 400360* 263 17 0 00 000000 RETURN 52
360 400361* 261 17 0 00 000001 SAVE A 53
361 400362* 261 17 0 00 000002 SAVE B 54
362 400363* 231 01 0 00 001750 IDIVI A,=D1000 54
363 400364* 241 01 0 00 00074 IDIVI A,=D60 55
364 400365* 222 02 0 00 001551* MOVM R,SECS# 56
365 400366* 231 01 0 00 000074 IDIVI A,=D60 57
366 400367* 262 02 0 00 000540* MOVM R,MINS# 58
367 400370* 260 17 0 00 400467* CALL DECOU 59
368 400371* 221 01 0 00 000072 MOVEI A,";" 60
369 400372* 260 17 0 00 400520* CALL CHROU 61
60 61 62 63 64 65 66 67

PULSAR - LANGUAGE FOR THE BEBOOTHERMENT OF PROGRAMMERS. MACRO %52(551) 14:07 12-APR-79 PAGE 5-2
 PULSAR THR S-APR-79 13:59 THESE ARE A GREAT MESS OF OUTPUT ROUTINES

Water Business Fund, Inc.

Patent Pending

SPEEDED UP WHEN BACK

Fox Fabrics

```

    8 370 400373* 200 01 0 00 000540* MOVE A,MJNS ;GET THE MINUTES
    9 371 400374* 200 07 2 00 000467* CALL DECOU ;OUTPUT THE MINUTES
    10 372 400375* 201 01 0 00 000072* MOVEI A,";" ;ANOTHER COLON
    11 373 400376* 200 17 0 00 000520* CALI CHROU ;OUT IT
    12 374 400377* 200 01 0 00 001551* MOVE A,SECS ;GET SECS
    13 375 400400* 200 17 0 00 000467* CALL DECOU ;OUTPUT IT
    14 376 400401* 200 17 0 00 000002* RES R ;RESTORE AC'S
    15 377 400402* 200 17 0 00 000001* RES A
    16 378 400403* 200 17 0 00 000000* RETURN ;GO BACK TO MAIN
    17 379 400404* 200 01 0 00 000105* SAVE A ;SAVE A
    18 380 400405* 200 01 0 00 000105* MOVE A,DEVY ;GET THE DEVICE
    19 381 400406* 200 17 0 00 000430* CALL SIXOUT ;OUTPUT IT
    20 382 400407* 201 01 0 00 000072* MOVEI A,";" ;OUT A COLON
    21 383 400410* 200 17 0 00 000520* CALL CHROU ;SAVE B
    22 384 400411* 201 17 0 00 000002* MOVEI B,FILEI ;GET THE FILE BLOCK
    23 385 400412* 201 02 0 00 000515* MOVE A,RBNAM(B) ;GET THE NAME
    24 386 400413* 200 01 0 02 000002* CALL SIXOUT ;OUTPUT IT
    25 387 400414* 200 17 0 00 000430* MOVEI A,";" ;OUTPUT A
    26 388 400415* 201 01 0 00 000056* CALL CHROU ;GET THE EXTENSION
    27 389 400416* 200 17 0 00 000520* HLLZ A,RBEXT(B)
    28 390 400417* 510 01 0 02 000003* CALL SIXOUT ;OUTPUT IT
    29 391 400420* 200 17 0 00 000430* MOVE A,PPNPN(B) ;GET THE PPN
    30 392 400421* 200 01 0 02 000001* CALL PPNOU ;AND OUTPUT THAT
    31 393 400422* 200 17 0 00 000274* RES B ;GET BACK B
    32 394 400423* 200 17 0 00 000002* RES A ;GET BACK A
    33 395 400424* 200 17 0 00 000001* RETURN ;AND RETURN
    34 396 400425* 200 17 0 00 000000* SIXOU: SETZM JFLG# ;SET THE JUSTIFY FLAG
    35 397 400426* 200 00 0 00 000533* JRST SYDO ;GO DO IT
    36 398 400427* 200 00 0 00 000431* SIXOUT: SETOM JFLG ;OU JUSTIFY
    37 399 400430* 476 00 0 00 000533* SAVE C ;SAVE C
    38 400 400431* 200 17 0 00 000003* SAVE B ;SAVE B
    39 401 400432* 200 17 0 00 000002* SAVE A ;SAVE A
    40 402 400433* 200 17 0 00 000001* MOVEM A,SIX# ;SAVE A WHICH IS THE WORK
    41 403 400434* 200 01 0 00 001552* MOVE B,(POINT 6,SIX) ;MAKE A POINTER
    42 404 400435* 200 02 0 00 000152* MOVEI C,6 ;SET LIM TO SIX CHARS
    43 405 400436* 201 03 0 00 000006* ILDB A,B ;GET THE NEXT CHAR
    44 406 400437* 134 01 0 00 000002* JFLG ;SEE IF THE FLAG SET
    45 407 400440* 332 00 0 00 000533* SKIPE A,B ;IF ZERO JUMP OUT
    46 408 400441* 332 01 0 00 000444* JUMPE A,B+3 ;ADD A SPACE
    47 409 400442* 271 01 0 00 000040* ADDI C,-5 ;OUTPUT THE CHAR
    48 410 400443* 200 17 0 00 000520* CALL CHROU ;LOOP WAY BACK
    49 411 400444* 367 03 0 00 000437* SOJG C,-5 ;GET A BACK
    50 412 400445* 200 17 0 00 000001* RES A ;GET B BACK
    51 413 400446* 200 17 0 00 000002* RES B ;GET C BACK
    52 414 400447* 200 17 0 00 000003* RES C ;AND GO AWAY
    53 415 400450* 200 17 0 00 000000* RETURN ;SAVE R
    54 416 400451* 200 17 0 00 000016* STR: SAVE A ;SAVE A
    55 417 400452* 200 17 0 00 000001* HLL R,(POINT 7,) ;MAKE A POINTER OUT OF IT
    56 418 400453* 500 16 0 00 001376* ILDB A,R ;GET THE FIRST
    57 419 400454* 134 01 0 00 000016* JUMPE A,B+3 ;IF END THEN JUMP OUT
    58 420 400455* 322 01 0 00 000460* CALL CHROU ;OUTPUT THE CHAR
    59 421 400456* 200 17 0 00 000522* JRST -3 ;JUMP BACK
    60 422 400457* 254 00 0 00 000454* RES A ;GET A BACK
    61 423 400460* 200 17 0 00 000001* RES R ;GET R BACK
    62 424 400461* 200 17 0 00 000016*
  
```


SUBTTL BREAK THE LINE UP INTO ITS PARTS

```
;BY THE WAY... ACS ARE NOT SACRED HERE
SERKLI: MOVE A,[POINT 7,HUFF] ;THIS POINTS TO THE ORIGINAL
        MOVE B,[POINT 7,OPCODE] ;THIS TO THE NEW OPCODE
        MOVE D,[POINT 7,DPARGE] ;AND THE DPARGUMENT
        ILDB C,A ;GET EHT FIRST PART
        JUNPE C,WIPOP ;WIPE THAT OPCODE AND ARG
        CAIN C," " ;IGNORE A SPACE
        JRST OPLP ;DO IT
        CAIN C," " ;SEE IF A TAB
        JRST OPLP ;AND IGNORE
        IDPB C,B ;NOW WE KNOW ... STICK IT
        ILDB C,A ;GET ANOTHER ...
        JUNPE C,WIPOP ;WIPE THA ARGUMENT
        CAIN C," " ;SEE IF A SPACE
        JRST DALP ;GO DO ARGMENT
        CATN C," " ;SEE IF A TAB
        JRST DALP ;AND GO DO ARGMENT
        CLTPU ;IS STILL OPCODE
        JRST CLTPU ;GET THE NEXT
        DALP: ILDB C,A ;WIPE THE ARG AGAIN
        JUNPE C,WIPOP ;IS A SPACE?
        CAIN C," " ;IGNORE
        JRST DALP ;IS A TAB?
        CAIN C," " ;IGNORE THAT TO
        JRST DALP ;STICK THAT ON THE ARG
        ARGPU: IDPB C,D ;AND GET A CHAR
        ILDB C,A ;AND PUT THE ZERO ON THE END OF EVERY
        JUNPE C,WIPOP ;AND LOOP BACK FOR ANOTER
        JRST ARGPU ;AND STICK THE ZERO
        WIPOP: SETZ C, ;STICK ON THE OPCODE
        IDPB C,D ;STICK ON THE ARGUMENT
        IEPB C,D ;AND RETURN
```

More Business Fixes, Inc.
 PATENT PENDING
 SPEEDLOAD WITH EYETRACK
 JOHN LABROO

SUBTTL VARIABLE AND PROCEDURE MANIP FUNTION

;COPY THE STRING POITED TO BY B AND WRITE OVER A
 COPY: SAVE A ;SAVE THE POINTER TO BE SET
 SAVE B ;AND SAVE THE OLD POINTER
 SAVE C ;AND SAVE THE TEMPORARY
 SAVE D ;SAVE ANOTHER
 SETD D, ;SET A FLAG

HLL A,[POINT 7,1] ;MAKE IT A POITNER
 HLL B,[POINT 7,1] ;AND MAKE THAT ONE

CFP: ILDR C,A ;GET A BYTE
 IDPB C,A ;AND STICK IT

JUMPE D,+4 ;SKIP OVER
 CAIN C,0 ;SEE IF IT IS ZERO

SETZM EA ;IF SO SET THE WHOLE WORD
 SETZ D, ;AND RESET THE FLAG

JUMPN C,CPP ;IF NOT END DO IT AGAIN

RES D, ;RESTORE IT
 RES C, ;GET BACK C

RES A, ;GET A

RETURN ;RETURN

;IT WILL FIND THE NEXT VAR AFTER A AND PUT BACK IN A

NEXTV: SAVE B ;SAVE B
 MOVM A,SVLST# ;SAVE THE LAST NAME
 HRRZ A,EA ;WIPE OUT THE LEFT SIDE AND
 RES B, ;STICK IT IN A AND RESTORE B
 RETURN ;AND RETURN

;THIS WILL TAKE VAR PT BYY A AND PUT NAME ADD IN B

NAMEV: HLRZ B,EA ;GET THE POINTER TO THE STRING
 HRRZ B,EB ;AND LOOP THE FIND
 RETURN ;AND RETURN

;THIS SHOULD COMPARE C AND B AND SKIP IF THEY ARE NOT EQUAL

CMPRE: SAVE B ;SAVE THE FIRST
 SAVE C ;SAVE THE SECOND

SAVE A ;SAVE A TEMPOR
 SAVE D ;SAVE ANOTHER

HLL B,[POINT 7,1] ;MAKE A POINTER
 HLL C,[POINT 7,1] ;MAKE ANOTHER

SETZM FLGE# ;MAKE A FLAG

CMPLP: ILDR A,B ;GET THE FIRST BYTE FROM THE FIRST
 IDPB D,C ;GET A BYTE FROM THE LAST

CAME D,A ;COMPARE THEM

SETOM FLGE# ;SET THE FLAG IF NOT

JUMPN A,CMPLP ;GO THERE IF NOT END

RES D, ;RES D

RES A, ;AND A

RES C, ;AND C

RES B, ;AND B

SKIPN FLGE# ;18K1P IF FLAG SET

PULSAR - LANGUAGE FOR THE BEDOTHERMENT OF PROGRAMMERS, MACRO Z52(551) 14:07 12-APR-79 PAGE 7-1
 PULSAR THR 5-APR-79 13:59 VARIABLE AND PROCEDURE MANIP FUNKTIONEN

```

    623 400727: 263 17 0 00 000000 RETURN
    624 400730: 254 00 0 00 400731, ;FLAG SET THEREFORE ERROR RETURN
    625 ;MAKE NORMAL REENTRY
    626
    627 ;THIS WILL DO A SKIP RETURN
    628 400731: 262 17 0 00 000000 POP1: RES
    629 400732: 271 00 0 00 000001 ADDI 1 ;GET OLD ADD
    630 400733: 261 17 0 00 000001 SAVE ;ADD 1 TO IT
    631 400734: 263 17 0 00 000000 RETURN ;AND PUT IT BACK
    632
    633 ;THESE WILL ZIP THROUGH THR LISTS AND FIND A VAR OR PRC NAMED
    634 ;B AND PUT THE POINTER IN A
    635
    636 400735: 201 00 0 00 003643, FINDVR: MOVEI 0,VARLIS ;SET TI TO FIND A VARIABLE
    637 400736: 254 00 0 00 400740, JRST +2 ;JUMP OVER PROC
    638 400737: 201 00 0 00 003642, FINDPR: MOVEI 0,PRCLIS ;FIND A PROCEDURE
    639 400740: 212 00 0 00 001554, MOVEM 0,SVOLG# ;SAVE THE POINTER HERE
    640 400741: 200 00 0 00 000000 MOVE 0,00
    641 400742: 261 17 0 00 000002 SAVE A
    642 400743: 261 17 0 00 000003 SAVE C
    643 400744: 261 17 0 00 000004 SAVE D
    644 400745: 200 01 0 00 000000 MOVE A,0 ;GET THE REAL POINTER
    645 400746: 306 01 0 00 777777, GAIN :SAVE THE POINTER
    646 400747: 254 00 0 00 400773, JRST :SAVE THE TEMPOR
    647 400756: 276 03 0 00 000002, FINDIN: MOVE C,B ;SAVE ANOTHER
    648 400751: 260 17 0 00 000003, CALL NAMEV ;GET THE POINTER
    649 400752: 260 17 0 00 400765, CALL CPRE ;SEE IF THAT IS IT
    650 400753: 254 00 0 00 400762, JRST RETFIN ;GET IT GO OUT
    651 400754: 250 00 0 00 400755, JRST NOTHER ;NEED ANOTHER
    652 400755: 260 17 0 00 000076, NOTHER: CALL NEXTV ;GET THE NEXT POINTER
    653 400756: 306 01 0 00 777777, GAIN A,ZENDB ;SEE IF NO MORE
    654 400757: 254 00 0 00 400764, JRST NONFND ;NONE THERE
    655 400758: 200 02 0 00 000005, MOVE BC ;GET THE OLD NAME FOR AGAIN
    656 400759: 254 00 0 00 400756, JRST FINDIN ;GO DO IT AGAIN
    657 400760: 200 02 0 00 000006, RETFIN: SETZM NONFN# ;SET THE FIND ONE
    658 400762: 200 00 0 00 000002, JRST +2 ;JUMP OVER THE SET
    659 400763: 254 00 0 00 400765, NONFND: SETOM NONFN ;SET THE FIND FLAG
    660 400764: 276 00 0 00 000004, RES D ;RESTORE ALL THE GARBAGE
    661 400765: 262 17 0 00 000003, RES C
    662 400766: 262 17 0 00 000002, RES B
    663 400767: 252 17 0 00 000000, SKIPE NONFN ;IF IT IS SET DON SKIP
    664 400770: 332 00 0 00 000542, RETURN ;ERROR RETURN
    665 400771: 253 17 0 00 000000, SKIPRET ;RETURN NORMAL
    666 400772: 254 00 0 00 400731, MAKELS: MOVE SVOLG ;GET THE OLD POINTER
    667 400773: 200 00 0 00 0001554, MOVEM SVLST ;AND SAVE IT IN THE LAST WORD
    668 400774: 200 00 0 00 001553, JRST NONFND ;AND GO REPROCESS
    669
    670 ;THIS APPENDS A AND B AND PUT IN C
    671
    672 400775: 261 17 0 00 000001 ADDSTR: SAVE A ;SAVE A
    673 400777: 261 17 0 00 000002 SAVE B ;SAVE B
    674 401000: 261 17 0 00 000003 SAVE C ;SAVE C
    675 401001: 261 17 0 00 000004 SAVE D ;AND A TEMP
    676 401002: 500 01 0 00 4001376, HLL A,IPOINT 7,1 ;MAKE A POINTER OUT OF EVERYTHING
    677 401003: 500 02 0 00 4001376, HLL B,IPOINT 7,1

```

PULSAR - LANGUAGE FOR THE BROTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 7-2
PULSAR THR 5-APR-79 13:59 VARIABLE AND PROCEDURE MANIP FUNCTION

678 4010066 500 03 0 00 001376^{*} JAHLP: HLL C, [POINT 7,1] ;WE ARE GOING TO DO ASSES\$CS HERE
679 4010055 134 04 0 00 000002 TLDI D,B,^{*} ;GET THE FIRST CHAR
680 4010065 322 04 0 00 001011 JUMPE D,NEXTJM ;THAT ONE IS DONE
681 4010077 136 04 0 00 000001 IDPR D,A ;STICK IT ON A
682 4010100 254 04 0 00 001005 JRST JAMLP ;AND GET ANOTHER
683 4010111 134 04 0 00 000003 NEXTJM: ILDB D,C ;AND GET THAT ONE
684 4010122 136 04 0 00 000001 TDPB D,A ;AND STICK IT
685 4010133 326 04 0 00 001011 JUMPN D,NEXTJM ;OKAY, GOT THAT ONE
686 4010144 262 17 0 00 000004 RES D ;RESTORE ALL THAT SHIT
687 4010155 262 17 0 00 000003 RES C
688 4010166 262 17 0 00 000002 RES B
689 4010177 262 17 0 00 000001 RES A
690 4010200 263 17 0 00 000000 RETURN ;AND RETURN

;RETURNS THE LENGTH OF A AND PUT IN B

691 4010211 261 17 0 00 000001 LENSTR: SAVE A ;MAKE A POINTER
692 4010222 261 17 0 00 000003 SAVE C ;SAVE AC'S
693 4010233 320 02 0 00 000000 SETZ B,^{*} ;ZERO THE COUNTER
694 4010244 500 01 0 00 001376^{*} HLL A, [POINT 7,1] ;MAKE A POINTER
695 4010255 322 03 0 00 000001 ILDB C,A ;GET A WORD
696 4010266 322 03 0 00 001030^{*} JUMPE C,+2 ;SEE IF ZERO
697 4010277 344 02 0 00 001025^{*} AOJA B,-2 ;INCREMENT THE COUNTER
698 4010300 262 17 0 00 000003 RES C ;GET THOSE BACK
699 4010311 262 17 0 00 000001 RES A ;AND THAT COUNT IS IN B
700 4010322 263 17 0 00 000000 RETURN ;AND RETURN

;THIS WILL TAKE THE NAME IN A AND MAKE A PROCEDURE CALLED THAT

701 4010333 261 17 0 00 000001 MAKPRC: SAVE A ;SAVE ALL OF THE VITAL AC'S
702 4010344 261 17 0 00 000002 SAVE B
703 4010355 261 17 0 00 000003 SAVE C
704 4010366 261 17 0 00 000004 SAVE D
705 4010377 262 01 0 00 001550^{*} MOVEM A,PTRSV# ;SAVE THE NAME POINTER
706 4010400 261 02 0 00 001455^{*} MOVT B,[0] ;TELL IT THAT IT HAS A NUM NAME
707 4010411 260 17 0 00 000737^{*} CALL FINDPR ;CALL THE PROC FINDER
708 4010422 255 00 0 00 000000 JFCL^{*} ;IF HE FOUND IT OR NOT
709 4010433 260 04 0 00 001553^{*} MOVE B,SVLST# ;GET THE LAST POINTER
710 4010444 500 01 1 00 000004 HLL A,0D ;GET THE OLD LEFT
711 4010455 500 01 1 00 003641^{*} HRR A,UNUSED ;CHANGE IT TO THE TOP OF CORE
712 4010466 262 01 1 00 000624^{*} MOVEM A,0D ;PUT IT BACK
713 4010477 260 01 0 00 001550^{*} MOVE A,PTRSV ;GET THE OLD POINTER
714 4010498 260 17 0 00 000001 CALL LENSTR ;SEE HOW LONGS IT IS
715 4010509 271 02 0 00 000001 ADDI B,1 ;TAKE CARE OF THE NULL BYT
716 4010510 271 02 0 00 000005 IDIVI B,5 ;HOW MANY WORDS
717 4010521 271 02 0 00 000005 CATE C,0 ;SKIP IF NO REMAINDER
718 4010532 271 02 0 00 000000 ADDI B,1 ;ADD ONE TO THE WORDS
719 4010543 271 02 0 00 000001 MOVEM B,WRDLN# ;SAVE THAT
720 4010554 271 02 0 00 000001 MOVE D,UNUSED ;GET THE LAST ADDRESS
721 4010565 271 02 0 00 000001 HRLI A,(0) ;GET THE POINTER
722 4010576 271 02 0 00 000002 HRR1 A,ZENDR ;SET THE RIGHT TO AN END
723 4010587 271 02 0 00 000002 MOVEH A,0D ;AND STICK IT
724 4010598 271 02 0 00 000001 HRFZ1 A,ZENDR ;AN END
725 4010609 271 02 0 00 000001 MOVEM A,(0) ;STORE THAT TOO
726 4010610 271 02 0 00 000001 HRL A,WRDLN ;GET THE LENGTH

PULSAR - LANGUAGE FOR THE REBOTHERMENT OF PROGRAMMERS
PULSAR THR 5-APR-79 13:59 MACRO XSP(551) 14:07 12-APR-79 PAGE 7-3
VARIABLE AND PROCEDURE NAME FUNCTION

733 401065: 541 01 0 04 000003 HRRI A,3(D) ;GET THE ADDRESS
734 401066: 282 01 0 04 000002 MOVEM A,2(D) ;STICK THAT TOO
735 401067: 261 01 0 04 000003 MOVEI A,3(D) ;GET THAT
736 401070: 200 02 0 00 00015500 MOVE B,PTRSV ;THE THE OLD NAME PONTER
737 401071: 261 17 0 00 00015500 CALL COPY ;AND PUT IT
738 401072: 200 01 0 00 00015500 MOVE A,MRDN ;GET THE LENGTH
739 401073: 271 01 0 00 000004 ADDI A,4 ;ADD THE POINTERS
740 401074: 272 01 0 00 00036400 ADDM A,UNUSED ;AND MAKE A NEW TOP PONTER
741 401075: 262 17 0 00 000004 RES D ;RESTORE ALL THOSE
742 401076: 262 17 0 00 000003 RES C
743 401077: 262 17 0 00 000002 RES B
744 401100: 262 17 0 00 000001 RES A
745 401101: 263 17 0 00 000000 RETURN ;AND GO TO HEL

;THIS WILL GET THE LAST IN THE BINDING LIST

747 LASTAD: SAVE C ;SAVE THE TEMP AC'S
748 750 401103: 261 17 0 00 000003 SAVE B
751 401104: 261 17 0 00 000002 LASENT: MOVET B,OLDNAM ;GET THE NAME
752 401105: 260 17 0 00 00023200 CALL FINDPR ;FIND THE ADDRESS OF THAT
753 401106: 254 00 0 00 000737 CALL JRST L,SMKIT ;CREATE IT IF NOT FOUND
754 401107: 356 00 0 00 000001 AOS 1 ;ADD ONE TO GET ON THE RIGHT LIST
755 401108: 552 01 0 00 0000531 LPLTE: HRRZM A,LSLC# ;SAVE THIS ONE ALSO
756 401111: 550 01 1 00 0000021 HRRZ A,PA ;GO INTO A LOOP UNTIL YOU FIND THE
757 401112: 302 01 0 00 777777 CAIE A,XENDB ;END POINTER AND RETURN THE LAST
758 401113: 254 00 0 00 4011110 JRST LPLTE ;LOOP BACK FOR ANOTHER
759 401114: 550 01 0 00 0000536 RETIT: HRRZ A,LSLC ;GET THE GOOD ADDRESS
760 401115: 262 17 0 00 000002 RES B ;RESTORE ALL THE USED AC'S
761 401116: 262 17 0 00 000003 RES C ;THAT TOO
762 401117: 263 17 0 00 000000 RETURN ;RETURN TO MAIN PROG
763 401120: 271 01 0 00 0002320 LSMKIT: MOVEI A,OLDNAM ;GET THE NAME
764 401121: 260 17 0 00 401033 CALL MAKPRC ;CALL THE PROCEDURE MAKER
765 401122: 254 00 0 00 401100 JRST LASENT ;AND GO DO IT OVER AGAIN

SUBTL THIS LOADS PROCEDURES

;ACS ARE NOT SACRED

```

    SLOPRC: CALL SERKLI ;BREAK THE LINE UP
    MOVEI A,TMPBF ;THIS IS A TEMPORY
    MOVEI R,BUFF ;THIS IS THE OLD
    CALL COPY ;MAKE A COPY
    AOS LINE# ;INCREMENT LINE #
    MOVEI B,IASCIZ "PROCEDURE" ;SEE IF THAT IS A PROC
    MOVEI C,OPCODE ;GET THE OPCODE STRING
    CALL CMPRE ;SEE IF THEY MATCH
    JRST DOAPRC ;GO DO THE PROCEDURE
    MOVEI B,IASCIZ "END" ;SEE IF IT IS AND END
    CALL CMPRE ;AND CHECK THE STRING
    JRST DOAEND ;GO THERE
    MOVEI B,IASCIZ "TYPE" ;TYPE A COMPILE MESSAGE
    CALL CMPRE ;SEE IF IT SAY THAT
    JRST TYPEME ;TYPE AN INTERMEDIATE MESSAGE
    MOVEI B,IASCIZ "IGNORE" ;DO HE WANT TO IGNORE COMMENTS
    CALL CMPRE ;HUM
    CAIA ;DO THIS
    JRST +3
    SETOM GNORE# ;SET IT UP
    RETURN ;AND DO THAT
    MOVEI B,IASCIZ "HEED" ;YES HE DO
    CALL CMPRE ;DOES HE WANT COMMENTS
    CAIA ;SKIP
    JRST +3
    SETZM GNORE ;AND SKIP THAT
    RETURN ;YES HE DOES
    MOVEI B,IASCIZ ";" ;GO HOME DO NOT PASS ETC
    CALL CMPRE ;IS HTAT A COMMENT
    SKIPE @A ;IF ZERO THAT IS CONSIDERED A COMMENT
    CALL CMPRE ;HGH
    CAIA ;YES IT IS
    JRST +3
    SKIPE GNORE ;NOT A COMMENT
    RETURN ;DO HE WANT IT IGNORED
    MOVEI B,IASCIZ "LISTON" ;YES HE DO
    CALL CMPRE ;DOES HE WANT A LISTING
    CAIA ;SEE
    JRST +3
    SETZM LISTIT# ;YES HE DOES WANT IT
    RETURN ;SEE
    MOVEI B,IASCIZ "LISTOFF" ;DOES HE NOT
    CALL CMPRE ;NO
    CAIA ;NO
    JRST +3
    SETOM LISTIT# ;YES HE DOES
    RETURN ;AND SEE IF THAT IS IT
    MOVEI B,IASCIZ "TOPT" ;DEFINE THE TOP LEVEL FUNCITON
    CALL CMPRE ;AND SEE IF THAT IS IT
    JRST TOPLVL ;IF SO GO DO IT
    SKIPE PROFLG ;HE GOT A PROC - GOOD
    JRST YESPRC ;YES HE HAS ONE
    OUTSTR IASCIZ -

```

PULSAR - LANGUAGE FOR THE BEBOTHERMENT OF PROGRAMMERS, MACRO X52(551) 14:07 12-APR-79 PAGE 8-1
PULSAR THU 5-APR-79 13:59 THIS LOADS PROCEDURES

2 ATTEMPT TO PUT CODE WHERE THE SUN (PULSAR WHATEVER) DON'T SHINE.

821 401206: 051 03 0 00 401711: **^1**
822 401207: 254 00 0 00 400545:
823 401210: 254 00 0 00 401271:
YESPRC: JRST FATALE ;HITCH AND BOMB
YESPRC: DOPRT ;IGO DO IT

827 401211: 051 03 0 00 402150:
828 401216: 051 03 0 00 401357:
829 401213: 263 17 0 00 000000:
TYPEME: OUTSTR OPARG ;OUTPUT THE ARGUMENT.
OUTSTE CRLF ;AND A CRLF
RETURN ;AND ELSEWISE IGNORE THE STATEMENT

831 401214: 201 01 0 00 002000: **TOPLVL:** MOVEI A,BUFF ;NEED TO BREAK IT UP AGAIN

832 401215: 201 02 0 00 002150: MOVET B,OPARG ;STICK THAT

833 401216: 260 17 0 00 400614: CALL SBRKLI ;BREAK IT UP

834 401217: 201 02 0 00 002150: MOVEI B,OPARG ;SAVE THAT

835 401218: 201 01 0 00 002554: MOVEI A,TOPF ;AS THE TOP LEVEL

836 401221: 260 17 0 00 400653: CALL COPY ;DO THAT

837 401222: 201 01 0 00 001200: MOVEI A,BUFF ;WE NEED TO RESOTE THAT

838 401223: 201 02 0 00 002404: MOVEI B,TRPBF ;FROM THAT

839 401224: 260 17 0 00 400653: CALL COPY ;AND THAT

840 401225: 263 17 0 00 000000: RETURN ;AND EXIT

841 401256: 082 00 0 00 601547: DOAPRC: SETZM PROGEN# ;IMPLICIT END

842 401227: 336 00 0 00 602234: SKIPN PROFLG ;SEE IF HE HAS NOT CLOSED LAST PROC

843 401230: 254 00 0 00 401236: JRST YESDUN ;YES HE HAS

844 401231: 476 00 0 00 601547: SETOM PROCNH ;IMPLICIT END

845 OUTSTR TASCIZ - ;INSERTED AT END OF MODULE **^1**

846 401232: 051 03 0 00 401730: % IMPLICIT END INSERTED AT END OF MODULE **^1**

847 401233: 051 03 0 00 002234: OUTSTR PROFLG ;OUTPUT THE OLD

848 OUTSTR LASCIZ - ;OUTPUT THE OLD

849 401234: 051 03 0 00 401371: **^1** ;PROPER ENGLISH PLEASE

850 401235: 254 00 0 00 401260: YESDUN: JRST DDAEND ;AND GO TO HELL

851 401236: 051 01 0 00 002000: MOVEI A,BUFF ;GOING TO OVERWRITE IT

852 401237: 260 02 0 00 002150: MOVEI B,OPARG ;THIS IS THE LAST PART

853 401240: 260 17 0 00 400653: CALL COPY ;AND WRITE IT OVER

854 401241: 200 02 0 00 000001: MOVEI B,A ;NEED A COPY OF THIS TOO

855 401242: 201 01 0 00 002470: MOVEI A,OLDPRO ;STORE THAT ARG FOR THE END

856 401243: 260 17 0 00 400653: CALL SBRKLI ;AND MAKE IT

857 401244: 260 17 0 00 400614: MOVEI I ;AND SPLIT IT AGAIN

858 401245: 201 00 0 00 000001: MOVEI LINE# ;THIS IS FIRST LINE #

859 401246: 260 00 0 00 000534: MOVEI A,PECFLG ;AND IT GOES IN THERE

860 401247: 260 00 0 00 000534: MOVEI B,OPCD ;TIS OKAY SO FAR

861 401251: 260 01 0 00 002234: MOVEI CALL COPY ;GET THAT TOO

862 401252: 260 01 0 00 002320: MOVEI A,OLDNAM ;SAVE THAT

863 401253: 260 17 0 00 400653: MOVEI A,OPCD ;SAVE THE NAME

864 401254: 201 01 0 00 002464: CALL COPY ;AND SET A TO OPCD

865 401255: 260 02 0 00 002464: MOVEI A,BUFF ;PUT IT BACK

866 401256: 260 17 0 00 400653: CALL DOPRT ;AND DO IT

867 401257: 258 00 0 00 002234: DOAEND: SKIPE PROFLG ;PRINT IT OUT

868 401258: 332 00 0 00 002234: JRST OKAYGO ;SEE IF A FUNCTION IN PROGRESS

869 401259: 254 00 0 00 401263: RETURN OKAYGO ;TIS OKAY NOW.

870 401260: 254 01 0 00 000000: OKAYGO: SETZM PROFLG ;ZERO THE POINTER

871 401261: 263 17 0 00 000000: MOVEI A,BUFF ;SAVE THIS

872 401262: 082 00 0 00 002234: MOVEI B,TASCIZ PEND ;NEED A THIS

873 401263: 201 01 0 00 002470: MOVEI C,OLDPRE ;ADDED TO THIS

1
2
3
4
5 PULSAR - LANGUAGE FOR THE HEROTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 8-3
6 PULSAR THR 5-APR-79 13:59 THIS LOADS PROCEDURES
7
8 931 481353* 262 17 0 00 000003
9 932 481354* 262 17 0 00 000002
10 933 481355* 262 17 0 00 000001
11 934 481356* 263 17 0 00 000000

RES C
RES B
RES A
RETURN ;AND END

PULSAR = LANGUAGE FOR THE BEDOTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 9
PULSAR THR 5-APR-79 13:59 THIS IS THE END OF THE HISEG. LAST MINUTE DEFS AND LITERALS

935
936
937 401357' 015 012 000 00000
938
939
940
941 401747'

SUBTL THIS IS THE END OF THE HISEG. LAST MINUTE DEFS AND LITERALS

CRLF: BYTE (7)15,12,0 ;THAT IS A CRLF

KILL(2)
UNKILL
TOPHEE.

PULSAR - LANGUAGE FOR THE REBOTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 10
PULSAR THR 5-APR-79 13:59 LOWSEGMENT

SUBTTL LOWSEGMENT

```
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995

RELOC 0
XPST: SKIPA R,[0] ;START UP
SETO R,-1 ;SET THE FLAG TO -1 IF START
MOVEI R,CCL# ;THIS IS THE CCL START
MOVEI R,PULSEI ;THIS IS THE OPSYSTEM
GETSEG R, ;CALL UP THE OPSYSTEM
DATA ;SKIP THE RETURNS
JRST %PULSE ;THIS IS THE HIGH SEG START FOR THAT
OUTSTR FASCIZ = ;LEAVE FROM HERE.

? OH MY GOD... I CANNOT FIND THE OPERATING TIME SYSTEM...
PANIC HORRORS AHHHHH
PLEASE DO SOMETHING. I CANNOT STAND THIS; I AM GOING CRAZY...?
PROGRAM WENT INSANE AT USER HC 003456
;)

MONRT SIXBIT =ALL= ;LEAVE FROM HERE.
PULSEI: SIXBIT =PULSE= ;LEAVE FROM HERE.
BLOCK 2
XPUL: %PUL BLOCK 2
XPUSE=000012
ZAHF: MOVEI A,[0] ;SET A ZERO NAME
GETSEG A, ;KILL THE HIGH SEGMENT
JFCI A, ;WHO CARES
MOVE A,NMSV ;TAKE THE NAME
SETNAM A, ;AND SET IT
MONRT ;EXIT
JRST %PST ;START IT UP
LIT
```



5 PULSAR - LANGUAGE FOR THE BEHOTHERMENT OF PROGRAMMERS. MACRO Z52(S51) 14:47 12-APR-79 PAGE 10-1
6 PULSAR THR 5-APR-79 13:59 LOWSEGMENT

NONDISMISSIVE FNS, INTR, I
SPECIFIC WITH FILEBACK FAULT PENDING
PAGE 1000A

8 996 000057 164 158 151 156 147
9 997 000060 156 048 048 111 048
10 998 000061 143 141 156 156 157
11 999 000062 164 040 163 164 141
12 1000 000063 156 144 040 164 150
13 1001 000064 151 163 054 040 111
14 1002 000065 040 141 155 048 147
15 1003 000066 157 151 156 147 040
16 1004 000067 143 162 141 172 171
17 1005 000070 056 056 056 001 001
18 1006 000071 001 001 001 077 015
19 1007 000072 012 001 001 001 001
20 1008 000073 001 001 077 120 162
21 1009 000074 157 147 162 141 155
22 1010 000075 048 167 145 156 164
23 1011 000076 040 151 156 163 141
24 1012 000077 156 145 040 141 164
25 1013 000100 040 165 163 145 162
26 1014 000101 040 126 103 048 060
27 1015 000102 060 053 064 065 066
28 1016 000103 015 012 000 000 000
29 1017 000104 000 000 000 000 000
30 1018 000105 40 63 53 00 00 00
31 1019 000106 000543 000530
32 1020
33 1021 000107 000 00 0 00 000000
34 1022 000110 000200 000313
35 1023 000111
36 1024 000312 000 00 0 00 000000
37 1025 000313 000200 000110
38 1026 000314
39 1027 000315
40 1028 000326
41 1029 000155
42 1030 000200
43 1031 000202
44 1032 000204
45 1033 000215
46 1034 000234
47 1035 000232
48 1036 000240
49 1037 000247
50 1038 000258
51 1039 0002554
52 1040 0002640 000 00 0 00 000000
53 1041 0002640
54 1042 0002641
55 1043 0002641
56 1044 0002641 000000 003640
57 1045 003641 000000 003640
58 1046 003642 000000 777777
59 1047 003643 000000 777777

DEV: IOASL FDSK² ;ASCIT LINE MODE
DEVY: SIXBX² ;THAT IS DEFAULT
OB, IB ;THESE ARE BUFFERS
ARRAY OB[3], IB[3] ;PUT THEM HERE
TBUF: Z ;BUFG, TBF2+1
BLOCK BUFG+1
TBF2: Z ;BUF2, TBF2+1
BLOCK BUFG+1
FILE1: FILE1, RBPV+1
VAR
BLOCK 3 ;RELOC 2000
BUFF: BLOCK BLKSZ ;LINE BUFFER
OPCODE: BLOCK BLKSZ ;OPCODE BUFFER
OPARG: BLOCK BLKSZ ;ARGUMENT BUFFER
PFCFLG: BLOCK BLKSZ ;PRESENT PROCEDURE TO HANDLE
OLDNAM: BLOCK BLKSZ ;THIS IS THE OLD NAME
TMPRF: BLOCK BLKSZ ;TEMPORARY BUFFER
OLDPRC: BLOCK BLKSZ ;PLACE TO STORE THE PROC THING
;ALL LOCS FROM DEV TO TBF2-1 ARE AVAILABLE FOR USE IN OTHER PROGS
;THEY WILL BE ZEROED FOR SIZE CONSUMPTION
TOPF: BLOCK BLKSZ ;TOP LEVEL BUFF
NMSV: Z ;THIS IS WHERE THE SUPERMOD IS HELD
TBK: BLOCK 1K ;SPACE FOR TBLOCKS

UNUSED: TOPC
PROCLIS: XENDR
VARLIS: XENDR



PULSAR - LANGUAGE FOR THE BEBOTHERMENT OF PROGRAMMERS. MACRO X52(551) 14:07 12-APR-79 PAGE 11
PULSAR THR 5-APR-79 13:59 THIS IS THE FINAL END OF PROGRAM

SUBTTL THIS IS THE FINAL END OF PROGRAM

```
TOPCE
IF1,<PRINTX [END OF PASS 1]>
IF2,<PRINTX [END OF PASS 2]>
END PULSAR
```

NO ERRORS DETECTED

HI-SEG. BREAK IS 401747
PROGRAM BREAK IS 003640
CPU TIME USED 00:18.884

12K CORE USED

Month Entered For. Int'l.	Patent Pending												Sponsored Whiteback		
	PCT				PCT & National				National				SUSP/ABN/		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
KILLVL	66*	77	938	939											5
LASENT	7512	765													6
LASTAD	749*	904													7
LENSTR	694*	720	907												8
LINE	126*	128	494		773*	775	859*	859	877	881					9
LISIT	110	827*	807	813*	813	878									10
LFLTE	755*	758													11
LIFY	129*	137	139												12
LSLC	755*	755	759												13
LSMK1T	753	763*													14
LSTICK	912*	912	921	927											15
MAKELS	646	666*													16
MAKPAC	707*	764													17
MHZT16	63*														18
MINS	366*	366	370												19
NAMEV	600*	648													20
NEXTJM	688	683*	685												21
NEXTTV	592*	652													22
NMSV	505	967	1041*												23
NOCHS	418*	318													24
NOTINAP	409	497*													25
NOLSPFT	879	891*													26
NONEN	657*	657	659	663											27
NONFND	654	659*	668												28
NOTHER	651	652*													29
OALP	558	552	554*	557	559										30
OB	1019	1020*													31
OCTOU	225	277	285	293	309	313	437*								32
OKAYGO	870	872*													33
OLDNAM	751	763	863	880	1035*										34
OLOPRC	855	875	1037*												35
OPAKE	629	827	832	834	852	1033*									36
OPCGD	538	775	861	1032*											37
OPLPL	540*	543	545												38
P	43*	80	81	83	84	101	308	312	884						39
PDL	106*	101													40
PLACE	130	297	467*	467											41
POPI	882	628*													42
PPNOL	305*	393													43
PRCENH	841*	841	844	891											44
PRCFLG	128	489	492	818	842	847	860	869	872	1034*					45
PRCLIS	638	1246*													46
PRDATE	331*	481													47
PTRSV	711*	711	719	736											48
PULEDT	31*	33													49
PULMJA	29*	33													50
PULMTN	29*	33													51
PULSAR	95*	203													52
PULSEI	947	958*													53
PULEHD	28*	33													54
R	42*	114	145	146	147	148	150	153	157	159	167	168	171	175	55
	181	163	192	350	416	418	419	424	470	482	504	505	944	945	56
	946	947	648												57
															58
															59
															60
															61
															62
															63
															64
															65
															66
															67
															68

1		5	
2		6	
3	RETFIN	650	657#
4	RETIT	769#	
5	SCRNCH	104	426#
6	SECS	364#	364
7	SIX	403#	403
8	SIXDO	308	400#
9	SIXOU	397#	
10	SIXOUT	381	387
11	SIZZ	52#	
12	STICKIT	895	900#
13	STR	351	416#
14	SVLST	593#	593
15	SVOLG	639#	639
16	TBF2	1022	1024#
17	TBK	1243#	
18	TBUF	121	121#
19	TIMDII	368#	477
20	TMPSF	770	838
21	TOPC	102	1045
22	TOPCFR	517	526#
23	TOPF	111	516
24	TOPH	000#	
25	TOPLVL	817	831#
26	TYPEME	782	827#
27	UNUSED	717	726
28	VARLIS	636	1047#
29	VERDII	274#	473
30	WIPPOP	541	548
31	WRDLN	725#	725
32	YEAR	337#	337
33	YESDUN	843	851#
34	YESPRC	819	824#
35	SRKLLT	537#	769
36	SLOPRC	158	769#
37	%XPUL	628	961
38	%XPULS	336	AB
39	XAHH	533	964#
40	XAHHE	522	529
41	ZCHANG	55#	
42	ZENDB	58#	645
43	XINT	48#	
44	XLOCK	54#	
45	XLOG	47#	
46	%NOBOU	56#	
47	ZPERM	53#	
48	ZPROC	51#	
49	ZPST	50#	944#
50	XPUL	961#	970
51	XPULSE	950	963#
52	XREA	58#	
53	XSTR	49#	
54	%OASL	1017	
55	JBAFR	502	
56			
57			
58			
59			
60			
61			
62			
63			
64			
65			
66			
67			
68			

1
2
3
4
5 SUBI 242
6 SUBITL 3
7 TITLE 1
8 TWOSEG 92
9 VAR 1028
10 XLIST 1958
11 Z 1821 1024 1041
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67

Above Business First, Inc. I
SPECIFIED WITH EYE TRACK PATENT PENDING
U.S. Patent & Trademark Office

1
2
3
4
5 SPOOLER RUNTIME 15 SECONDS, 46 KCS, 139 DISK READS, 2 DISK WRITES, 36 PAGES
6 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
7
8
9 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
10
11
12
13 **END** USER TND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
14
15
16
17 **END** USER TND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
18
19
20
21 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
22
23
24
25 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
26
27
28
29 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
30
31
32
33 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
34
35
36
37 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
38
39
40
41 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
42
43
44
45 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
46
47
48
49 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
50
51
52
53 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
54
55
56
57 **END** USER IND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
58
59
60
61 **END** USER TND PROJECTS [3000,3000] JOB PULSTAR SEQ. 17951 DATE 12-APR-79 14:34:19 MONITOR ALBUQUERQUE SCHOOLS 603 **END**
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100