

L B A K U P D
S Q S S 1

Spool Queue Line #: 3
IRIS LU/Filename : 18/L.BAKUPDS.9291

Printed on/at : FEB 5, 1990 17:13:12
For Group/User: 0 , 1
On Port No: 5

Print control parameters :
Printer Class code : 0
Form Code/paper type : ?
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : Y
Notify User when done: N
Comments, optional : For R9.5 RELSE CNT

```
. EOT ; "RxxJCL.BAKUPDS" (e. i. CALL 79 used by MA  
. EOT  
. EOT ; "DSUBDEFS" FOR IRIS  
. END ; "R90BAKUPDSUB"
```

ASM 18/A. BAKUPDS. 9291!, @18/L. BAKUPDS. 9291!, B050, -B051, B052
FEB 5, 1990 16:56:36

; Batchfile: R95JCL. BAKUPDS

; ; D=9291 (YDDD)

; -R95DEFSPZ
; -R95DSUBDEFSD
; R92BAKUPDSUBSA
;

.EOT ; "RxxJCL. BAKUPDS" (e. i. CALL 79 used by MAPACTIVATE, ect. " FOR R9. x

```
;          << SI = R92BAKUPDSUBSA; BO = 18/A. BAKUPDS. 9291! >>
; READS A BINARY PROGRAM FILE INTO MEMORY AND JUMPS TO IT.
; DATE EDITED 16 MAR 84 BY RB.
; LAST EDITED BY RDC FOR B. 3
; *****
; BASIC CALLING SEQUENCE:
;   CALL 79, P$ {,PARAMETER LIST}
; WHERE P$ = FILENAME {,OPTIONAL STARTING ADDRESS OFFSET}
; P$ MUST BE DIM'ED LARGE ENOUGH TO HOLD THE FULL NUMBER OF BLOCKS
; IN THE PROGRAM FILE. THE DISCSUB WILL READ THE PROGRAM FILE NAMED IN P$
; INTO P$ AND THEN EXECUTE IT. IF A STARTING ADDRESS OFFSET IS GIVEN,
; EXECUTION WILL BEGIN AT THAT OFFSET.
; ALTERNATE CALL:
;   IF THE SECOND WORD OF P$ IS 125252 OCTAL (IE. 3RD AND 4TH BYTES = ** )
;   THEN P$ ITSELF WILL BE EXECUTED, IE. THE DISCSUB WILL JUMP TO THE
;   FIRST WORD OF P$.
; IN EITHER CASE WHEN THE DISCSUB JUMPS INTO P$ IT WILL DO SO WITH
;   A2 = POINTER TO ORIGINAL CALL PARAMETER LIST
;   A1 = POINTER TO WORK AREA CONTAINING PROGRAM FILE RDA LIST
;       (= INDETERMINATE IN ALTERNATE CALL)
;   A3 = RETURN ADDRESS TO A JMP @.NRET, FOLLOWED BY A JMP @.SRET
; THE CODE EXECUTED IN P$ MUST END IN A JMP @.NRET OR JMP @.SRET OR
;   JMP 0,3 OR JMP 1,3.
; ANY BASIC PROGRAM CALLING THIS DISCSUB MUST BE GUARDED WITH
; THE CAL79DOOM BIT.
; *****
```

106000 .LOC LBAKU

0 PROGM= 0 ; OFFSET TO PROGRAMNAME
1 SIZEA= PROGM+1 ; OFFSET TO ITS DIM

106000 173 START:BAKU ; DSUB #
106001 3 ENTER-START
106002 177541 START-END

<< SI = R92BAKUPDSUBSA; BO = 18/A. BAKUPDS. 9291! >>

```
106003 50564 ENTER: STA 2,ACTM2 ;SAVE POINTER TO VARIABLE TABLE
106004 20561 LDA 0,V79DOOM
106005 6101 CALL
106006 172 DOOMCHECK ;CHECK IF PROGRAM IS PROPERLY GUARDED
106007 2112 JMP @.NRET ; IT ISN'T -- RETURN WITH A3 --> MESSAGE
106010 152400 SUB 2,2
106011 50555 STA 2,NODEA ;INIT TO NO NODE AQUIRED
106012 30555 LDA 2,ACTM2
106013 25001 LDA 1,SIZEA,2;PICK UP DIM OF PROGRAM AREA TO BE USED (P*)
106014 125113 SSN 1,1 ;IS IT A STRING ?
106015 6112 JSR @.NRET ; NO, SYNTAX ERROR
106016 125300 MOVS 1,1 ;DIVIDE NUMBER OF WORDS BY 256.
106017 20052 LDA 0,C177
106020 123405 AND 1,0,SNR ;EXTRACT INTEGER PORTION OF RESULT
106021 502 JMP JER34 ; DIM NOT COMPATIBLE !
106022 40551 STA 0,NOSEC ;SAVE SIZE OF P* IN TERMS OF # OF SECTORS
106023 31000 LDA 2,PROGM,2;PICK UP CORE ADDRESS OF P*
106024 50560 STA 2,STADR ;SAVE AS STARTING ADDRESS
106025 21001 LDA 0,1,2 ;PICK UP SECOND WORD OF P*
106026 24550 LDA 1,CONST
106027 106415 SNE 0,1 ;IS IT 125252 ?
106030 527 JMP JMP2P ; YES, JUMP STRAIGHT INTO IT
106031 20040 LDA 0,SBA
106032 112540 SUBOL 0,2 ;CONVERT START OF P* TO REL. BYTE POINTER
106033 50545 STA 2,SBPTR ;SAVE AS SOURCE BYTE POINTER
106034 152400 SUB 2,2
106035 6107 FREENODE ;PICK UP POINTER TO WORKING AREA
106036 50530 STA 2,NODEA ;SAVE NODE ADDRESS
106037 155000 MOV 2,3
106040 54534 STA 3,TABLE ;SAVE POINTER TO THIS WORK AREA
106041 54536 STA 3,POINT
106042 165120 MOVZL 3,1 ;AC1=BYTE POINTER
106043 44536 STA 1,DBPTR ;SAVE AS DESTINATION BYTE POINTER
106044 44536 STA 1,FNAME ;AND FILENAME POINTER, FOR LATER
106045 20536 LDA 0,Csize
106046 103000 ADD 0,0 ;CONVERT WORKING AREA SIZE TO # BYTES
106047 40526 STA 0,COUNT ;SET A LOOP COUNTER TO MAX. # BYTES
106050 24530 LOOP2: LDA 1,SBPTR
106051 10527 ISZ SBPTR
106052 6144 XGETBYTE ;GET A BYTE FROM P*
106053 141005 MOV 2,0,SNR ;NULL BYTE (TERMINATOR) ?
106054 423 JMP STERM ; YES, END OF FILENAME
106055 24530 LDA 1,COMMA ;NO
106056 106415 SNE 0,1 ;IS IT A COMMA ?
106057 407 JMP RDSAD ; YES, READ STARTING ADDRESS FROM P*
106060 24521 LDA 1,DBPTR ;NO
106061 10520 ISZ DBPTR
106062 6134 PUTBYTE ;PUT BYTE INTO WORKING AREA
106063 14512 DSZ COUNT ;DECREMENT NUMBER OF BYTES TO GO COUNTER; DONE ?
106064 764 JMP LOOP2 ; NO, GO MOVE NEXT BYTE
106065 412 JMP STERM ;YES, STORE A TERMINATOR BYTE
```

<< SI = R92BAKUPDSUBSA; BO = 18/A. BAKUPDS. 9291! >>

```
106066 24512 RDSAD: LDA 1, SBPTR ; READ STARTING ADDRESS
106067 20026 LDA 0, C6
106070 6120 DECIMAL
106071 100010 NOP
106072 6121 FIX ; CONVERT TO BINARY
106073 513 JMP JNRET ; OVERFLOW !?
106074 20510 LDA 0, STADR
106075 123000 ADD 1, 0 ; ADD OFFSET TO STARTING ADDRESS
106076 40506 STA 0, STADR
106077 102400 STERM: SUB 0, 0 ; STORE TERMINATOR BYTE IN WORKING AREA
106100 24501 LDA 1, DBPTR
106101 6134 PUTBYTE
106102 24500 LDA 1, FNAME
106103 102400 SUB 0, 0 ; DEFAULT LU = 0
106104 6101 CALL ; ATTEMPT TO FIND FILENAME FROM P*
106105 3 FFILE
106106 502 JMP NTFND ; FILE NOT FOUND - GO OUTPUT ERROR MESSAGE
106107 40461 STA 0, UNITT ; SAVE LOGICAL UNIT NUMBER FILE WAS FOUND ON
106110 25407 LDA 1, 7, 3 ; PICK UP HEADER BLOCK DISC ADDRESS OF P* FILE
106111 30456 LDA 2, ACTM2 ; RESTORE POINTER TO VARIABLE TABLE
106112 31000 LDA 2, PROG, 2 ; PICK UP CORE ADDRESS OF P*
106113 50456 STA 2, CORE ; SAVE P* POINTER FOR LATER READS
106114 6135 READBLOCK ; READ HEADER BLOCK OF P* FILE INTO P*
106115 21011 LDA 0, NBLK, 2 ; PICK UP NUMBER OF BLOCKS BELONGING TO P* FILE
106116 126520 SUBZL 1, 1 ; AC1=1
106117 122405 SUB 1, 0, SNR ; AC0=NUMBER OF BLOCKS IN P* FILE EXCLUDING HEADER BLOCK
106120 472 JMP ERR36 ; NO BLOCKS !?
106121 24452 LDA 1, NOSEC
106122 106033 SLS 0, 1 ; NUMBER OF BLKS. IN P* FILE > DIMENSION OF P* ?
106123 471 JER34: JMP ERR34 ; YES, "DIMENSIONS NOT COMPATIBLE"
106124 24457 LDA 1, CSIZE
106125 106433 SLE 0, 1 ; TOO MANY BLOCKS FOR WORKING AREA ?
106126 466 JMP ERR34 ; YES
106127 40443 STA 0, NOBLK ; SAVE NUMBER OF BLOCKS IN P* FILE (EXCLUDING HEADER)
106130 100400 NEG 0, 0 ; AC0= NEGATIVE NOBLK
106131 24053 LDA 1, C200
106132 133000 ADD 1, 2 ; AC2= POINTER TO BLOCK TABLE IN HEADER OF P* FILE
106133 34441 LDA 3, TABLE ; PICK UP CORE ADDRESS OF PREVIOUSLY LOCATED WORK AREA
```

```

;<< SI = R92BAKUPDSUBSA; BO = 18/A.BAKUPDS.9291! >>
106134 25000 LOOP3: LDA 1,0,2 ; PICK UP A BLOCK ADDRESS FROM HEADER
106135 125005 MOV 1,1,SNR ; SKIP IF BLOCK ADDRESS NOT ZERO
106136 454 JMP ERR36 ; PROBLEM WITH P* FILE TYPE
106137 45400 STA 1,0,3 ; STORE BLOCK ADDRESS IN WORK AREA
106140 151400 INC 2,2 ; INCREMENT BLOCK TABLE POINTER
106141 175400 INC 3,3 ; INCREMENT WORK AREA POINTER
106142 101404 INC 0,0,SZR ; INCREMENT NUMBER OF WORDS TO GO - SKIP IF DONE
106143 771 JMP LOOP3
106144 20424 LOOP1: LDA 0,UNITT ; LOAD UNIT NUMBER OF P* FILE
106145 26427 LDA 1,@TABLE ; LOAD A BLOCK ADDRESS FROM WORK AREA
106146 30423 LDA 2,CORE ; LOAD CORE ADDRESS OF P*
106147 6135 READBLOCK ; READ A BLOCK OF THE P* FILE INTO P* AREA
106150 10424 ISZ TABLE ; INCREMENT WORK AREA POINTER BY 1
106151 20420 LDA 0,CORE ; COMPUTE CORE = CORE + 400 OCTAL
106152 24065 LDA 1,C400
106153 123000 ADD 1,0
106154 40415 STA 0,CORE ; STORE UPDATED RESULT BACK IN CORE
106155 14415 DSZ NOBLK ; DECREMENT NUMBER OF BLOCKS TO GO BY 1; ALL DONE ?
106156 766 JMP LOOP1 ; NO, GO READ NEXT BLOCK OF P* FILE
106157 30410 JMP2P: LDA 2,ACTM2 ; RESTORE ORIGINAL CALL'S VARIABLE TABLE POINTER
106160 24417 LDA 1,POINT
106161 6423 JSR @STADR ; GO EXECUTE PROGRAM JUST READ INTO P* AREA
;
106162 424 JMP JNRET ; NON-SKIP RETURN
106163 4446 JSR RENOD ; RETURN NODE
106164 2113 JMP @.SRET ; SKIP RETURN

```

```

106165 4 V79DOOM: CAL79DOOM; DOOM BIT FOR CALL 79
106166 0 NODEA: 0 ; A(NODE) FOR FILENAME
106167 0 ACTM2: 0 ; POINTER TO CALL PARAMETER LIST
106170 0 UNITT: 0 ; LU OF PROGRAM FILE
106171 0 CORE: 0 ; CORE POINTER FOR READING PROGRAM BLOCKS
106172 0 NOBLK: 0 ; NO. BLOCKS IN PROGRAM FILE
106173 0 NOSEC: 0 ; NO. SECTORS THAT P* CAN HOLD
106174 0 TABLE: 0 ; POINTER TO WORKING AREA
106175 0 COUNT: 0 ; COUNTER
106176 125252 CONST: 125252 ; CODE FOR ALTERNATE CALL
106177 0 POINT: 0 ; POINTER TO WORKING AREA
106200 0 SBPTR: 0 ; SOURCE BYTE POINTER
106201 0 DBPTR: 0 ; DESTINATION BYTE POINTER
106202 0 FNAME: 0 ; BYTE POINTER TO FILENAME STRING
106203 40 CSIZE: 40 ; SIZE OF NODE
106204 0 STADR: 0 ; STARTING ADDRESS IN P*
106205 254 COMMA: 200+";

```

```

106206 4423 JNRET: JSR RENOD ; RETURN NODE
106207 2112 JMP @.NRET ; NON-SKIP RETURN

```

<< SI = R92BAKUPDSUBSA; BO = 18/A. BAKUPDS. 9291! >>

; ERROR HANDLERS

```

      12 .RDX  10
106210  4407 NTFND: JSR   ERRET
106211  52    42    ; "FILE NOT FOUND"

106212  4405 ERR36: JSR   ERRET
106213  44    36    ; "SUBROUTINE NOT IN STORAGE"

106214  4403 ERR34: JSR   ERRET
106215  42    34    ; "DIMENSIONS NOT COMPATIBLE"

      10 .RDX  8
106216  0    ERRAD: 0    ; ADDRESS OF ERROR

106217  54777 ERRET: STA   3,ERRAD ; SAVE ERROR ADDRESS
106220  4411   JSR   RENOD ; RETURN NODE
106221  34775   LDA   3,ERRAD
106222  25400   LDA   1,0,3 ; PICK UP ERROR NUMBER
106223  102000  ADC   0,0 ; ACO=-1
106224  6101   CALL  ; OUTPUT ERROR MESSAGE TO USER
106225  13     MESSAGE
106226  2113   JMP   @.SRET ; RETURN TO BASIC
106227  2113   JMP   @.SRET ; RETURN TO BASIC

106230  0      0
106231  54777 RENOD: STA   3,-1
106232  30734   LDA   2,NODEA ; A(NODE)
106233  151015  SNZ   2,2 ; DID WE GET A NODE?
106234  2774   JMP   @RENOD-1 ; NO
106235  6107   FREENODE ; YES, RETURN IT
106236  2772   JMP   @RENOD-1

106237  END=.

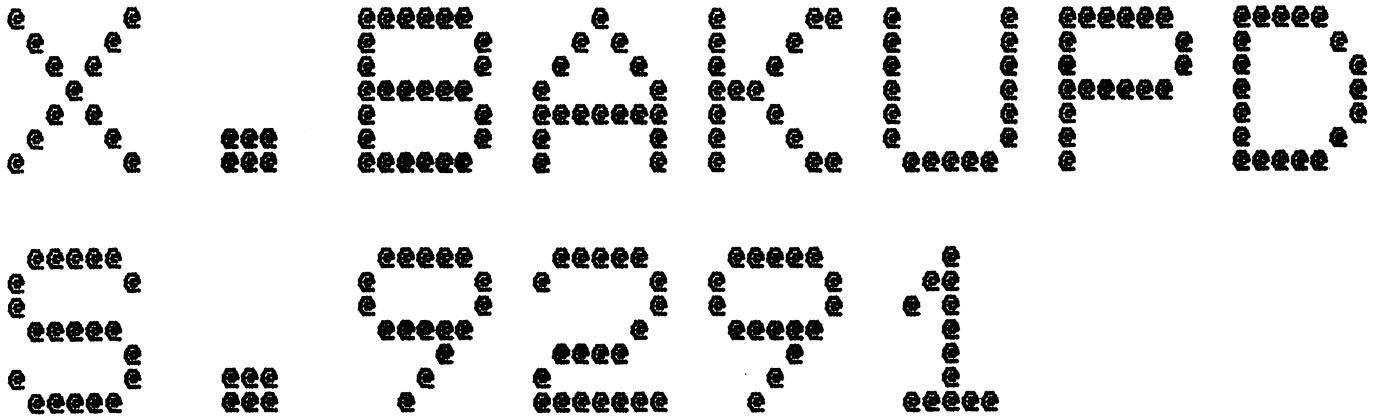
      0      .ERR  START+400<. ; OVERFLOW CHECK

      .END  ; "R90BAKUPDSUB"
```


ACTM2	106167	BINDI	6115	BINMU	6116	BPI	16	BSACF	75
BUMPU	6117	C10	30	C100	51	C1000	67	C11	31
C12	32	C13	33	C14	34	C15	35	C16	36
C160	174	C163	175	C166	176	C17	37	C170K	21
C171	177	C177	52	C1777	70	C2	2	C20	42
C200	53	C2000	71	C205	54	C215	55	C240	56
C244	57	C260	60	C271	61	C3	3	C300	62
C334	63	C37	43	C377	64	C4	24	C40	44
C400	65	C4000	72	C5	25	C6	26	C600	100
C7	27	C77	50	C774C	22	C777	66	CALL	6101
CHANN	6106	CM400	23	COMMA	106205	CONST	106176	CORE	106171
COUNT	106175	CSIZE	106203	DA	160	DAC	164	DAS	165
DATAP	6110	DB	166	DBA	41	DBC	172	DBPTR	106201
DBS	173	DECIM	6120	DFTCA	34106	DMCAL	34110	DGUEU	6105
END	106237	ENTER	106003	ERR34	106214	ERR36	106212	ERRAD	106216
ERRET	106217	ERRF	76	ESCF	73	ETSF	74	FINDL	6123
FIX	6121	FLAGC	6102	FLOAT	6122	FNAME	106202	FREEN	6107
GETBY	6124	HALTS	6153	INBYT	6125	INSTB	6126	IOCAL	34103
IOP	6	ISA2D	6127	ISA2L	6130	JER34	106123	JFLTO	151
JMP2P	106157	JNRET	106206	LACNT	4000	LAFSE	13000	LALCO	47400
LALLO	1400	LATOE	36000	LBAKU	106000	LBILD	5000	LBUIL	4400
LCALL	75000	LCHAN	41000	LCHFL	30000	LCHSU	61000	LCLEA	7400
LCLOS	7000	LCLPY	76000	LCNVA	11400	LCNVD	12000	LCOMM	33400
LDALC	2000	LDALL	1000	LDB7A	114000	LDB7B	114400	LDB7C	115000
LDB7D	115400	LDB7E	116000	LDB7F	116400	LDB7G	117000	LDB7H	117400
LDB7I	120000	LDEKE	52400	LDELE	3400	LDIRE	50400	LDLTP	20400
LDREN	37400	LDSB1	400	LDSB2	22400	LDSB3	47000	LDSB4	65000
LDSB5	77000	LDSB6	106400	LDSB7	113400	LECHO	37000	LEOB7	105400
LERRO	23000	LFAUL	400	LEFFIL	2400	LEFXD	57400	LFNDC	112000
LFNDL	20000	LFOFI	17000	LQETR	10000	LQHOP	107400	LQHOS	107000
LGMUX	16000	LHCON	17400	LIBCA	44400	LIBEN	45000	LIBTR	45400
LIDAT	103000	LLINK	35400	LLOAD	34400	LLGI	32000	LLUIN	112400
LMAPB	73000	LMDE0	65000	LMDE1	66000	LMDE5	71400	LMRC3	56400
LMPFH	57000	LMPFI	54000	LMTAP	55400	LMTAS	54400	LMTFP	56000
LMTFY	60400	LMTNX	55000	LMTPL	60000	LOADD	6131	LOOP1	106144
LOOP2	106050	LOOP3	106134	LOPEN	6000	LOPNM	13400	LPATG	110000
LPEXP	23400	LPFAB	72000	LPFLN	73400	LPFNA	3000	LPFRL	72400
LPFSE	67000	LPFSH	70000	LPSFX	70400	LPLDG	24400	LPPWR	33000
LPRAN	36400	LPRCO	71000	LPSIN	25400	LPSGR	22400	LPTAN	25000
LQIBF	63400	LQICL	63000	LQIOP	62400	LRDFH	26400	LRDIS	31400
LRDSE	110400	LREDC	50000	LREDI	11000	LREDM	14000	LREDP	74000
LRENA	15000	LREOP	53000	LRESO	42000	LRWIT	113000	LRWMB	14400
LRWSX	111400	LS105	77000	LS152	102000	LS153	101000	LS154	100400
LS156	101400	LS157	100000	LSAVE	43000	LSAVP	43400	LSEAB	64000
LSEAR	51000	LSETF	40000	LSHUF	52000	LSIQP	12400	LSING	40400
LSMCS	106400	LSPPEC	27000	LSTRI	32400	LSYSC	30400	LTP01	102400
LTP03	104000	LTP04	104400	LTP05	105000	LVMUX	42400	LWRIT	47000
LXMIN	62000	NOBLK	106172	NODEA	106166	NOSEC	106173	NTFND	106210
OUTBY	6132	OUTTE	6133	PIB	4	POINT	106177	PROGM	0
PUTBY	6134	QCHAR	6103	QUEUE	6104	RDSAD	106066	READB	6135
RELJM	6136	RENOD	106231	RJSR	6136	RTP	7	RUP	5
SBA	40	SBPTR	106200	SCDCA	34147	SIZEA	1	SPINP	6146
STADR	106204	START	106000	STERM	106077	STINP	6140	STINT	6147
STORD	6137	STOUT	6141	TABLE	106174	TASKG	15	TRAPF	6142
UNITT	106170	V79DO	106165	WRITB	6143	XGETB	6144	XPUTB	6145
.ABA	14	.BPS	77	.BSA	10	.DA	174	.DA3	175
.DB	176	.DB3	177	.FLTO	152	.HBA	11	.HXA	12
.INFO	100	.INTR	111	.LCM	114	.NRET	112	.SRET	113

. SSA

13



Spool Queue Line #: 4
IRIS LU/Filename : 18/X.BAKUPDS.9291

Printed on/at : FEB 5, 1990 17:14:26
For Group/User: 0, 1
On Port No: 5

Print control parameters :
Printer Class code : 0
Form Code/paper type : ?
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : Y
Notify User when done: N
Comments, optional : For R9.5 RELSE CNT

X BAKUPD

9291

Spool Queue Line #: 4
IRIS LU/Filename : 18/X. BAKUPDS. 9291

Printed on/at : FEB 5, 1990 17:14:32
For Group/User: 0, 1
On Port No: 5

Print control parameters :
Printer Class code : 0
Form Code/paper type : ?
Print Priority (0-9) : 5
Starting Page Number : 1
This is copy number : 1
Keep file (Y/N) : Y
Notify User when done: N
Comments, optional : For R9.5 RELSE CNT

***** J O B S T A T I S T I C S *****

0	TOTAL # DUPLICATE KEYS
0	TOTAL # DIR. RE-ORGS
148	TOTAL # KEYS INSERTED
0	TOTAL # ASSEMBLY ERRS

.ERR	6.046				
.NRET	3.011	3.017	5.055		
.RDX	6.009	6.020			
.SRET	5.031	6.031	6.032		
ACTM2	3.007	3.014	4.026	5.025	5.037:
BAKU	2.051				
C177	3.019				
C200	4.042				
C400	5.020				
C6	4.008				
CAL79	5.035				
CALL	3.009	4.021	6.029		
COMMA	3.049	5.051:			
CONST	3.026	5.044:			
CORE	4.028	5.016	5.019	5.022	5.039:
COUNT	3.043	3.055	5.043:		
CSIZE	3.041	4.037	5.049:		
DBPTR	3.039	3.052	3.053	4.017	5.047:
DECIM	4.009				
DDOMC	3.010				
END	2.053	6.044=			
ENTER	2.052	3.007:			
ERR34	4.036	4.039	6.017:		
ERR36	4.033	5.008	6.014:		
ERRAD	6.022:	6.024	6.026		
ERRET	6.011	6.014	6.017	6.024:	

FFILE	4.022					
FIX	4.011					
FNAME	3.040	4.019	5.048:			
FREEN	3.033	6.039				
JER34	3.021	4.036:				
JMP2P	3.028	5.025:				
JNRET	4.012	5.029	5.054:			
LBAKU	2.044					
LOOP1	5.014:	5.024				
LOOP2	3.044:	3.056				
LOOP3	5.006:	5.013				
MESSA	6.030					
NBLK	4.030					
NOBLK	4.040	5.023	5.040:			
NODEA	3.013	3.034	5.036:	6.036		
NOP	4.010					
NOSEC	3.022	4.034	5.041:			
NTFND	4.023	6.011:				
POINT	3.037	5.026	5.045:			
PRDGM	2.047=	2.048	3.023	4.027		
PUTBY	3.054	4.018				
RDSAD	3.051	4.007:				
READB	4.029	5.017				
RENDD	5.030	5.054	6.025	6.035:	6.038	6.040
SBA	3.029					
SBPTR	3.031	3.044	3.045	4.007	5.046:	

FEB 5, 1990 16:56

18/L. BAKUPDS. 9291!

PAGE 3

SIZEA	2.048=	3.015			
STADR	3.024	4.013	4.015	5.027	5.050:
START	2.051:	2.052	2.053	6.046	
STERM	3.048	3.057	4.016:		
TABLE	3.036	4.044	5.015	5.018	5.042:
UNITT	4.024	5.014	5.038:		
V79DO	3.008	5.035:			
XGETB	3.046				