

.REM -

IDENTIFICATION

PRODUCT CODE: AC-E959R-MC
PRODUCT NAME: CXAABBO AA11-K MODULE
PRODUCT DATE: SEPTEMBER 1978
MAINTAINER: DEC/X11 SUPPORT GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITALS COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1976,1978 DIGITAL EQUIPMENT CORPORATION

1. ABSTRACT:

"AABR" IS A IOMOD THAT EXERCISES THE A11-K SCOPE CONTROLLER. A CONFIDENCE LOGIC TEST IS EXECUTED ON THE CONTROL STATUS LOGIC. DACS, ARE REPORTED TO THE CONSOLE TELETYPE. THE MAJOR PORTION OF THIS MODULE IS DEFERRED TO LEVEL 0 SERVICE. A SIX LETTER MESSAGE (A11-K) WILL BE PLOTTED ON THE SCOPE IS CONNECTED. ALTERNATING CHARACTERS MAY APPEAR TO HAVE ALTERNATING INTENSITY LEVELS. WHEN A VRI4 SCOPE IS CONNECTED CHANGING THE CHANNEL SWITCH TO CHANNEL ONE ONLY WILL DISPLAY "A 1 _ _". PLACING THE SWITCH TO CHANNEL TWO ONLY WILL DISPLAY "A 1 K".

2. REQUIREMENTS:

HARDWARE: A11-K INTERFACE WITH A SCOPE DISPLAY INSTALLED

STORAGE: AAB REQUIREMENTS:
1. DECIMAL WORDS: 453
2. OCTAL WORDS: 0705
3. OCTAL BYTES: 1612

3. PASS DEFINITION:

ONE PASS OF THE AAB MODULE CONSISTS OF DISPLAYING 55,296 POINTS ON THE SCREEN. THIS MEANS THAT 55,296 DATA TRANSFERS OCCURRED ON THE UNIBUS.

4. EXECUTION TIME:

VARIABLES WITH SCOPE DELAY BUT SHOULD TAKE AN AVERAGE OF SIXTY SECONDS TO COMPLETE ONE PASS. WHEN RUNNING ALONE ON AN 11/05.

5. CONFIGURATION PARAMETERS:

DEFAULT PARAMETERS:

DVA: 170416, VCT: 360, BRI: 4

REQUIRED PARAMETERS:

NONE

6. DEVICE OPTION SETUP:

A. TURN ON SCOPE POWER.

B. PLACE CHANNEL SW TO 1 & 2 (IF VR14)

7. MODULE OPERATION:

7.1 TEST SEQUENCE:

A. START:

USING THE DEVICE ADDRESS, THIS SECTION OF CODE DETERMINES THE CONTROL, X AND Y POSITION ADDRESSES, AND VECTORS.

B. TSTNRG:

THIS SECTION OF CODE PERFORMS A CONFIDENCE REGISTER TEST OF THE CONTROL DAC 0, DAC 1, DAC 2, AND DAC 3 REGISTERS.

C. PRIME:

IN THIS SECTION, THE FOUR DAC REGISTERS AND CONTROL REGISTERS ARE LOADED. THE SCOPE IS ENABLED AND AN 'EXIT' RETURN TO THE MONITOR.

D. AA11K:

UPON A SCOPE INTERRUPT, THE PROGRAM WILL RETURN TO THIS CODE. ENTER DEFERRED SERVICE MODE AND TEST FOR A MODE FLAG. IF NO MODE FLAG, REPORT IT AS AN ERROR.

E. AAL1KA: THRU CHAR13: THIS SECTION SELECTS THE PROPER POINTS TO BE INTENSIFIED ON THE SCREEN.

F. CHAR11: IN THIS CODE, THE CHANNEL BIT ALTERNATED TO DISPLAY EACH CHANNEL. IF A 611/613 SCOPE IS CONNECTED ALTERNATING CHARACTERS WILL HAVE ALTERNATING INTENSITY LEVELS.

G. CHAR20: IN THIS SECTION, THE PASS COUNT IS DECREMENTED AND TESTED. IF IT DID NOT BECOME ZERO, THEN SELECT ANOTHER RATE AND RESUME COUNTING UPON A ZERO PASS COUNT. THE CONTROL AND PRESET REGISTERS ARE CLEARED AND FENDPAS IS REPORTED.

8. OPERATOR OPTIONS:

A. LOCATION (VCPASS) CAN BE MODIFIED TO VARY THE NO. LOOPS THRU TEST BEFORE END OF PASS IS REPORTED.

9. NON-STANDARD PRINTOUTS:

NONF: ALL PRINTOUTS HAVE THE STANDARD FORMATS DESCRIBED IN THE DEC/X11 DOCUMENT

```
178  
179  
180  
181 000000  
182 000000  
183  
184  
185  
186  
187 000000  
188 000000 040501 041102 040  
189 000005 000  
190 000006 170416  
191 000010 000360  
192 000017 000200  
193 000014 000  
194 000014 000001  
195 000016 000000  
196 000020 000000  
197 000024 000000  
198 000024 000000  
199  
200 000026 140000  
201 000030 000244  
202 000032 000224  
203 000034 000000  
204 000036 003000  
205 000040 000000  
206 000042 000000  
207 000044 000000  
208 000046 000000  
209 000048 000000  
210 000052 000000  
211 000054 000000  
212 000056 000000  
213 000056 000000  
214 000060 000000  
215 000062 000000  
216 000064 000000  
217 000066 000000  
218 000070 000000  
219 000072 000000  
220 000074 000000  
221 000076 000000  
222 000100 000000  
223 000102  
224 000102 000000  
225 000104 000000  
226 000104 000000  
227 000106 000000  
228 000106 000000  
229 000110 000000  
230 000112 000416  
231 000114 000000  
232 000116 000000  
233 000120 000000
```

```
*****  
;AA11-K DEC/X11 EXERCISER MODULE  
IDMOD <AAAB> 170416 360 4 3000 123  
MODULE 140000 AAAB 170416 360 4 3000 123  
; TITLE AAAB DEC/X11 SYSTEM EXERCISER MODULE  
; DDCOM VERSION 6 23-MAY-78  
; LIST BIN  
*****  
BFCIN: *****  
MODNAM: ASCII /AAAB / ;MODULE NAME  
XPLAC: BYTE OPEN ;USED TO KEEP TRACK OF WBUF USAGE  
ADDR: 170416+0 ;LIST DEVICE ADDR  
VRCOR: 360+0 ;LIST DEVICE VECTOR  
BR1: -BYTE PRTV4+0 ;LIST BR LEVEL  
BR2: -BYTE PRTV+0 ;2ND BR LEVEL  
DVIDL: +1 ;DEVICE INDICATOR 1  
SR1: OPEN ;SWITCH REGISTER 1  
SR2: OPEN ;SWITCH REGISTER 2  
SR3: OPEN ;SWITCH REGISTER 3  
SR4: OPEN ;SWITCH REGISTER 4  
*****  
STAT: 140000 ;STATUS WORD  
INIT: START ;MODULE START ADDR  
SPOINT: MODSP ;MODULE STACK POINTER  
PASCNT: 0 ;PASS COUNTER  
ICOUNT: 3000 ;# OF ITERATIONS PER PASS=3000  
SOFCNT: 0 ;LOC TO COUNT ITERATIONS  
HRDCAS: 0 ;LOC TO SAVE TOTAL SOFT ERRORS  
SOFPAS: 0 ;LOC TO SAVE TOTAL HARD ERRORS  
HRDPAS: 0 ;LOC TO SAVE SOFT ERRORS PER PASS  
SYSNUM: 0 ;LOC TO SAVE HARD ERRORS PER PASS  
RANNUM: 0 ;# OF SYS ERRORS ACCUMULATED  
; HOLDS RANDOM # WHEN RAND MACRO IS CALLED  
CONFIG: *****  
RES1: 0 ;RESERVED FOR MONITOR USE  
RES2: 0 ;RESERVED FOR MONITOR USE  
SVRO: OPEN ;LOC TO SAVE R0  
SVR1: OPEN ;LOC TO SAVE R1  
SVR2: OPEN ;LOC TO SAVE R2  
SVR3: OPEN ;LOC TO SAVE R3  
SVR4: OPEN ;LOC TO SAVE R4  
SVR5: OPEN ;LOC TO SAVE R5  
SVR6: OPEN ;LOC TO SAVE R6  
SVR7: OPEN ;LOC TO SAVE R7  
SVR8: OPEN ;LOC TO SAVE R8  
SVR9: OPEN ;LOC TO SAVE R9  
SPADK: *****  
ACSR: OPEN ;ADDR OF CORRENT CSR  
WASADP: *****  
WASADP: OPEN ;ADDR OF GOOD DATA, OR  
; CONTENTS OF CSR  
WASADP: *****  
ERRTYP: OPEN ;ADDR OF BAD DATA, OR  
; STATUS REG CONTENTS  
ERRTYP: *****  
ASB: OPEN ;TYPE OF ERROR  
ASB: OPEN ;EXPECTED DATA  
AWAS: OPEN ;ACTUAL DATA  
RSTRT: RSTRT ;RESTART ADDRESS AFTER END OF PASS  
WDT0: OPEN ;WORDS TO MEMORY PER ITERATION  
WDFR: OPEN ;WORDS FROM MEMORY PER ITERATION  
INTR: OPEN ;# OF INTERRUPTS PER ITERATION  
*****
```

```
234 000122 000123  
235 000122 000040  
236  
237  
238  
239  
240 000224  
241  
242
```

```
IDNUM: 123 ;MODULE IDENTIFICATION NUMBER=123  
;REPT SPSIZ ;MODULE STACK STARTS HERE.  
;NLIST  
;WORD 0  
;LIST  
;ENDR  
MODSP: *****  
*****
```

```
243  
244  
245  
246 000224* 000000  
247  
248  
249  
250  
251 000226* 170416  
252 000230* 170420  
253 000232* 170424  
254 000234* 170424  
255 000236* 170426  
256  
257  
258  
259 000240* 000360  
260 000242* 000362  
261  
262  
263  
264 000244* 016767 177566 177752  
265 000252* 005367 177746  
266 000256* 012767 000365 177634  
267 000264* 012767 000365 177622  
268 000272* 012767 000365 177616  
269 000300* 016767 177502 177720  
270 000306* 016767 177474 177714  
271 000314* 062767 000002 177706  
272 000322* 016767 177460 177702  
273 000330* 062767 000004 177674  
274 000336* 016767 177444 177670  
275 000344* 062767 000006 177662  
276 000352* 016767 177430 177656  
277 000360* 062767 000010 177650  
278 000366* 016767 177410 177644  
279 000374* 016767 177410 177640  
280 000402* 062767 000002 177632  
281 000410* 016767 177612 177462  
282
```

;
;A11-K OPERATOR CHANGEABLE LOCATIONS
VCPAS: 0
;
; COMMON A11-K DEVICE ADDRESSES
VCSTAT: 170416 ;SCOPE STATUS REGISTER
VCXREG: 170420 ;SCOPE X AXIS REGISTER
VCYREG: 170424 ;SCOPE Y AXIS REGISTER
VCDAC2: 170424 ;DAC #2 ADDRESS
VCDAC3: 170426 ;DAC #3 ADDRESS
;
;COMMON A11-K DEVICE VECTOR
VCIV: 360 ;SCOPE INTERRUPT VECTOR
VCIVS: 362
;
;NOW SET UP THE ADDRESS AND VECTOR DISPATCH LOC.
START: MOV ICNT,VCPAS ;SAVE ICNT
DEC VCPAS ;MUST BE ONE LESS
MOV #245,-INTR ;245- INTERRUPTS/ITERATION
MOV #245,-WDTO ;245- WORDS TO NEW/ITERATION
MOV #245,-WDFR ;245- WORDS FROM NEW/ITERATION
MOV ADDR,VCSTAT
MOV ADDR,VCXREG
ADD #2,VCXREG
MOV ADDR,VCYREG
ADD #4,VCYREG
MOV ADDR,VCDAC2
MOV ADDR,VCDAC3
MOV VECTOR,VCIV
VECTOR,VCIVS
ADD #2,VCIVS ;LOAD DEVICE ADDRESS

```
283  
284  
285 000416* 005077 177604  
286 000422* 005077 177602  
287 000426* 005077 177600  
288 000432* 005077 177576  
289 000436* 005077 177574  
290 000442* 005067 001114  
291  
292  
293  
294 000446* 016767 177556 177424  
295 000454* 012767 010000 177422  
296 000462* 006267 177416  
297 000466* 001422  
298 000470* 016777 177410 177532  
299 000476* 017767 177526 177376  
300 000504* 026767 177374 177370  
301 000512* 001763  
302 000514* 012767 000025 177364  
303  
304 000522* 104405 000000* 000000  
305  
306 000530* 005267 001026  
307 000534* 000240  
308  
309  
310  
311 000536* 016767 177470 177334  
312 000544* 012767 010000 177332  
313 000552* 006267 177326  
314 000556* 001422  
315 000560* 016777 177320 177444  
316 000566* 017767 177440 177306  
317 000574* 026767 177304 177300  
318 000602* 001763  
319 000604* 012767 000025 177274  
320  
321 000612* 104405 000000* 000000  
322  
323 000620* 005267 000736  
324 000624* 000240
```

;
;A11-K LOGIC TEST
RESTR: CLR QVSTAT ;CLEAR STATUS
CLR QVCXREG ;CLEAR X
CLR QVCYREG ;CLEAR Y
CLR QVCDAC2 ;CLEAR DAC #2
CLR QVCDAC3 ;CLEAR DAC #3
CLR FATAL ;CLEAR FATAL ERROR FLAG
;
;NOW TEST DAC#0 REGISTER BITS
TSTXRC: MOV VCYREG,CSRA ;LOAD BUS ADDRESS
MOV #RIT1,ASTAT ;LOAD EXPECTED BIT
LS: ASR ASTAT ;CHANGE THE DATA
SRF 2S ;RR IF DONE
MOV ASTAT,QVCXREG ;LOAD DAC # 0
MOV QVCXREG,ACSR ;READ DAC #
CMP ASTAT,ACSR ;COMPARE RESULT
SRF 1S ;RR IF CORRECT
MOV #25,ERRTYP ;RR STUCK
;*****
RRORS,REGIN,NULL ;DAC # 0 IN ERROR
;*****
INC FATAL ;SET FATAL ERROR FLAG
NOP
2S:
;
;NOW TEST DAC #1 REGISTER BITS
TSTXRC: MOV VCYREG,CSRA ;LOAD BUS ADDRESS
MOV #RIT1,ASTAT ;LOAD EXPECTED BIT
LS: ASR ASTAT ;CHANGE THE DATA
SRF 2S ;RR IF DONE
MOV ASTAT,QVCYREG ;LOAD DAC # 1
MOV QVCYREG,ACSR ;READ DAC #
CMP ASTAT,ACSR ;COMPARE RESULT
SRF 1S ;RR IF CORRECT
MOV #25,ERRTYP ;RR STUCK
;*****
RRORS,REGIN,NULL ;DAC # 1 IN ERROR
;*****
INC FATAL ;SET FATAL ERROR FLAG
NOP
2S:

```
325 ;NOW TEST DAC #2 REGISTER BITS
326
327
328 000626 016767 177402 177244 TST2RG: MOV VCDAC2,CSRA ;LOAD BUS ADDRESS
329 000634 012767 010000 177242 MOV #BIT12,ASTAT ;LOAD EXPECTED BIT
330 000642 006267 177236 1S: ASR ASTAT ;CHANGE THE DATA
331 000646 001422 BFC 25 ;RR IF DONE
332 000650 016777 177230 177356 MOV ASTAT,@VCDAC2 ;LOAD DAC # 2
333 000656 017767 177352 177216 MOV @VCDAC2,ACSR ;READ DAC #
334 000664 016767 177214 177210 CMP ASTAT,ACSR ;COMPARE RESULT
335 000672 001763 BFC 15 ;RR IF CORRECT
336 000674 012767 000025 177204 MOV #25,ERRTYP ;BIT STUCK
337 ;*****
338 000702 104405 000000 000000 HRDERS,REGIN,NULL ;DAC # 2 IN ERROR
339 ;*****
340 000710 005267 006646 2S: INC FATAL ;SET FATAL ERROR FLAG
341 000714 000240 NOP
342
343 ;NOW TEST DAC #3 REGISTER BITS
344
345 000716 016767 177314 177154 TST3RG: MOV VCDAC3,CSRA ;LOAD BUS ADDRESS
346 000724 012767 010000 177152 MOV #BIT12,ASTAT ;LOAD EXPECTED BIT
347 000732 006267 177146 1S: ASR ASTAT ;CHANGE THE DATA
348 000736 001422 BFC 25 ;RR IF DONE
349 000740 016777 177140 177270 MOV ASTAT,@VCDAC3 ;LOAD DAC # 3
350 000746 017767 177264 177126 MOV @VCDAC3,ACSR ;READ DAC #
351 000754 016767 177124 177120 CMP ASTAT,ACSR ;COMPARE RESULT
352 000762 001763 BFC 15 ;RR IF CORRECT
353 000764 012767 000025 177114 MOV #25,ERRTYP ;BIT STUCK
354 ;*****
355 000772 104405 000000 000000 HRDERS,REGIN,NULL ;DAC # 3 IN ERROR
356 ;*****
357 001000 005267 000556 2S: INC FATAL ;SET FATAL ERROR FLAG
358 001004 000240 NOP
359
360 001006 005767 000550 TST FATAL ;TEST IF ANY FATAL LOGIC ERRORS
361 001012 001402 BFC PRIME
362 001014 104410 000000 ENDS,REGIN ;DROP MODULE DUE TO DAC REGISTER ERROR
363
364 001020 005077 177202 PRIME: CLR @VCSTAT ;ENSURE CLEAR STATUS
365 001024 012777 001052 177205 MOV #A11K,@VCIV ;SET UP LPSVC VECTOR
366 001032 116777 176754 177202 MOVPR #R1,@VCIVS ;SET UP
367 001040 012777 000101 177160 MOV #L01,@VCSTAT ;START DISPLAY AND INTERRUPT ENABLE
368
369 001046 104400 000000 EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
370
```

```
371 ;DISPLAY "A11-K" ON THE SCOPE
372
373 001052 A11K:
374 001052 000004 000000 001060 ;PIRQS,BEGIN,1S ; QUEUE UP TO CONTINUE AT 1S AND RTI
375
376 1S: TSTB @VCSTAT
377 001060 105777 177142 BML A11KA
378 001064 104415 MOV @VCSTAT,ACSR ;LOAD VALUE
379 001068 017767 177134 177006 CLR @VCSTAT
380 001074 005077 177126 MOV #11,ERRTYP ;ILLEGAL INTERRUPT
381 001100 012767 000011 177000 ;*****
382 001106 104405 000000 000000 HRDERS,REGIN,NULL ;NO DISPLAY READY FLAG
383 ;*****
384 001114 104410 000000 ENDS,REGIN
385
386 001120 012767 002000 000436 A11KA: MOV #2000,XPOS ;LOAD X AXIS
387 001126 012767 005000 000432 MOV #5000,YPOS ;LOAD Y AXIS
388 001134 012767 000006 000426 MOV #6,CNTR ;SET UP FOR 6 CHARACTERS
389 001142 012767 001512 000422 TXT1: MOV #TEXT,PNTR ;TEXT A11-K
390 001150 012777 001276 177062 MOV #CHAR3,@VCIV ;LOAD INTERRUPT VECTOR
391 001156 017767 000410 000416 MOV #PNTR,AAR2 ;LOAD
392
393 ;PLOT CHARACTER
394
395 001164 016767 000376 000402 CHAR: MOV YPOS,YPT ;INIT POINT
396 001172 042777 000016 177026 BIC #16,@VCSTAT
397 001200 016777 000360 177022 MOV XPOS,@VCXREG
398 001206 016777 000354 177016 MOV YPOS,@VCYREG
399
400 001214 052777 000010 177004 CHAR5: BIS #10,@VCSTAT ;LOAD STATUS
401 001222 012767 177773 000346 MOV #5,AAR0 ;MATRIX COUNT <ROW>
402 001230 012767 177771 000342 CHAR1: MOV #7,AAR1 ;MATRIX COUNT <COLUMN>
403 001236 117767 000340 000340 MOVPR #AAR2,AAR3 ;GET CHARACTER
404 001244 005267 000332 INC AAR2,AAR3
405 001250 106167 000330 CHAR2: ROLR AAR3
406 001254 100033 BPL CHAR13
407 001256 016777 000302 176744 MOV XPOS,@VCXREG ;LOAD X
408 001264 016777 000276 176740 MOV YPOS,@VCYREG
409 001272 104400 000000 CHAR3: EXITS,BEGIN ;EXIT TO MONITOR. MODULE WAIT FOR INTERRUPT.
410 001276
411
412 001276 000004 000000 001304 ;PIRQS,BEGIN,1S ; QUEUE UP TO CONTINUE AT 1S AND RTI
413
414 1S: TSTR @VCSTAT
415 001304 105777 176716 BML CHAR3
416 001312 017767 176710 176562 MOV @VCSTAT,ACSR ;LOAD VALUE
417 001320 005077 176702 CLR @VCSTAT
418 001324 012767 000011 176554 MOV #11,ERRTYP ;ILLEGAL INTERRUPT
419 ;*****
420 001332 104405 000000 000000 HRDERS,REGIN,NULL ;NO DISPLAY READY
421 ;*****
422 001340 104410 000000 ENDS,BEGIN
```

```

XAAAB0-P11 12-OCT-78 11:40
423 001344* 062767 000070 000214 CHAR13: ADD #70,YPOS ;NEXT POINT
424 001352* 005257 000222 INC AAR1 ;ALL POINTS IN THE COLUMN
425 001356* 001334 BNE CHAR2
426 001360* 016767 000210 000200 MOV YPT,YPOS ;LOAD NEXT COLUMN
427 001366* 062767 000070 000170 ADD #70,XPOS ;ADD SCALCE
428 001374* 005257 000176 INC AAR0 ;DONE ALL COLUMN
429 001400* 001313 BNE CHAR1
430 001402* 062767 000070 000154 ADD #70,XPOS ;YES, NEXT CHARACTER
431 001410* 062767 000002 000154 ADD #2,PNTR
432 001416* 001777 001000 176502 CHAR11: BIT BRIT9,@VCSTAT ;TEST CHANNEL
433 001424* 001004 BNE IS
434 001426* 052777 001000 176572 BIS BRIT9,@VCSTAT ;SET CHANNEL
435 001434* 000403 CHAR20 BR BRIT,@VCSTAT
436 001436* 042777 030000 176562 BIC BRIT,@VCSTAT ;CLEAR CHANNEL
437
438 001444* 005367 000120 CHAR20: DEC CNTR
439 001450* 001017 BNE ZS
440 001452* 042777 000002 176546 BIC #2,@VCSTAT
441 001460* 026767 176540 176352 CMP VCPAS,ICOUNT ;TIME FOR FINAL ITERATION ?
442 001466* 001404 BEQ IS ;YES BRANCH
443 001470* 104413 000000* ENDITS,REGIN ;SIGNAL END OF ITERATION.
444 ;MONITOR SHALL TEST END OF PASS
445 001474* 000167 177420 JMP AAL1KA
446 001500* 005077 176522* 1S: CLR @VCSTAT
447 001504* 104413 000000* ENDITS,REGIN ;SIGNAL END OF ITERATION.
448 ;MONITOR SHALL TEST END OF PASS
449
450 001510* 000167 177434 2S: JMP TXT1
    
```

```

XAAAB0-P11 12-OCT-78 11:40
451 ;TEXT FOR THE AALL-K SCOPE OPTION
452 ;TEXT = "AALL-K"
453
454 001514* 001530* TEXT: A
455 001516* 001530* A
456 001520* 001554* M1
457 001522* 001554* M1
458 001524* 001547* DASH
459 001526* 001535* K
460
461 001530* 176 021 021 A: .BYTE 176,21,21,21,176
462 001533* 021 176 024 K: .BYTE 177,10,24,42,101
463 001535* 177 010 111 S: .BYTE 46,111,111,111,62
464 001540* 042 101 010 DASH: .BYTE 0,10,10,10,0
465 001542* 046 111 177 M1: .BYTE 0,102,177,100,0
466 001545* 111 062
467 001547* 000 010
468 001552* 010 000
469 001554* 000 102
470 001557* 100 000
471
472 001562* .EVEN
473
474 001562* 000000 FATAL: 0
475 001564* 000000 XPOS: 0
476 001566* 000000 YPOS: 0
477 001570* 000000 CNTR: 0
478 001572* 000000 PNTR: 0
479 001574* 000000 YPT: 0
480 001576* 000000 AAR0: 0
481 001600* 000000 AAR1: 0
482 001602* 000000 AAR2: 0
483 001604* 000000 AAR3: 0
484 001606* 000000 TEMP1: 0
485 001610* 000000 TEMP2: 0
486
487 000001 .END
    
```


