KDJ11-B

EEPROM DUTCH LANG LDR
COEEBA0

AH-FF19A-MC
1 OF 1 JUL 1985

digital
MADE IN USA

```
        .TITLE  COEEBA EEPROM DUTCH LANG LDR


                                                        .REM &

                        IDENTIFICATION
                        --------------

        PRODUCT CODE:   AC-FF18A-MC

        PRODUCT NAME:   COEEBAO EEPROM DUTCH LANG LDR

        PRODUCT DATE:   FEBRUARY, 1985

        MAINTAINER:     DIAGNOSTIC ENGINEERING









        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 DOCUMENT.

        NO RESPONSIBILITY IS ASSUMED  FOR  THE  USE  OR  RELIABILITY  OF
        SOFTWARE ON EQUIPMENT THAT IS NOT  SUPPLIED  BY  DIGITAL  OR ITS
        AFFILIATED COMPANIES.

        COPYRIGHT (C) 1985 BY DIGITAL EQUIPMENT CORPORATION


THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

        DIGITAL         PDP             UNIBUS          MASSBUS
        DEC             DECUS           DECTAPE
```

```
                              TABLE OF CONTENTS

                    1.        PROGRAM ABSRACT

                    2.        SYSTEM REQUIREMENTS

                    3.        LOADING AND STARTING PROCEDURES

                    4.        SPECIAL ENVIRONMENTS

                    5.        PROGRAM OPTIONS

                    6.        EXECUTION TIMES

                    7.        ERROR INFORMATION

                    8.        EXAMPLES

                    9.        PROGRAM DESCRIPTION
```

1. PROGRAM ABSTRACT

The KDJ11-B is a PDP-11 CPU that incorporates the J11 chip set as the heart of the processor. It is a quad height Q22 bus module. The KDJ11-B has two on-board ROM's. One of them, the 16-bit addressable ROM, contains the self-test and the boot codes. The other ROM, the 8-bit addressable one, contains the base area with hardware selection parameters, optional bootstraps, optional UFD (User Friendly Diagnostic) system desctiption area, and optional foreign language text.

On units to be shipped to non-English speaking countries, a dummy or "null" language is loaded into the EEPROM. The purpose of this is to disable English language error messages when the system is first installed. If and when the system passes its internal self tests, the user will be instructed to run a UFD (User Friendly Diagnostics) package which will be part of a "country kit" for each seperate language. The UFD package will use the local language for the particular country and, in addition, will load diagnostic and error messages in the local language into the EEPROM, so each subsequent power-up or reboot will have diagnostic and error messages in the user's own language.

The purpose of this program is to load the local language into the EEPROM. If it detects an error, the program will attempt to restore the "old" language, if any and will print a message informing the user of that fact.

2. SYSTEM REQUIREMENTS

Hardware Requirements

To run successfully this utility needs:

1. KDJ11-B CPU module
2. console terminal
3. at least 28K of memory

3. LOADING AND STARTING PROCEDURES

To start-up this program:

1. Boot XXDP+

2. Type "R NAME", where NAME is the name of the BIN or BIC file for this program.

The starting address of the program is 1000.
Note: if trying to restart the program in an arbitrary place after HALT on Break the following registers should be set up:

```
17777572=0      to disable memmory management
17777520=1000   to clear diagnostic mode (bit 8), but still save
                HALT on Break
17777746=400    to flush the cache
```

4. SPECIAL ENVIRONMENTS

The program is not APT compatible.

5. PROGRAM OPTIONS

None.

6. EXECUTION TIMES

The program runs in under 20 seconds.

7. ERROR INFORMATION

7.1 DEFECTIVE BYTE IN EEPROM

After each write, the Byte which should have been written is compared to the Byte in the proper location, and if it is not correct, the following error message is displayed:

```
        EEPROM write error, PCR page n, address mmmmmm.
        Data written qqq, data read rrr.
```

where n is the EEPROM page selected by the Page Control Register (PCR), mmmmmm is the physical address of the bad byte in question, qqq is the byte value that was written out to the address and rrr what was read back in after the write. (should be identical to qqq)

7.2 PROCESSOR NOT KDJ11-B

The program checks the type of CPU it is running on, which must be a KDJ11-B processor (MFPT returns 5 in r0). If not, the following message is printed:

```
        Language area not supported by this processor.
```

7.3 "OLD" BOOT ROM CODE, LANGUAGE AREA NOT SUPPORTED

The program checks to see if the ROM code version is 7.0 or later. Earlier versions do not support the language area in the EEPROM and would print garbage if one was loaded. The program prints the following message in that case:

```
        Current Boot ROM version does not support language area.
```

In addition, the language bit in the setup area of the EEPROM is cleared, to prevent "garbage" from being printed.

7.4 CHECKSUM ERROR IN SETUP AREA

The checksum in the setup area is checked to see if it contains a valid checksum. Also, bytes 6 and 103 (addresses 17765022 and 17765314, respectively) are checked to see if they contain 0 and 252 octal, respectively. If any of these conditions is not met, the following message is printed:

```
        EEPROM checksum error in setup area.
```

No attempt is made to correct a checksum error.

7.5 DIFFERENCES BETWEEN UFD "QUIET" MODE AND "STANDALONE" MODE

When this program is run in UFD "Quiet" mode (which will usually be the case) none of the error messages will appear. If no error is detected, no messages whatsoever are printed. If any error is detected, the program will attempt to restore the UFD and language areas to the state they were in when the program was started. If the restoration was successful, the following message is printed in the user's language:

Unable to load <language>

where <language> is the name of the language. If the restoration was not successful, or there was no local language, the following message is printed.

Unable to load <language> - reverting to U.S. English

where <language> is as above. The program then clears the bit in the EEPROM setup area selecting a local language which means that the ROM English will be used from now on.

8. EXAMPLES

After booting XXDP+ and running the program, no message should appear, just the XXDP dot prompt (.)

If a problem occurred, one of the messages in section 7 should appear.

9. PROGRAM DESCRIPTION

The program consists of a body of code which loads the language into the local language area of the EEPROM. The routine that performs the write first checks the current value of the byte to be written and if it is the same, no write is performed. This is done to extend the life of the EEPROM. The write routine also checks the value in the EEPROM after the write to insure it was written correctly. After a succesful run, no message appears, after an unsuccessful attempt to write any of the bytes in the EEPROM, one of the message in section 7 appears. If run under UFD "Quiet" mode, no message is printed if the program was successful, otherwise one of the messages in 7.5 appear. In both cases, the XXDP prompt appears.

```
                                       .SBTTL  PROGRAM CONSTANTS
000000                                 .ENABL  ABS
                                       .NLIST  MD,CND
                                       .LIST   ME

        177520                         BCSR    =      177520
        177522                         PCR     =      177522
        177522                         PCRLB   =      177522
        165000                         E2PROM  =      165000
        165316                         E2PAR   =      E2PROM+316              ;E2PROM PARITY BYTE
        165006                         E2LLB   =      E2PROM+6                ;LOCAL LANGUAGE BIT IN E2PROM
        166000                         ENDE2R  =      E2PROM+1000             ;LAST ADDRESS OF E2PROM+2
        173002                         RMVTST  =      173002                  ;WORD TO TEST ROM VERSION NUMBER
        025370                         DELAY   =      11000.
        000140                         LNGHDR  =      140                     ;I.D. OF A LANGUAGE AREA
        000040                         UFDHDR  =      040                     ;I.D. OF A UFD BLOCK
        000002                         RETRY   =      2                       ;NUMBER OF ATTEMPTS TO WRITE A
                                                                              ;BYTE IN E2PROM BEFORE GIVING UP
        000004                         MAXERR  =      4                       ;NO. OF ERRORS ALLOWED IN LOCAL
                                                                              ;LANGUAGE TEXT BEFORE QUITTING
        177524                         BDR     =      177524
        000015                         CR      =      15
        000012                         LF      =      12
        000200                         BIT7    =      200
        000100                         BIT6    =      100
        000011                         tab     =      11
        000010                         backsp  =      10
        000040                         space   =      40
        000033                         esc     =      33

        001625                         ROMSZ   =      FLEND-TEXT              ;SIZE IN BYTES OF TEXT TO BE
                                                                              ;LOADED INTO EEPROM


```

```
                                                .SBTTL  CHECK FOR CERTAIN EXCEPTIONS FIRST

        001000                          .=1000

001000  005037  177522                  START:  CLR     @#PCR                   ;SELECT PAGE 0 OF EEPROM
001004  013746  177520                          MOV     @#BCSR,-(SP)            ;SAVE OLD BCSR VALUE
001010  112737  000067  177520                  MOVB    #67,@#BCSR              ;WRITE ENABLE THE E2PROM & ENABLE ROM

001016  000007                                  MFPT                            ;GET PROCESSOR TYPE
001020  020027  000005                          CMP     R0,#5                   ;CHECK TO SEE IF ORION
001024  001404                                  BEQ     1$                      ;YES - CONTINUE
001026                                          .TYPMSG #FMSG2                  ;FIELD-SERVICE MESSAGE
        000001                                  .NARG   NARGS
        000027                                  .NTYPE  NTYPE,#FMSG2
001026  012700  002563                            MOV   #FMSG2,R0
001032  104003                                    EMT   3
001034  000443                                  BR      99$

001036  012700  165000                  1$:     MOV     #E2PROM,R0              ;STARTING ADDRESS TO CHECKSUM
001042  005001                                  CLR     R1                      ;INITIALIZE CHECKSUM
001044  012703  000151                          MOV     #105.,R3                ;NO. OF BYTES TO CKSUM
001050  012005                          201$:   MOV     (R0)+,R5                ;GET A BYTE
001052  042705  177400                          BIC     #177400,R5              ;NO BUS NOISE, THANK YOU.
001056  060501                                  ADD     R5,R1                   ;ACCUMULATE CHECKSUM
001060  077305                                  SOB     R3,201$                 ;CONTINUE TILL DONE
001062  105701                                  TSTB    R1                      ;IS CKSUM 0?
001064  001007                                  BNE     202$                    ;NO, ERROR
001066  105737  165022                          TSTB    @#E2PROM+22             ;BYTE TO TEST FOR VALID ROM, SHOULD BE 0
001072  001004                                  BNE     202$                    ;NO, ERROR
001074  123727  165314  000252                  CMPB    @#E2PROM+314,#252               ;BYTE TO TEST FOR VALID ROM
001102  001404                                  BEQ     300$                    ;GO TO NEXT CHECK IF OK
001104                                  202$:   .TYPMSG #FMSG4                  ;FIELD SERVICE MESSAGE
        000001                                  .NARG   NARGS
        000027                                  .NTYPE  NTYPE,#FMSG4
001104  012700  002737                            MOV   #FMSG4,R0
001110  104003                                    EMT   3
001112  000414                                  BR      99$                     ;QUIT
001114  005067  001304                  300$:   CLR     OLDSIZ                  ;SET FLAG THAT ROM EXISTS, CURRENTLY NO LANGUAGE
001120  012737  000016  177522                  MOV     #7*2,@#PCR              ;SEL. LAST PAGE OF 2K E2PROM, PG0 OF ROM
001126  023727  173002                          CMP     @#RMVTST,(PC)+          ;SEE IF ROM VER. 7 OR LATER (CAN SUPPORT LANGUAGE AREA)
001132  000250                                  CLN
001134  001405                                  BEQ     2$                      ;YES - CONTINUE
001136                                          .TYPMSG #FMSG3
        000001                                  .NARG   NARGS
        000027                                  .NTYPE  NTYPE,#FMSG3
001136  012700  002644                            MOV   #FMSG3,R0
001142  104003                                    EMT   3
001144  000167  000636                  99$:    JMP     QUIT1


                                                .SBTTL  SAVE OLD LANGUAGE/UFD AREA IN CASE IT MUST BE RESTORED

001150  012700  165776                  2$:     MOV     #ENDE2R-2,R0            ;LAST ADDRESS (CKSUM) OF E2PROM
001154  012701  000005                          MOV     #5,R1                   ;NO. OF BYTES IN HEADER TO CHECKSUM
001160  010005                                  MOV     R0,R5                   ;SAVE ADDRESS
001162  005003                                  CLR     R3                      ;
001164  111004                          4$:     MOVB    (R0),R4                 ;GET A BYTE
001166  060403                                  ADD     R4,R3                   ;ACCUMULATE CHECKSUM
```

```
001170  005740                          TST     -(R0)           ;CORRECT ADDRESS
001172  077104                          SOB     R1,4$           ;LOOP FOR 5 BYTES
001174  105703                          TSTB    R3              ;IF NOT ZERO, NO LANGUAGE LOADED
001176  001131                          BNE     WRLANG          ;NON-EXISTANT OR CORRUPTED LANGUAGE - SKIP

001200  014504                          MOV     -(R5),R4        ;HIGH BYTE OF BYTE COUNT
001202  014546                          MOV     -(R5),-(SP)     ;LOW BYTE OF BYTE COUNT
001204  110466  000001                  MOVB    R4,1(SP)        ;SET UPPER BYTES OF SIZE
001210  042704  177437                  BIC     #177437,R4      ;EXTRACT ID CODE
001214  012601                          MOV     (SP)+,R1        ;GET SIZE BACK
001216  042701  160000                  BIC     #160000,R1      ;R1 NOW CONTAINS SIZE OF BLOCK IN BYTES
001222  062701  000005                  ADD     #5,R1           ;ADD BYTE COUNT FOR HEADER BLOCK
001226  120427  000040                  CMPB    R4,#UFDHDR      ;SEE IF IT IS A UFD BLOCK
001232  001013                          BNE     LANG            ;NO, CHECK FOR A LANGUAGE
001234  010104                          MOV     R1,R4           ;SAVE SIZE
001236  012702  004733                  MOV     #BUFF,R2        ;ADDRESS OF SAVE BUFFER
001242  004767  000666                  CALL    MOVROM          ;MOVE UFD AREA TO MEMORY
001246  001105                          BNE     WRLANG          ;BAD CKSUM, QUIT
                                                                ;NOTE - R3 CONTAINS CHECKSUM OF BLOCK AND HEADER
                                                                ;HOWEVER THE CHECKSUM OF HEADER IS ALREADY KNOWN
                                                                ;TO BE 0 SO R3 IS A VALID CHECK OF UFD BLOCK
001250  010167  001150                  MOV     R1,OLDSIZ       ;SAVE TOTAL SIZE
001254  010167  001146                  MOV     R1,UFDSIZ       ;SAVE SIZE OF UFD AREA
001260  000500                          BR      WRLANG

001262  120427  000140          LANG:   CMPB    R4,#LNGHDR      ;IS THIS A LANGUAGE HEADER?
001266  001075                          BNE     WRLANG          ;NO - QUIT
001270  010167  001130                  MOV     R1,OLDSIZ       ;SAVE SIZE FOR NOW
001274  062701  000005                  ADD     #5,R1           ;ADD SIZE OF (POSSIBLE) UFD HEADER
001300  004767  001036                  CALL    ROMADR          ;SET UP PCR AND R0
001304  005003                          CLR     R3              ;INITIALIZE CKSUM
001306  004767  001002                  CALL    REAROM          ;GET A BYTE
001312  004767  000776                  CALL    REAROM          ;GET A BYTE
001316  004767  000772                  CALL    REAROM          ;GET A BYTE
001322  010546                          MOV     R5,-(SP)        ;SAVE LOW BYTE OF SIZE FOR LATER
001324  004767  000764                  CALL    REAROM          ;GET A BYTE
001330  110566  000001                  MOVB    R5,1(SP)        ;SAVE HIGH BYTE OF SIZE AND ID
001334  004767  000754                  CALL    REAROM          ;GET A BYTE
001340  116600  000001                  MOVB    1(SP),R0        ;GET I.D.
001344  012601                          MOV     (SP)+,R1        ;GET SIZE
001346  105703                          TSTB    R3              ;SEE IF VALID CKSUM
001350  001025                          BNE     1$              ;NO - WE HAVE LANGUAGE ONLY.

001352  042700  177437                  BIC     #177437,R0      ;GET ID ONLY
001356  120027  000040                  CMPB    R0,#UFDHDR      ;IS THIS A UFD BLOCK?
001362  001020                          BNE     1$              ;NO, IGNORE IT.

                                ;WE HAVE BOTH A LANGUAGE AREA AND A UFD BLOCK.  SAVE THE UFD BLOCK.

001364  042701  160000                  BIC     #160000,R1      ;GET RID OF ID
001370  062701  000005                  ADD     #5,R1           ;SIZE OF HEADER
001374  010104                          MOV     R1,R4           ;BYTE COUNT TO MOVE
001376  010167  001024                  MOV     R1,UFDSIZ       ;SAVE UFD SIZE
001402  066701  001016                  ADD     OLDSIZ,R1       ;ADD SIZE OF LANGUAGE AREA
001406  012702  004733                  MOV     #BUFF,R2        ;MEMORY ADDRESS TO SAVE TO
001412  004767  000516                  CALL    MOVROM          ;SAVE UFD AREA
001416  001404                          BEQ     2$              ;YES, IT IS VALID, CONTINUE
```

```
001420  005067  001002                    CLR    UFDSIZ           ;NO UFD AREA
001424  012702  004733          1$:       MOV    #BUFF,R2         ;RESET R2
001430  016701  000770          2$:       MOV    OLDSIZ,R1        ;SIZE OF LANGUAGE AREA
001434  010104                            MOV    R1,R4            ;BYTES TO MOVE
001436  066767  000764  000760            ADD    UFDSIZ,OLDSIZ    ;OLDSIZ IS THE TOTAL SIZE
001444  004767  000464                    CALL   MOVROM           ;SAVE LANGUAGE AREA
001450  001404                            BEQ    WRLANG           ;LANGUAGE IS GOOD
001452  005067  000746                    CLR    OLDSIZ           ;NO LANGUAGE
001456  005067  000744                    CLR    UFDSIZ           ;NO UFD AREA

                                ;GENERATE CHECKSUM FOR FOREIGN LANGUAGE TEST FILE & WRITE TO THE MEMORY IMAGE

001462  012700  003106          WRLANG:   MOV    #TEXT,R0         ;ADDRESS OF BEGINNING OF TEXT
001466  005001                            CLR    R1               ;INIT CHECKSUM
001470  112002                  25$:      MOVB   (R0)+,R2         ;READ A BYTE
001472  160201                            SUB    R2,R1            ;ACCUMULATE CHECKSUM
001474  020027  004725                    CMP    R0,#CKSUM        ;FINISHED ALL TEXT ?
001500  001373                            BNE    25$              ;NO-CONTINUE
001502  110110                            MOVB   R1,(R0)          ;WRITE THE CHECKSUM

                                          .SBTTL  LOAD LOCAL LANGUAGE INTO E2PROM

                                ;WRITE UFD & LOCAL LANGUAGE BLOCKS

001504  016701  000716                    MOV    UFDSIZ,R1        ;GET THE LENGTH OF THE UFD
001510  062701  001625                    ADD    #ROMSZ,R1        ;... & THE TEXT AREA
001514  004767  000622                    JSR    PC,ROMADR        ;COMPUTE E2PROM PAGE AND ADDR
001520  016701  000702                    MOV    UFDSIZ,R1        ;SIZE OF UFD AREA TO SAVE
001524  001406                            BEQ    40$              ;NO UFD AREA - SKIP
001526  012702  004733                    MOV    #BUFF,R2         ;ADDRESS OF BEGINNING OF UFD AREA
001532  112205                  35$:      MOVB   (R2)+,R5         ;GET SOME DATA
001534  004767  000126                    CALL   E2WRIT           ;GO WRITE IT
001540  077104                            SOB    R1,35$           ;FINISHED UFD?
                                                                  ;YES-DO LANGUAGE
001542  012702  003106          40$:      MOV    #TEXT,R2         ;ADDRESS OF EEPROM LANGUAGE TEXT
001546  012701  001625                    MOV    #ROMSZ,R1        ;BYTES TO MOVE
001552  112205                  50$:      MOVB   (R2)+,R5         ;GET SOME DATA
001554  004767  000106                    CALL   E2WRIT           ;WRITE A BYTE
001560  077104                            SOB    R1,50$           ;ARE WE DONE?
                                                                  ;YES - EXIT
001562  112705  000200                    MOVB   #BIT7,R5         ;TURN ON LOCAL LANGUAGE BIT IN
                                                                  ;SETUP AREA, THEN EXIT

001566  105037  177522          EXIT:     CLRB   @#PCRLB          ;SELECT PAGE 0
001572  012700  165006                    MOV    #E2LLB,R0        ;E2PROM WORD CONTAINING LOCAL LANG. BIT
001576  111001                            MOVB   (R0),R1
001600  142701  177577                    BICB   #↑CBIT7,R1       ;GET CURRENT LOCAL LANGUAGE BIT
001604  120501                            CMPB   R5,R1            ;SEE IF BIT ALREADY CORRECT
001606  001415                            BEQ    EXIT1            ;YES, JUST RETURN
001610  112701  000200                    MOVB   #BIT7,R1         ;LOCAL LANGUAGE BIT
001614  111005                            MOVB   (R0),R5          ;GET OLD WORD AGAIN
001616  074105                            XOR    R1,R5            ;FLIP THE BIT
001620  004767  000336                    CALL   WRBYTE           ;CHANGE LOCAL LANGUAGE BIT IN E2PROM
001624  001006                            BNE    EXIT1            ;WOULD NOT WRITE, JUST GIVE UP
001626  012700  165316                    MOV    #E2PAR,R0        ;ADDRESS OF CKSUM BYTE
001632  111005                            MOVB   (R0),R5          ;GET OLD CKSUM BYTE
001634  074105                            XOR    R1,R5            ;CORRECT THE CKSUM
```

```
001636  004767  000320                  CALL    WRBYTE          ;UPDATE E2ROM

001642                          EXIT1:  .FRCTYP #CRLF           ;COMPLETE LINE
        000001                          .NARG   NARGS
        000027                          .NTYPE  NTYPE,#CRLF
001642  012700  002560                    MOV   #CRLF,R0
001646  104044                            EMT   44
001650  142716  000060                  BICB    #60,(SP)        ;BE SURE ROM IS DISABLED
001654  012637  177520                  MOV     (SP)+,@#BCSR    ;RESTORE BCSR
001660  005037  177522                  CLR     @#PCR           ;
001664  000207                          RTS     PC

001666  004767  000270          E2WRIT: CALL    WRBYTE          ;WRITE THE BYTE TO E2PROM
001672  001431                          BEQ     3$              ;OK THIS TIME
001674  005267  000522                  INC     WERR            ;FLAG BAD BYTE

001700  026727  000516  000004          CMP     WERR,#MAXERR    ;CHECK TO SEE IF PAST THE MAXIMUM ERROR
001706  003036                          BGT     QUIT            ;LIMIT OF BAD BYTES ALLOWED

001710  020227  003203                  CMP     R2,#M001        ;CHECK TO SEE IF ERROR IS IN MESSAGE
001714  101433                          BLOS    QUIT            ;BYTE COUNT (MUST BE CORRECT)

001716  020227  004724                  CMP     R2,#MEND1       ;CHECK TO BE SURE DICTIONARY AND UFD
001722  101030                          BHI     QUIT            ;BLOCKS ARE NOT CORRUPTED

001724  132705  000140                  BITB    #140,R5         ;CHECK TO SEE IF IT SHOULD BE A CONTROL
001730  001425                          BEQ     QUIT            ;CODE (POSSIBLY DICTIONARY ENTRY)

001732  132710  000140                  BITB    #140,(R0)       ;IF CONTROL CODE (DICTIONARY REFERENCE
001736  001422                          BEQ     QUIT            ;PERHAPS) CALL IT QUITS

001740  111004                          MOVB    (R0),R4         ;WE WILL LIVE WITH THIS ERROR, CORRECT
001742  116703  002757                  MOVB    CKSUM,R3        ;THE CHECKSUM TO ACCOUNT FOR NEW VALUE
001746  060503                          ADD     R5,R3           ;CANCEL OUT WHAT WAS SUPPOSED TO BE
001750  160403                          SUB     R4,R3           ;CORRECT FOR ERRONEOUS VALUE
001752  110367  002747                  MOVB    R3,CKSUM        ;PUT BACK CORRECTED VALUE

001756  062700  000002          3$:     ADD     #2,R0           ;INCREMENT LOCATION
001762  020027  166000                  CMP     R0,#ENDE2R      ;FINISHED THIS PAGE ?
001766  001005                          BNE     10$             ;NO-RETURN
001770  012700  165000                  MOV     #E2PROM,R0      ;YES-RESET ADDRESS
001774  062737  000002  177522          ADD     #2,@#PCR        ;INCREMENT PCR TO NEXT PAGE
002002  000207                  10$:    RETURN

002004  005726                  QUIT:   TST     (SP)+           ;CORRECT STACK
002006  032737  000100  000052  QUIT1:  BIT     #BIT6,@#52      ;SEE IF UFD QUIET
002014  001403                          BEQ     5$              ;NO
002016                                  .FRCTYP #MSG000         ;MESSAGE FOR USER IN HIS OWN LANGUAGE
        000001                          .NARG   NARGS
        000027                          .NTYPE  NTYPE,#MSG000
002016  012700  003006                    MOV   #MSG000,R0
002022  104044                            EMT   44
002024  016701  000374          5$:     MOV     OLDSIZ,R1
002030  100704                          BMI     EXIT1           ;ERROR WAS NOT ORION OR CKSUM ERROR, DO NOT
                                                                ;TRY TO CLEAR LANGUAGE BIT
002032  001427                          BEQ     40$             ;IF NO OLD LANGUAGE TO RESTORE
002034  004767  000302                  JSR     PC,ROMADR       ;COMPUTE STARTING ADDRESS OF OLD LANG IN E2PROM
```

```
002040  012702  004733                          MOV     #BUFF,R2        ;STARTING ADDRESS OF OLD LANGUAGE TEXT
002044  112205                          10$:    MOVB    (R2)+,R5        ;GET A BYTE
002046  004767  000110                          CALL    WRBYTE          ;WRITE IT OUT
002052  001017                                  BNE     40$             ;IF ERROR, GIVE UP
002054  062700  000002                          ADD     #2,R0           ;INCREMENT LOCATION
002060  020027  166000                          CMP     R0,#ENDE2R      ;FINISHED THIS PAGE ?
002064  001005                                  BNE     20$             ;NO-CONTINUE
002066  012700  165000                          MOV     #E2PROM,R0      ;YES-RESET ADDRESS
002072  062737  000002  177522                  ADD     #2,@#PCR        ;INCREMENT PCR TO NEXT PAGE
002100  077117                          20$:    SOB     R1,10$          ;LOOP UNTIL DONE
002102  026767  000320  000314                  CMP     UFDSIZ,OLDSIZ   ;IF THE SAME THEN NO LANGUAGE
002110  001254                                  BNE     EXIT1           ;IF LANGUAGE, LEAVE E2PROM LANG. BIT AS IT WAS
002112  005005                          40$:    CLR     R5              ;TURN OFF LOCAL LANGUAGE BIT IN E2PROM
002114  036737  175760  000052                  BIT     BIT6,@#52       ;SEE IF UFD QUIET
002122  001621                                  BEQ     EXIT            ;NO
002124                                          .FRCTYP #MSG001
        000001                                  .NARG   NARGS
        000027                                  .NTYPE  NTYPE,#MSG001
002124  012700  003046                            MOV   #MSG001,R0
002130  104044                                    EMT   44
002132  000615                                  BR      EXIT            ;AND CALL IT A DAY

                                                .SBTTL  PROGRAM SUBROUTINES

                                        ;MOVROM - MOVE BYTES FROM EEPROM TO MEMORY
                                        ;ENTRY- R1 = STARTING ADDRESS IN EEPROM (# OF BYTES FROM END)
                                        ;       R2 = ADDRESS OF MEMORY BUFFER
                                        ;       R4 = # OF BYTES TO MOVE
                                        ;EXIT   R1 - UNCHANGED
                                        ;       R2 - UPDATED MEMORY ADDRESS
                                        ;       R3 = (BYTE) 0 IF VALID CKSUM
                                        ;       "Z" FLAG SET IF CKSUM VALID

002134  010403                          MOVROM: MOV     R4,R3           ;SAVE R4
002136  004767  000200                          CALL    ROMADR          ;LOAD PCR AND R0 WITH LANGUAGE START AREA
002142  010304                                  MOV     R3,R4           ;RESTORE BYTE COUNT
002144  005003                                  CLR     R3              ;INIT CHECKSUM
002146  004767  000142                  5$:     CALL    REAROM          ;GET A BYTE
002152  110522                                  MOVB    R5,(R2)+        ;SAVE IT
002154  077404                                  SOB     R4,5$           ;LOOP TILL DONE
002156  105703                                  TSTB    R3              ;IS CHECKSUM GOOD?
002160  000207                                  RETURN

002162  120510                          WRBYTE: CMPB    R5,(R0)         ;IS THE NEW DATA DIFFERENT ?
002164  001452                                  BEQ     10$             ;NO-DO NOT WRITE OVER

002166  012703  000002                          MOV     #RETRY,R3
002172  010510                          1$:     MOV     R5,(R0)         ;WRITE A LOCATION
002174  012704  025370                          MOV     #DELAY,R4       ;11 MS WAIT
002200  077401                                  SOB     R4,.            ;WASTE TIME
002202  120510                                  CMPB    R5,(R0)         ;SEE IF IT TOOK
002204  001442                                  BEQ     10$             ;YES, ALL OKAY
002206  077307                                  SOB     R3,1$           ;IF AT FIRST YOU DON'T SUCCEED...
002210  113704  177522                          MOVB    @#PCRLB,R4      ;PCR PAGE OF BAD BYTE
002214  106204                                  ASRB    R4              ;CONVERT TO PAGE #
002216  062704  000060                          ADD     #'0,R4          ;CONVERT TO OCTAL
002222  110467  000237                          MOVB    R4,FMSG1A       ;STORE IT FOR PRINTING
```

PROGRAM SUBROUTINES

```
571 002226  010046                              MOV     R0,-(SP)           ;SAVE ROM ADDRESS
572 002230                                      .ITOA   ,#FMSG1B           ;CONVERT ROM ADDRESS TO OCTAL
            000002                              .NARG   NARGS
            000027                                 .NTYPE       NTYPE,#FMSG1B
    002230  012701  002500                         MOV  #FMSG1B,R1
    002234  104030                                 EMT  30
573 002236                                      .TYPMSG #FMSG1             ;PRINT OUT FIRST PART OF MESSAGE
            000001                              .NARG   NARGS
            000027                              .NTYPE  NTYPE,#FMSG1
    002236  012700  002430                        MOV   #FMSG1,R0
    002242  104003                                EMT   3
574 002244  042705  177400                      BIC     #177400,R5         ;MAKE SURE R5 IS POSITIVE AND A BYTE
575 002250                                      .ITOA   R5,#DUMMY1         ;CONVERT TO OCTAL
            000002                              .NARG   NARGS
            000005                                 .NTYPE       NTYPE,R5
    002250  010500                                 MOV  R5,R0
            000027                                 .NTYPE       NTYPE,#DUMMY1
    002252  012701  002526                         MOV  #DUMMY1,R1
    002256  104030                                 EMT  30
576 002260                                      .TYPMSG #FMSG1C            ;PRINT OUT LAST 3 DIGITS OF NUMBER & MESSAGE
            000001                              .NARG   NARGS
            000027                              .NTYPE  NTYPE,#FMSG1C
    002260  012700  002531                        MOV   #FMSG1C,R0
    002264  104003                                EMT   3
577 002266  013600                              MOV     @(SP)+,R0          ;GET BYTE AT ROM ADDRESS
578 002270  042700  177400                      BIC     #177400,R0         ;GET RID OF BUS NOISE
579 002274                                      .ITOA   ,#DUMMY2           ;CONVERT TO OCTAL
            000002                              .NARG   NARGS
            000027                                 .NTYPE       NTYPE,#DUMMY2
    002274  012701  002551                         MOV  #DUMMY2,R1
    002300  104030                                 EMT  30
580 002302                                      .TYPMSG #FMSG1D            ;PRINT LOWER 3 BYTES & REST OF MESSAGE
            000001                              .NARG   NARGS
            000027                              .NTYPE  NTYPE,#FMSG1D
    002302  012700  002554                        MOV   #FMSG1D,R0
    002306  104003                                EMT   3
581 002310  000244                              CLZ                        ;COULDN'T DO IT, SET ERROR FLAG
582 002312  000207                      10$:    RETURN
583
584                                     ;REAROM - READS A BYTE FROM E2PROM ADDRESS (R0)+ INTO R5. AUTOMATICLY ADJUSTS
585                                     ;PCRLB.  UPDATES CKSUM IN R3
586                                     ;       ENTRY - R0      ADDRESS IN ROM TO READ FROM
587                                     ;               R3      PARTIAL CKSUM
588                                     ;               PCRLB   CORRECT VALUE FOR BYTE TO READ
589                                     ;       EXIT    R0      ADDRESS OF NEXT BYTE
590                                     ;               R3      UPDATED CKSUM
591                                     ;               R5      BYTE READ
592                                     ;               PCRLB   CORRECT VALUE FOR NEXT BYTE
593
594 002314  012005                      REAROM: MOV     (R0)+,R5           ;GET A BYTE & UPDATE ADDR. BY 2
595 002316  060503                              ADD     R5,R3              ;UPDATE CKSUM
596 002320  020027  166000                      CMP     R0,#ENDE2R         ;SEE IF WE SHOULD SWITCH PAGES
597 002324  001005                              BNE     10$                ;NO
598 002326  012700  165000                      MOV     #E2PROM,R0         ;YES - GO TO START OF PAGE
599 002332  062737  000002  177522              ADD     #2,@#PCR                  ;ADVANCE A PAGE
600 002340  000207                      10$:    RETURN
601
```

```

                                  ;ROMADR - CALCULATE PAGE OFFSET FROM END OF ROM GIVEN SIZE IN BYTES
                                  ;        ENTRY - R1      SIZE IN BYTES
                                  ;        EXIT -  R0      INITIAL ADDRESS FOR FIRST BYTE IN ROM
                                  ;                R1      SIZE IN BYTES
                                  ;                PCRLB   CORRECT VALUE FOR FIRST BYTE IN ROM

002342  010100                    ROMADR: MOV     R1,R0           ;COPY BYTE COUNT
002344  010105                            MOV     R1,R5           ;SECOND COPY
002346  072527  177770                    ASH     #-8.,R5         ;DIVIDE BYTE COUNT BY 256. BYTE PAGES
002352  012704  000010                    MOV     #7+1,R4         ;LAST PAGE IN 2 K PART + 1
002356  160504                            SUB     R5,R4           ;STARTING PAGE NUMBER

002360  042700  177400                    BIC     #177400,R0      ;LEAVE ONLY BITS 7:0
002364  006300                            ASL     R0              ;DOUBLE VALUE
002366  001003                            BNE     20$
002370  012700  165000                    MOV     #E2PROM,r0      ;
002374  000406                            BR      30$             ;IF 0

002376  005400                    20$:    NEG     R0              ;MAKE STARTING ADDRESS BITS 8:0
002400  042700  177000                    BIC     #177000,R0      ;
002404  052700  165000                    BIS     #E2PROM,R0      ;MAKE A E2PROM ADDRESS
002410  005304                            DEC     R4              ;DECREMENT PAGE NUMBER BY 1

002412  006304                    30$:    ASL     R4              ;MAKE PAGE NUMBER CORRECT FOR PCR
002414  110437  177522                    MOVB    R4,@#PCRLB      ;CORRECT PAGE IN PCRLB
002420  000207                            RTS     PC              ;RETURN

002422  000000                    WERR:   0                       ;FLAG FOR BAD BYTE
002424  177777                    OLDSIZ: -1                      ;>0 - SIZE IN BYTES OF OLD LANGUAGE, 0 IF NO
                                                                  ;LANGUAGE, -1 IF E2PROM MAY BE BAD/NONEXISTANT
002426  000000                    UFDSIZ: 0                       ;SIZE IN BYTES OF OLD UFD AREA

                                  .SBTTL  "FIELD SERVICE MODE" ERROR MESSAGES

                                  .ENABL LC
002430     105     105     120    FMSG1:  .ASCII  /EEPROM write error, PCR page /
002433     122     117     115
002436     040     167     162
002441     151     164     145
002444     040     145     162
002447     162     157     162
002452     054     040     120
002455     103     122     040
002460     160     141     147
002463     145     040
002465     130     054     040    FMSG1A: .ASCII  /X, address /
002470     141     144     144
002473     162     145     163
002476     163     040
002500                            FMSG1B: .BLKB   6               ;FOR ADDRESS
002506     015     012     104            .ASCIZ  <CR><LF>/Data written /
002511     141     164     141
002514     040     167     162
002517     151     164     164
002522     145     156     040
002525     000
```

```
   642 002526                         DUMMY1: .BLKB   3               ;3 UPPER BYTES NOT TO BE PRINTED
   643 002531                         FMSG1C: .BLKB   3
   644 002534    054    040    104            .ASCIZ  /. Data read /
       002537    141    164    141
       002542    040    162    145
       002545    141    144    040
       002550    000
   645 002551                         DUMMY2: .BLKB   3               ;3 UPPER BYTES NOT TO BE PRINTED
   646 002554                         FMSG1D: .BLKB   3
   647 002557    056                          .ASCII  /./
   648 002560    015    012    000    CRLF:   .ASCIZ  <CR><LF>
   649 002563    114    141    156    FMSG2:  .ASCIZ  /Language Area not supported on this processor./<CR><LF>
       002566    147    165    141
       002571    147    145    040
       002574    101    162    145
       002577    141    040    156
       002602    157    164    040
       002605    163    165    160
       002610    160    157    162
       002613    164    145    144
       002616    040    157    156
       002621    040    164    150
       002624    151    163    040
       002627    160    162    157
       002632    143    145    163
       002635    163    157    162
       002640    056    015    012
       002643    000
   650 002644    103    165    162    FMSG3:  .ASCIZ  /Current boot ROM version does not support language area./<CR><LF>
       002647    162    145    156
       002652    164    040    142
       002655    157    157    164
       002660    040    122    117
       002663    115    040    166
       002666    145    162    163
       002671    151    157    156
       002674    040    144    157
       002677    145    163    040
       002702    156    157    164
       002705    040    163    165
       002710    160    160    157
       002713    162    164    040
       002716    154    141    156
       002721    147    165    141
       002724    147    145    040
       002727    141    162    145
       002732    141    056    015
       002735    012    000
   651 002737    103    150    145    FMSG4:  .ASCIZ  /Checksum error in EEPROM setup area./<CR><LF>
       002742    143    153    163
       002745    165    155    040
       002750    145    162    162
       002753    157    162    040
       002756    151    156    040
       002761    105    105    120
       002764    122    117    115
       002767    040    163    145
```

"FIELD SERVICE MODE" ERROR MESSAGES

```
        002772      164     165     160
        002775      040     141     162
        003000      145     141     056
        003003      015     012     000
    652                                     .SBTTL  TRANSLATED LOADER ERROR MESSAGES
    653 003006      015     114     141     MSG000: .ASCIZ  <CR>!Laden Nederlands niet mogelijk!
        003011      144     145     156
        003014      040     116     145
        003017      144     145     162
        003022      154     141     156
        003025      144     163     040
        003030      156     151     145
        003033      164     040     155
        003036      157     147     145
        003041      154     151     152
        003044      153     000
    654 003046      040     055     040     MSG001: .ASCIZ  ! - Systeem gebruikt US Engels.!<CR>
        003051      123     171     163
        003054      164     145     145
        003057      155     040     147
        003062      145     142     162
        003065      165     151     153
        003070      164     040     125
        003073      123     040     105
        003076      156     147     145
        003101      154     163     056
        003104      015     000
    655                                     .SBTTL  START OF AREA TO BE LOADED INTO E2PROM
    656
    657                                     .SBTTL  Nederlands LANGUAGE TEXT
    658
    659 003106      075                     TEXT:   .BYTE   M001-TEXT
    660 003107      013                             .BYTE   M002-M001
    661 003110      002                             .BYTE   M003-M002
    662 003111      005                             .BYTE   M004-M003
    663 003112      006                             .BYTE   M005-M004
    664 003113      005                             .BYTE   M006-M005
    665 003114      002                             .BYTE   M007-M006
    666 003115      002                             .BYTE   M010-M007
    667 003116      002                             .BYTE   M011-M010
    668 003117      000                             .BYTE   M012-M011
    669 003120      000                             .BYTE   M013-M012
    670 003121      000                             .BYTE   M014-M013
    671 003122      000                             .BYTE   M015-M014
    672 003123      000                             .BYTE   M016-M015
    673 003124      000                             .BYTE   M017-M016
    674 003125      000                             .BYTE   M020-M017
    675 003126      037                             .BYTE   M021-M020
    676 003127      030                             .BYTE   M022-M021
    677 003130      036                             .BYTE   M023-M022
    678 003131      112                             .BYTE   M024-M023
    679 003132      012                             .BYTE   M025-M024
    680 003133      001                             .BYTE   M026-M025
    681 003134      031                             .BYTE   M027-M026
    682 003135      005                             .BYTE   M030-M027
    683 003136      007                             .BYTE   M031-M030
    684 003137      020                             .BYTE   M032-M031
```

```
003140   002                        .BYTE   M033-M032
003141   045                        .BYTE   M034-M033
003142   000                        .BYTE   M035-M034
003143   001                        .BYTE   M036-M035
003144   000                        .BYTE   M037-M036
003145   002                        .BYTE   M040-M037
003146   033                        .BYTE   M041-M040
003147   000                        .BYTE   M042-M041
003150   014                        .BYTE   M043-M042
003151   026                        .BYTE   M044-M043
003152   026                        .BYTE   M045-M044
003153   024                        .BYTE   M046-M045
003154   022                        .BYTE   M047-M046
003155   040                        .BYTE   M050-M047
003156   031                        .BYTE   M051-M050
003157   030                        .BYTE   M052-M051
003160   021                        .BYTE   M053-M052
003161   030                        .BYTE   M054-M053
003162   022                        .BYTE   M055-M054
003163   026                        .BYTE   M056-M055
003164   076                        .BYTE   M057-M056
003165   012                        .BYTE   M060-M057
003166   000                        .BYTE   M061-M060
003167   010                        .BYTE   M062-M061
003170   002                        .BYTE   M063-M062
003171   013                        .BYTE   M064-M063
003172   024                        .BYTE   M065-M064
003173   003                        .BYTE   M066-M065
003174   024                        .BYTE   M067-M066
003175   046                        .BYTE   M070-M067
003176   006                        .BYTE   M071-M070
003177   003                        .BYTE   M072-M071
003200   057                        .BYTE   M073-M072
003201   003                        .BYTE   M074-M073
003202   041                        .BYTE   MEND1-M074
003203   116   145   144   M001:    .ASCIZ  !Nederlands!
003206   145   162   154
003211   141   156   144
003214   163   000
003216   077   000         M002:    .ASCIZ  !?!
003220   110   105   114   M003:    .ASCIZ  !HELP!
003223   120   000
003225   123   124   101   M004:    .ASCIZ  !START!
003230   122   124   000
003233   124   117   117   M005:    .ASCIZ  !TOON!
003236   116   000
003240   177   000         M006:    .ASCIZ  <177>           ;Setup command
003242   177   000         M007:    .ASCIZ  <177>           ;Map command
003244   177   000         M010:    .ASCIZ  <177>           ;Test command
003246                     M011:
003246                     M012:
003246                     M013:
003246                     M014:
003246                     M015:
003246                     M016:
003246                     M017:
003246   101   160   160   M020:    .ASCII  !Apparaat Eenheden Beschrijving!<CR>
```

```
        003251     141     162     141
        003254     141     164     040
        003257     105     145     156
        003262     150     145     144
        003265     145     156     040
        003270     102     145     163
        003273     143     150     162
        003276     151     152     166
        003301     151     156     147
        003304     015
    736 003305     124     157     157  M021:   .ASCII  !Toon opstartprogramma's!<CR>
        003310     156     040     157
        003313     160     163     164
        003316     141     162     164
        003321     160     162     157
        003324     147     162     141
        003327     155     155     141
        003332     047     163     015
    737 003335     123     171     163  M022:   .ASCII  !Systeem wordt opgestart vanaf !
        003340     164     145     145
        003343     155     040     167
        003346     157     162     144
        003351     164     040     157
        003354     160     147     145
        003357     163     164     141
        003362     162     164     040
        003365     166     141     156
        003370     141     146     040
    738 003373     015     117     160  M023:   .ASCII  <CR>!Opdracht Beschrijving!<CR><CR>!START!<TAB>! Laad en star!
        003376     144     162     141
        003401     143     150     164
        003404     040     102     145
        003407     163     143     150
        003412     162     151     152
        003415     166     151     156
        003420     147     015     015
        003423     123     124     101
        003426     122     124     011
        003431     040     114     141
        003434     141     144     040
        003437     145     156     040
        003442     163     164     141
        003445     162
    739 003446     164     040     163          .ASCII  !t systeem vanaf apparaat!<CR>!TOON!<TAB>! !
        003451     171     163     164
        003454     145     145     155
        003457     040     166     141
        003462     156     141     146
        003465     040     141     160
        003470     160     141     162
        003473     141     141     164
        003476     015     124     117
        003501     117     116     011
        003504     040
    740 003505     015     103     157  M024:   .ASCII  <CR>!Controle !
        003510     156     164     162
        003513     157     154     145
        003516      040
    741 003517      057                     M025:   .ASCII  '/'
    742 003520      104     162     165     M026:   .ASCII  !Druk op de RETURN-toets: !
        003523      153     040     157
        003526      160     040     144
        003531      145     040     122
        003534      105     124     125
        003537      122     116     055
        003542      164     157     145
        003545      164     163     072
        003550      040
    743 003551      106     157     165     M027:   .ASCII  !Fout !
        003554      164     040
    744 003556      040     141     144     M030:   .ASCII  ! adres !
        003561      162     145     163
        003564      040
    745 003565      102     145     172     M031:   .ASCII  !Bezig met testen!
        003570      151     147     040
        003573      155     145     164
        003576      040     164     145
        003601      163     164     145
        003604      156
    746 003605      060     055             M032:   .ASCII  /0-/
    747 003607      015     124     171     M033:   .ASCII  <CR>!Typ een opdracht en druk op RETURN: !
        003612      160     040     145
        003615      145     156     040
        003620      157     160     144
        003623      162     141     143
        003626      150     164     040
        003631      145     156     040
        003634      144     162     165
        003637      153     040     157
        003642      160     040     122
        003645      105     124     125
        003650      122     116     072
        003653      040
    748 003654                              M034:
    749 003654      011                     M035:   .BYTE   TAB
    750 003655                              M036:
    751 003655      015     040             M037:   .BYTE   CR,SPACE
    752 003657      102     145     172     M040:   .ASCII  !Bezig met opstarten via ROM!
        003662      151     147     040
        003665      155     145     164
        003670      040     157     160
        003673      163     164     141
        003676      162     164     145
        003701      156     040     166
        003704      151     141     040
        003707      122     117     115
    753 003712                              M041:
    754 003712      015     102     145     M042:   .ASCII  <CR>!Bericht 06!<CR>
        003715      162     151     143
        C03720      150     164     040
        003723      060     066     015
    755 003726      101     141     !56     M043:   .ASCII  !Aandrijver niet gereed!
        003731      144     162     151
        003734      152     166     145
        003737      162     040     156
        003742      151     145     164
        003745      040     147     145
        003750      162     145     145
        003753      144
    756 003754      115     145     144  M044:    .ASCII  !Media niet opstartbaar!
        003757      151     141     040
        003762      156     151     145
        003765      164     040     157
        003770      160     163     164
        003773      141     162     164
        003776      142     141     141
        004001      162
    757 004002      107     145     145  M045:    .ASCII  !Geen schijf aanwezig!
        004005      156     040     163
        004010      143     150     151
        004013      152     146     040
        004016      141     141     156
        004021      167     145     172
        004024      151     147
    758 004026      107     145     145  M046:    .ASCII  !Geen tape aanwezig!
        004031      156     040     164
        004034      141     160     145
        004037      040     141     141
        004042      156     167     145
        004045      172     151     147
    759 004050      116     151     145  M047:    .ASCII  !Niet-bestaande besturingsmodule,!
        004053      164     055     142
        004056      145     163     164
        004061      141     141     156
        004064      144     145     040
        004067      142     145     163
        004072      164     165     162
        004075      151     156     147
        004100      163     155     157
        004103      144     165     154
        004106      145     054
    760 004110      116     151     145  M050:    .ASCII  !Niet-bestaande aandrijver!
        004113      164     055     142
        004116      145     163     164
        004121      141     141     156
        004124      144     145     040
        004127      141     141     156
        004132      144     162     151
        004135      152     166     145
        004140      162
    761 004141      117     156     147  M051:    .ASCII  !Ongeldig eenheidsnummer !
        004144      145     154     144
        004147      151     147     040
        004152      145     145     156
        004155      150     145     151
        004160      144     163     156
        004163      165     155     155
        004166      145     162     040
    762 004171      117     156     147  M052:    .ASCII  !Ongeldig apparaat!
        004174      145     154     144
        004177      151     147     040
```

H2

Nederlands LANGUAGE TEXT

```
     004202     141     160     160
     004205     141     162     141
     004210     141     164
 763 004212     106     157     165  M053:    .ASCII  !Fout in besturingsmodule!
     004215     164     040     151
     004220     156     040     142
     004223     145     163     164
     004226     165     162     151
     004231     156     147     163
     004234     155     157     144
     004237     165     154     145
 764 004242     106     157     165  M054:    .ASCII  !Fout in aandrijver!
     004245     164     040     151
     004250     156     040     141
     004253     141     156     144
     004256     162     151     152
     004261     166     145     162
 765 004264     015     015     102  M055:    .ASCII  <CR><CR>!Bezig met opstarten !
     004267     145     172     151
     004272     147     040     155
     004275     145     164     040
     004300     157     160     163
     004303     164     141     162
     004306     164     145     156
     004311     040
 766 004312     015     132     151  M056:    .ASCII  <CR>!Zie "Handleiding bij de apparatuur" voor informatie hier!
     004315     145     040     042
     004320     110     141     156
     004323     144     154     145
     004326     151     144     151
     004331     156     147     040
     004334     142     151     152
     004337     040     144     145
     004342     040     141     160
     004345     160     141     162
     004350     141     164     165
     004353     165     162     042
     004356     040     166     157
     004361     157     162     040
     004364     151     156     146
     004367     157     162     155
     004372     141     164     151
     004375     145     040     150
     004400     151     145     162
 767 004403     157     166     145           .ASCII  !over!<CR>
     004406     162     015
 768 004410     033     133     062  M057:    .ASCII  <ESC>/[2J/      ;Erase screen
     004413     112
 769 004414     033     133     065           .ASCII  <ESC>/[5;0H/    ;Set cursor to line 5 and col 1
     004417     073     060     110
 770 004422                          M060:
 771 004422     102     145     162  M061:    .ASCII  !Bericht !
     004425     151     143     150
     004430     164     040
 772 004432     015     015          M062:    .BYTE   CR,CR
 773 004434     015     015     113  M063:    .ASCII  <CR><CR>/KDJ11-B >/
     004437     104     112     061
```

```
        004442      061      055      102
        004445      040      076
    774 004447      015      105      105  M064:    .ASCII  <CR>!EEPROM opstartfout!<CR>
        004452      120      122      117
        004455      115      040      157
        004460      160      163      164
        004463      141      162      164
        004466      146      157      165
        004471      164      015
    775 004473      010      040      010  M065:    .BYTE   BACKSP,SPACE,BACKSP
    776 004476      015      117      156  M066:    .ASCII  <CR>!Ongeldige opdracht!<CR>
        004501      147      145      154
        004504      144      151      147
        004507      145      040      157
        004512      160      144      162
        004515      141      143      150
        004520      164      015
    777 004522      015      015      117  M067:    .ASCII  <CR><CR>!Opdrachten zijn HELP, START en TOON.!
        004525      160      144      162
        004530      141      143      150
        004533      164      145      156
        004536      040      172      151
        004541      152      156      040
        004544      110      105      114
        004547      120      054      040
        004552      123      124      101
        004555      122      124      040
        004560      145      156      040
        004563      124      117      117
        004566      116      056
    778 004570      101      144      162  M070:    .ASCII  !Adres !
        004573      145      163      040
    779 004576      040      075      040  M071:    .ASCII  / = /
    780 004601      107      145      145  M072:    .ASCII  !Geef apparaat en eenheid op en druk op RETURN: !
        004604      146      040      141
        004607      160      160      141
        004612      162      141      141
        004615      164      040      145
        004620      156      040      145
        004623      145      156      150
        004626      145      151      144
        004631      040      157      160
        004634      040      145      156
        004637      040      144      162
        004642      165      153      040
        004645      157      160      040
        004650      122      105      124
        004653      125      122      116
        004656      072      040
    781 004660      011      040      040  M073:    .ASCII  <TAB>!  !
    782 004663      015      102      145  M074:    .ASCII  <CR>!Bezig met automatisch opstarten!<CR>
        004666      172      151      147
        004671      040      155      145
        004674      164      040      141
        004677      165      164      157
        004702      155      141      164
        004705      151      163      143
```

```
004710     150     040     157
004713     160     163     164
004716     141     162     164
004721     145     156     015
004724                             MEND1:
                                   .SBTTL  NULL DICTIONARY BLOCK, CHECKSUM AND LANGUAGE HEADER
004724                             wb:
004724     001                     ENGWRD: .BYTE   ENDBLK-ENGWRD
004725                             ENDBLK:


004725                             WEND:

004725     000                     CKSUM:  .byte   0        ;checksum


004726                             MEND:                            ;END OF NULL TEXT

004726                             ME:
004726                             WE:

                                   ;FOREIGN LANGUAGE HEADER

        000002                     B1      =       WE-WB&377                   ;DICTIONARY BYTE COUNT 7:0
        000000                     B2      =       WE-WB&17400/256.            ;DICTIONARY BYTE COUNT 10:8
        000220                     B3      =       MEND-text&377               ;TEXT BYTE COUNT 7:0
        000143                     B4      =       MEND-text&017400/256.!140   ;TEXT BYTE COUNT 12:8 & ID=011

004726     002                             .BYTE   B1
004727     000                             .BYTE   B2
004730     220                             .BYTE   B3
004731     143                             .BYTE   B4
004732     013                             .BYTE   -<B1+B2+B3+B4>&377          ;THIS BYTE IS HEADER CHECKSUM

004733                             FLEND:
004733                             BUFF:           ;TEMPORARY SAVE AREA FOR OLD AREA
        001000                             .END    START
```

K2

```
Symbol table

BACKSP= 000010          FLEND   004733          M010    003244          M042    003712          M074    004663
BCSR  = 177520          FMSG1   002430          M011    003246          M043    003726          NARGS = 000001
BDR   = 177524          FMSG1A  002465          M012    003246          M044    003754          NTYPE = 000027
BIT6  = 000100          FMSG1B  002500          M013    003246          M045    004002          OLDSIZ  002424
BIT7  = 000200          FMSG1C  002531          M014    003246          M046    004026          PCR   = 177522
BUFF    004733          FMSG1D  002554          M015    003246          M047    004050          PCRLB = 177522
B1    = 000002          FMSG2   002563          M016    003246          M050    004110          QUIT    002004
B2    = 000000          FMSG3   002644          M017    003246          M051    004141          QUIT1   002006
B3    = 000220          FMSG4   002737          M020    003246          M052    004171          REAROM  002314
B4    = 000143          LANG    001262          M021    003305          M053    004212          RETRY = 000002
CKSUM   004725          LF    = 000012          M022    003335          M054    004242          RMVTST= 173002
CR    = 000015          LNGHDR= 000140          M023    003373          M055    004264          ROMADR  002342
CRLF    002560          MAXERR= 000004          M024    003505          M056    004312          ROMSZ = 001625
DELAY = 025370          ME      004726          M025    003517          M057    004410          SPACE = 000040
DUMMY1  002526          MEND    004726          M026    003520          M060    004422          START   001000
DUMMY2  002551          MEND1   004724          M027    003551          M061    004422          TAB   = 000011
ENDBLK  004725          MOVROM  002134          M030    003556          M062    004432          TEXT    003106
ENDE2R= 166000          MSG000  003006          M031    003565          M063    004434          UFDHDR= 000040
ENGWRD  004724          MSG001  003046          M032    003605          M064    004447          UFDSIZ  002426
ESC   = 000033          M001    003203          M033    003607          M065    004473          WB      004724
EXIT    001566          M002    003216          M034    003654          M066    004476          WE      004726
EXIT1   001642          M003    003220          M035    003654          M067    004522          WEND    004725
E2LLB = 165006          M004    003225          M036    003655          M070    004570          WERR    002422
E2PAR = 165316          M005    003233          M037    003655          M071    004576          WRBYTE  002162
E2PROM= 165000          M006    003240          M040    003657          M072    004601          WRLANG  001462
E2WRIT  001666          M007    003242          M041    003712          M073    004660

. ABS.  004733    000   (RW,I,GBL,ABS,OVR)
        000000    001   (RW,I,LCL,REL,CON)
Errors detected:  0

*** Assembler statistics


Work  file  reads: 0
Work  file writes: 0
Size of work file: 8553 Words  ( 34 Pages)
Size of core pool: 19402 Words  ( 74 Pages)
Operating  system: RSX-11M/PLUS (Under VAX/VMS)

Elapsed time: 00:00:23.99
OEEBA0.BIC,COEEBA0/CR/-SP=COEEBA0
```

```
SYMBOL CROSS REFERENCE                                   CREF   V02

SYMBOL   VALUE              REFERENCES

BACKSP   = 000010           #5-260      6-775      6-775
BCSR     = 177520           #5-239      6-315     *6-316     *6-473
BDR      = 177524           #5-254
BIT6     = 000100           #5-258      6-510      6-531
BIT7     = 000200           #5-257      6-452      6-458      6-461
BUFF       004733            6-370      6-409      6-413      6-441      6-518     #6-814
B1       = 000002           #6-802      6-807      6-811
B2       = 000000           #6-803      6-808      6-811
B3       = 000220           #6-804      6-809      6-811
B4       = 000143           #6-805      6-810      6-811
CKSUM      004725            6-428      6-497     *6-500     #6-792
CR       = 000015           #5-255      6-641      6-648      6-649      6-650      6-651      6-653      6-654      6-735
                             6-736      6-738      6-738      6-738      6-739      6-740      6-747      6-751      6-754
                             6-754      6-765      6-765      6-766      6-767      6-772      6-772      6-773      6-773
                             6-774      6-774      6-776      6-776      6-777      6-777      6-782      6-782
CRLF       002560            6-471      6-471     #6-648
DELAY    = 025370           #5-247      6-562
DUMMY1     002526            6-575      6-575     #6-642
DUMMY2     002551            6-579      6-579     #6-645
ENDBLK     004725            6-786     #6-787
ENDE2R   = 166000           #5-245      6-349      6-503      6-523      6-596
ENGWRD     004724           #6-786      6-786
ESC      = 000033           #5-262      6-768      6-769
EXIT       001566           #6-455      6-532      6-534
EXIT1      001642            6-460      6-465     #6-471      6-514      6-529
E2LLB    = 165006           #5-244      6-456
E2PAR    = 165316           #5-243      6-466
E2PROM   = 165000           #5-242      5-243      5-244      5-245      6-324      6-333      6-335      6-505      6-525
                             6-598      6-618      6-623
E2WRIT     001666            6-443      6-449     #6-477
FLEND      004733            5-264     #6-813
FMSG1      002430            6-573      6-573     #6-638
FMSG1A     002465           *6-570     #6-639
FMSG1B     002500            6-572      6-572     #6-640
FMSG1C     002531            6-576      6-576     #6-643
FMSG1D     002554            6-580      6-580     #6-646
FMSG2      002563            6-321      6-321     #6-649
FMSG3      002644            6-344      6-344     #6-650
FMSG4      002737            6-337      6-337     #6-651
LANG       001262            6-368     #6-380
LF       = 000012           #5-256      6-641      6-648      6-649      6-650      6-651
LNGHDR   = 000140           #5-248      6-380
MAXERR   = 000004           #5-252      6-481
ME         004726           #6-797
MEND       004726           #6-795      6-804      6-805
MEND1      004724            6-487      6-719     #6-783
MOVROM     002134            6-371      6-410      6-417     #6-547
MSG000     003006            6-512      6-512     #6-653
MSG001     003046            6-533      6-533     #6-654
M001       003203            6-484      6-659      6-660     #6-720
M002       003216            6-660      6-661     #6-721
M003       003220            6-661      6-662     #6-722
```

M2

```
SYMBOL CROSS REFERENCE                                      CREF   V02

SYMBOL  VALUE           REFERENCES

M004     003225          6-662      6-663      #6-723
M005     003233          6-663      6-664      #6-724
M006     003240          6-664      6-665      #6-725
M007     003242          6-665      6-666      #6-726
M010     003244          6-666      6-667      #6-727
M011     003246          6-667      6-668      #6-728
M012     003246          6-668      6-669      #6-729
M013     003246          6-669      6-670      #6-730
M014     003246          6-670      6-671      #6-731
M015     003246          6-671      6-672      #6-732
M016     003246          6-672      6-673      #6-733
M017     003246          6-673      6-674      #6-734
M020     003246          6-674      6-675      #6-735
M021     003305          6-675      6-676      #6-736
M022     003335          6-676      6-677      #6-737
M023     003373          6-677      6-678      #6-738
M024     003505          6-678      6-679      #6-740
M025     003517          6-679      6-680      #6-741
M026     003520          6-680      6-681      #6-742
M027     003551          6-681      6-682      #6-743
M030     003556          6-682      6-683      #6-744
M031     003565          6-683      6-684      #6-745
M032     003605          6-684      6-685      #6-746
M033     003607          6-685      6-686      #6-747
M034     003654          6-686      6-687      #6-748
M035     003654          6-687      6-688      #6-749
M036     003655          6-688      6-689      #6-750
M037     003655          6-689      6-690      #6-751
M040     003657          6-690      6-691      #6-752
M041     003712          6-691      6-692      #6-753
M042     003712          6-692      6-693      #6-754
M043     003726          6-693      6-694      #6-755
M044     003754          6-694      6-695      #6-756
M045     004002          6-695      6-696      #6-757
M046     004026          6-696      6-697      #6-758
M047     004050          6-697      6-698      #6-759
M050     004110          6-698      6-699      #6-760
M051     004141          6-699      6-700      #6-761
M052     004171          6-700      6-701      #6-762
M053     004212          6-701      6-702      #6-763
M054     004242          6-702      6-703      #6-764
M055     004264          6-703      6-704      #6-765
M056     004312          6-704      6-705      #6-766
M057     004410          6-705      6-706      #6-768
M060     004422          6-706      6-707      #6-770
M061     004422          6-707      6-708      #6-771
M062     004432          6-708      6-709      #6-772
M063     004434          6-709      6-710      #6-773
M064     004447          6-710      6-711      #6-774
M065     004473          6-711      6-712      #6-775
M066     004476          6-712      6-713      #6-776
M067     004522          6-713      6-714      #6-777
```

```
                                                            N2
COEEBAO    CREATED BY  MACRO  ON 16-FEB-85 AT 13:54      PAGE 3

                                                                                          SEQ 0026
SYMBOL CROSS REFERENCE                                   CREF   V02

SYMBOL   VALUE          REFERENCES

M070       004570        6-714     6-715    #6-778
M071       004576        6-715     6-716    #6-779
M072       004601        6-716     6-717    #6-780
M073       004660        6-717     6-718    #6-781
M074       004663        6-718     6-719    #6-782
NARGS    = 000001       #6-321     6-321    #6-337     6-337    #6-344     6-344    #6-471     6-471    #6-512
                         6-512    #6-533     6-533    #6-572     6-572     6-572    #6-573     6-573    #6-575
                         6-575     6-575    #6-576     6-576    #6-579     6-579     6-579    #6-580     6-580
NTYPE    = 000027       #6-321     6-321    #6-337     6-337    #6-344     6-344    #6-471     6-471    #6-512
                         6-512    #6-533     6-533    #6-572     6-572    #6-573     6-573    #6-575     6-575
                        #6-575     6-575    #6-576     6-576    #6-579     6-579    #6-580     6-580
OLDSIZ     002424       *6-339    *6-376    *6-382     6-408     6-414    *6-416    *6-419     6-513     6-528
                        #6-631
PCR      = 177522       #5-240    *6-314    *6-340    *6-474    *6-506    *6-526    *6-599
PCRLB    = 177522       #5-241    *6-455     6-567    *6-627
QUIT       002004        6-482     6-485     6-488     6-491     6-494    #6-509
QUIT1      002006        6-345    #6-510
REAROM     002314        6-386     6-387     6-388     6-390     6-392     6-551    #6-594
RETRY    = 000002       #5-250     6-560
RMVTST   = 173002       #5-246     6-341
ROMADR     002342        6-384     6-438     6-517     6-548    #6-609
ROMSZ    = 001625       #5-264     6-437     6-447
SPACE    = 000040       #5-261     6-751     6-775
START      001000       #6-314     6-815
TAB      = 000011       #5-259     6-738     6-739     6-749     6-781
TEXT       003106        5-264     6-424     6-446    #6-659     6-659     6-804     6-805
UFDHDR   = 000040       #5-249     6-367     6-399
UFDSIZ     002426       *6-377    *6-407    *6-412     6-416    *6-420     6-436     6-439     6-528    #6-633
WB         004724       #6-785     6-802     6-803
WE         004726       #6-798     6-802     6-803
WEND       004725       #6-790
WERR       002422       *6-479     6-481    #6-630
WRBYTE     002162        6-464     6-469     6-477     6-520    #6-557
WRLANG     001462        6-358     6-372     6-378     6-381     6-418    #6-424
```

```
MACRO CROSS REFERENCE                                   CREF   V02

MACRO NAME       REFERENCES

.FRCTY           #5-299      6-471      6-512      6-533
.ITOA            #5-278      6-572      6-575      6-579
.TYPMS           #5-267      6-321      6-337      6-344      6-573      6-576      6-580
```