```
0001  MUU00

                ;                                       RCSL:    43-GL 4
                ;                                       AUTHOR:  JBP
                ;                                       EDITED:  74.02.15

                ;               MUU00

                ; KEYWORD:      MUS, DRIVER, UTILITY, LISTING.
                ; ABSTRACT:     MUS SYSTEM, UTILITY PROCEDURES.
                ;               ASCII PAPER TAPE: RCSL 43-GL5.
                ;               REL. BINARY PAPER TAPE: RCSL 43-GL6.
```

```
↑ 0002  MUU00
                ;   ******   SYSTEM UTILITY PROCEDURES   ******

                .TITL MUU00
                .NREL
          000012 .RDX 10
          000001 .TXTM 1


          000164 .LOC NEXTOP-GOS
    00164 000000'       A34         ; NEXT OPERATION
          000167 .LOC WAITOP-GOS
    00167 000050'       A340        ; WAIT OPERATION
          000165 .LOC RETURN-GOS
    00165 000067'       A35         ; RETURN ANSWER
          000171 .LOC SETRES-GOS
    00171 000110'       A37         ; SET RESERVATION
          000172 .LOC SETCON-GOS
    00172 000120'       A38         ; SET CONVERSION
          000173 .LOC CONBYTE-GOS
    00173 000124'       A39         ; CONBYTE
          000174 .LOC GETBYTE-GOS
    00174 000130'       A40         ; GETBYTE
          000175 .LOC PUTBYTE-GOS
    00175 000140'       A41         ; PUTBYTE
          000176 .LOC MULTIPLY-GOS
    00176 000155'       A42         ; MULTIPLY
          000177 .LOC DIVIDE-GOS
    00177 000171'       A43         ; DIVIDE
          000232 .LOC BINDEC-GOS
    00232 000206'       BD00        ; BINDEC
          000233 .LOC DECBIN-GOS
    00233 000243'       DB00        ; DECBIN

                .NREL
```

```
↑ 0005  MUU00
                ; PROCEDURE NEXT OPERATION(MODE,COUNT,BUF);
                ; WAITS FOR AN EVENT ARRIVING AT THE EVENT QUEUE OF THE
                ; CALLING PROCESS. ANSWERS ARE UNTOUCHED. TRANSPUT MESSAGES
                ; WITH COUNT EQUAL TO ZERO ARE ANSWERED WITH STATUS ZERO.
                ;
                ;         CALL:                 RETURN:                 LINK
                ; AC0                           MODE=OP(0:13)             +0: CONTROL
                ; AC1                           COUNT                     +1: INPUT
                ; AC2     CUR                   CUR                       +2: OUTPUT
                ; AC3     LINK                  BUF

00000'055024 A34:   STA     3    SAVE,2     ; NEXT OPERATION:
00001'152400 A341:  SUB     2,2             ;   BUF:= 0;
00002'006006        WAITEVENT               ;   WAIT EVENT(OP,COUNT,BUF,
00003'000777        JMP         .-1         ;     +0: WAIT NEXT);
00004'024443        LDA     1   A347        ;
00005'045426        STA     1   ADDRESS,3   ;   ADDRESS.CUR:=NEXT OP;
00006'025004 A3411:LDA      1   SENDER,2    ;
00007'021430        LDA     0   RESERVER,3; 
00010'101004        MOV     0,0 SZR         ;   IF RESERVER.CUR<>0 AND
00011'106405        SUB     0,1 SNR         ;    SENDER.BUF<>RESERVER.CUR THEN
00012'000402        JMP         .+2         ;   GOTO REJECT IT;
00013'000426        JMP         A345        ;
00014'021006        LDA     0   MESS0,2     ;
00015'101223        MOVZR   0,0 SNC         ;   IF OP(15:15)=0 THEN
00016'000427        JMP         A346        ;     GOTO SET CONTROL;
00017'025425        LDA     1   BUF,3       ;
00020'125405        INC     1,1 SNR         ;   IF BUF.CUR=-1 THEN
00021'000417        JMP         A344        ;    GOTO RETURN IT;
00022'025007        LDA     1   MESS1,2     ;
00023'125005        MOV     1,1 SNR         ;   IF COUNT=0 THEN
00024'000414        JMP         A344        ;     GOTO RETURN IT;
00025'011424        ISZ         SAVE,3      ;   RETURN:= INPUT;
00026'101222        MOVZR   0,0 SZC         ;   IF OP(14:14)=1 THEN
00027'011424        ISZ         SAVE,3      ;     RETURN:= OUTPUT;
00030'155000 A342:  MOV     2,3             ;
00031'030040        LDA     2   CUR         ;
00032'055025        STA     3   BUF,2       ;   BUF.CUR:=BUF;
00033'025410        LDA     1   MESS2,3     ;
00034'045026        STA     1   ADDRESS,2   ;   ADDRESS.CUR:=MESS2.BUF;
00035'025407        LDA     1   MESS1,3     ;
00036'045027        STA     1   COUNT,2     ;   COUNT.CUR:=MESS1.BUF;
00037'003024        JMP@        SAVE,2      ;   RETURN;
                A344:                       ; RETURN IT:
00040'102401        SUB     0,0 SKP         ;   STATUS:= 0 OR
00041'020107 A345:  LDA     0   ILLEGAL     ;   STATUS:=ILLEGAL;
00042'126400        SUB     1,1             ;
00043'006007        SENDANSWER              ;   SEND ANSWER(STATUS,0,BUF);
00044'003426        JMP@        ADDRESS,3   ;   GOTO ADDRESS.CUR;
                A346:                       ; SET CONTROL:
00045'101220        MOVZR   0,0             ;   RETURN:= CONTROL;
00046'000762        JMP         A342        ;   RETURN;

00047'000001'A347:  A341                    ; NEXT EVENT
```

```
↑ 0004  MUU00
                ; PROCEDURE WAIT OPERATION(TIMER,DEVICE,BUF.CUR,
                ;                          BUF, MODE, COUNT);
                ; WAITS FOR AN INTERRUPT OR TIMER OR EVENT AFTER BUF.CUR.
                ; TRANSPUT MESSAGES ARE HANDLED AS FOR NEXTOPERATION, WITH THE
                ; EXCEPTION THAT THE PROCEDURE RETURNS AFTER A SENDANSWER.
                ;       CALL:              RETURN:          LINK:
                ; AC0     TIMER              TIMER(MODE(0:13))
                ; AC1     DEVICE             DEVICE(COUNT)
                ; AC2     CUR                CUR(CUR)
                ; AC3     LINK                 CUR(BUF)
                ;
                ; RETURNS:
                ; +0:     TIMER EXPIRED
                ; +1:     DEVICE INTERRUPT
                ; +2:     EMPTY MESSAGE RETURNED
                ; +3:     CONTROL
                ; +4:     INPUT
                ; +5:     OUTPUT
                ;
00050'175400 A340:  INC     3,3             ; WAIT OPERATION:
00051'055024        STA     3   SAVE,2      ;
00052'031025        LDA     2   BUF,2       ;   BUF:=BUF.CUR;
00053'151415        INC#    2,2 SNR         ;   IF BUF.CUR=-1 THEN
00054'152400        SUB     2,2             ;    BUF:=0;
00055'006002        WAIT                    ;   WAIT(TIMER,DEVICE,BUF,MODE,COUNT);
00056'015424 A3401:DSZ          SAVE,3      ; +0: TIMER;
00057'030040        LDA     2   CUR         ; +1: INTERRUPT;
00060'003424        JMP@        SAVE,3      ; +2: ANSWER(IRR);
00061'011424        ISZ         SAVE,3      ; +3: MESSAGE:
00062'011424        ISZ         SAVE,3      ;
00063'024403        LDA     1   A3404       ;
00064'045426        STA     1   ADDRESS,3   ;   ADDRESS.CUR:=EMPTY RETURN;
00065'000721        JMP         A3411       ;   GOTO TEST MESSAGE;

00066'000056'A3404:A3401                    ;
```

```
↑ 0005  MUU00
                ; PROCEDURE RETURN ANSWER(STATUS);
                ; CALCULATES THE NUMBER OF TRANSFERRED BYTES AND RETURNS
                ; THE BUFFER. IF STATUS CONTAINS ONE OR MORE CLEAN BITS,
                ; BUF OF CURRENT IS SET TO -1.
                ;         CALL:                 RETURN:
                ; AC0     STATUS                STATUS
                ; AC1     SPECIAL ANSWER        DESTROYED
                ; AC2                           CUR
                   ; AC3    LINK                   DESTROYED

00067'030040 A35:  LDA     2   CUR         ; RETURN ANSWER:
00070'055024       STA     3   SAVE,2      ;
00071'031025       LDA     2   BUF,2       ;   BUF:= BUF.CUR;
00072'035010       LDA     3   MESS2,2     ;   BYTES:= ADDR.CUR;
00073'045010       STA     1   MESS2,2     ;   MESS2.BUF:=SPECIAL;
00074'030040       LDA     2   CUR         ;
00075'025026       LDA     1   ADDRESS,2   ;   BYTES:=BYTES-ADDRESS.CUR;
00076'031025       LDA     2   BUF,2       ;
00077'166400       SUB     3,1             ;
00100'006007       SENDANSWER              ;   SEND ANSWER(STATUS,BYTES,BUF);
00101'171000       MOV     3,2             ;
00102'024405       LDA     1   A350        ;   MASK:= CLEAN BITS;
00103'107404       AND     0,1 SZR         ;   IF (STATUS AND MASK)<>0 THEN
00104'126000       ADC     1,1             ;    BUF.CUR:=-1
00105'045025       STA     1   BUF,2       ;   ELSE BUF.CUR:=0;
00106'003024       JMP@        SAVE,2      ;     RETURN;

00107'161762 A350: 7B2+7B8+7B11+1B14       ; CLEAN BITS: 0-2, 6-11, 14

                ; PROCEDURE SET RESERVATION(MODE);
                ;         CALL:                 RETURN:
                ; AC0     MODE(0:13)            MODE(0:12)
                ; AC1                           DESTROYED
                ; AC2     CUR                   CUR
                ; AC3     LINK                  DESTROYED

                A37:                            ; SET RESERVATION:
00110'031025       LDA     2   BUF,2       ;   BUF:= BUF.CUR;
00111'025007       LDA     1   MESS1,2     ;   SENDER:= MESS1.BUF;
00112'125004       MOV     1,1 SZR         ;   IF SENDER<>0 THEN
00113'025004       LDA     1   SENDER,2    ;     SENDER:= SENDER.BUF;
00114'030040       LDA     2   CUR         ;
00115'101222       MOVZR   0,0 SZC         ;   IF MODE(13:13)<>0 THEN
00116'045030       STA     1   RESERVER,2  ;     RESERVER.CUR:= SENDER;
00117'001400       JMP        +0,3         ;   RETURN;

                   ; PROCEDURE SET CONVERSION(MODE);
                ;         CALL:                 RETURN:
                ; AC0     MODE(0:12)            MODE(0:11)
                ; AC1                           DESTROYED
                ; AC2     CUR                   CUR
                ; AC3     LINK                  DESTROYED

                A38:                            ; SET CONVERSION
00120'025026       LDA     1   ADDRESS,2   ;   TABLE:=MESS2.BUF.CUR;
00121'101222       MOVZR   0,0 SZC         ;   IF MODE(12:12)=1 THEN
00122'045031       STA     1   CONVT,2     ;     CONV TABLE.CUR:= TABLE;
00123'001400       JMP        +0,3         ;   RETURN;
```

```
↑ 0006  MUU00
                ; PROCEDURE CONBYTE(BYTE);
                ;         CALL:                 RETURN:
                ; AC0     BYTE                  BYTE (CONVERTED)
                ; AC1                           DESTROYED
                ; AC2     CUR                   CUR
                ; AC3     LINK                  DESTROYED

00124'025031 A39:  LDA     1   CONVT,2    ; CONBYTE:
00125'125005       MOV     1,1 SNR        ;   IF CONV TABLE.CUR<>0 THEN
00126'001400       JMP        +0,3        ;     GETBYTE(BYTE+CONV TABLE.CUR,BYTE)
00127'107000       ADD     0,1            ;   RETURN;

                ; PROCEDURE GETBYTE(ADDR,BYTE);
                ;         CALL:                 RETURN:
                ; AC0                           BYTE
                ; AC1     ADDR                  ADDR
                ; AC2                           CUR
                ; AC3     LINK                  DESTROYED

00130'131220 A40:  MOVZR   1,2            ; GETBYTE:
00131'021000       LDA     0  +0,2        ;   VALUE:= 0.(ADDR//2);
00132'101003       MOV     0,0 SNC        ;   IF ADDR(15:15)=0 THEN
00133'101300       MOVS    0,0            ;     BYTE:= VALUE(0:7)
00134'030143       LDA     2  .255        ;   ELSE
00135'143400       AND     2,0            ;     BYTE:= VALUE(8:15);
00136'030040 A400: LDA     2   CUR        ;
00137'001400       JMP        +0,3        ;   RETURN;

                ; PROCEDURE PUTBYTE(ADDR,BYTE);
                ; BYTE MUST BE IN THE RANGE 0 TO 255.
                ;         CALL:                 RETURN:
                ; AC0     BYTE                  BYTE
                ; AC1     ADDR                  ADDR
                ; AC2                           CUR
                ; AC3     LINK                  DESTROYED

00140'131220 A41:  MOVZR   1,2            ; PUTBYTE:
00141'025000       LDA     1  +0,2        ;   VALUE:= 0.(ADDR//2);
00142'055000       STA     3  +0,2        ;   SAVE(LINK);
00143'034147       LDA     3  .M256       ;
00144'175003       MOV     3,3 SNC        ;   IF ADDR(15:15)=0 THEN
00145'125300       MOVS    1,1            ;     VALUE(0:7):= BYTE
00146'167400       AND     3,1            ;   ELSE
00147'107003       ADD     0,1 SNC        ;     VALUE(8:15):= BYTE;
00150'125300       MOVS    1,1            ;
00151'035000       LDA     3  +0,2        ;   RESTORE(LINK);
00152'045000       STA     1  +0,2        ;   0.(ADDR//2):= VALUE;
00153'145100       MOVL    2,1            ;   COMMENT: CARRY CONTAINS ADDR(15:15);
00154'000762       JMP        A400        ;   RETURN;
```

```
↑ 0007  MUU00
                ; PROCEDURE MULTIPLY(OP1,OP2,RESULT);
                ; COMPUTES RESULT:= OP1*OP2.
                ;         CALL:                 RESULT:
                ; AC0     OP1                   RESULT(0:15)
                ; AC1     OP2                   RESULT(16:31)
                ; AC2                           CUR
                   ; AC3     LINK                  DESTROYED

00155'030040 A42:   LDA     2   CUR         ; MULTIPLY:
00156'055024        STA     3   SAVE,2      ;   SAVE(LINK);
00157'152400        SUB     2,2             ;   RESULT(0:15):= 0;
00160'034146        LDA     3  .M16         ;   STEPS:= 16;
00161'125203 A420:  MOVR    1,1 SNC         ;   COMMENT:
00162'151201        MOVR    2,2 SKP         ;       MULTIPLICATION
00163'113220        ADDZR   0,2             ;       AS SHOWN IN
00164'175404        INC     3,3 SZR         ;       HOW TO USE
00165'000774        JMP         A420        ;       THE NOVA
00166'125260        MOVCR   1,1             ;       COMPUTERS;
00167'141000        MOV     2,0             ;
00170'000414        JMP         A431        ;   RETURN;

                ; PROCEDURE DIVIDE(DIVIDEND,DIVISOR,QUOTIENT,REMAINDER);
                ;         CALL:                 RETURN:
                ; AC0     DIVIDEND              QUOTIENT
                ; AC1     DIVISOR               DIVISOR
                  ; AC2                           CUR
                ; AC3     LINK                  REMAINDER

00171'030040 A43:   LDA     2   CUR         ; DIVIDE:
00172'055024        STA     3   SAVE,2      ;   SAVE(LINK);
00173'176400        SUB     3,3             ;   HIGH PART:= 0;
00174'030146        LDA     2  .M16         ;   STEPS:= 16;
00175'101120        MOVZL   0,0             ;
00176'175100 A430:  MOVL    3,3             ;   COMMENT:
00177'136412        SUB#    1,3 SZC         ;       DIVISION
00200'136400        SUB     1,3             ;       AS SHOWN IN
00201'101100        MOVL    0,0             ;       HOW TO USE
00202'151404        INC     2,2 SZR         ;       THE NOVA
00203'000773        JMP         A430        ;       COMPUTERS;
00204'030040 A431:  LDA     2   CUR         ;
00205'003024        JMP@        SAVE,2      ; RETURN;
```

```
↑ 0008  MUU00
                ; PROCEDURE BINDEC(WORD,ADDR,CUR);
                ;         CALL:                 RETURN:
                ; AC0     WORD                  DESTROYED
                ; AC1     ADDR                  DESTROYED
                ; AC2     CUR                   CUR
                ; AC3     LINK                  DESTROYED
00206'055024 BD00: STA     3    SAVE,2     ; SAVE:=LINK;
00207'045027       STA     1    SAVE+3,2   ; COUNT:=ADDR;
00210'041025       STA     0    SAVE1,2    ; SAVE1:=WORD;
00211'034423       LDA     3    BD04       ;
00212'055026       STA     3    SAVE+2,2   ; SAVE+2:=ADDR.TENTABLE;
                                           ;  NEW:
00213'025025 BD01: LDA     1    SAVE1,2    ;   WORD:=SAVE1;
00214'037026       LDA@    3    SAVE+2,2   ;   TEN:=WORD(SAVE2);
00215'011026       ISZ             SAVE+2,2   ;   SAVE2:=SAVE2+1;
00216'161005       MOV     3,0  SNR        ;   IF TEN=0 THEN
00217'000407       JMP             BD03       ;   GOTO PUTBYTE;
00220'020135       LDA     0    .48        ;   BASE:= "0";
                                           ;  CREATE:
00221'166422 BD02: SUBZ    3,1  SZC        ;   WORD:=WORD-TEN;
                                           ;   IF TEN<WORD THEN
00222'101401       INC     0,0    SKP      ;   BASE:=BASE+1 ELSE
00223'167001       ADD     3,1    SKP      ;   BEGIN
                                           ;   WORD:=WORD+TEN;
00224'000775       JMP             BD02       ;   SAVE1:=WORD;
00225'045025       STA     1    SAVE1,2    ;   GOTO PUTBYTE;
                                           ;   END;
00226'025027 BD03: LDA     1    SAVE+3,2   ;  PUTBYTE:
00227'011027       ISZ             SAVE+3,2   ;   ADDR:=ADDR+1;
00230'006175       PUTBYTE                    ;    PUTBYTE(WORD,ADDR);
00231'101004       MOV     0,0  SZR        ;   IF WORD<>0 THEN
00232'000761       JMP             BD01       ;   GOTO NEW;
00233'003024       JMP@            SAVE,2     ; RETURN;

00234'000235'BD04: .+1                     ;
00235'023420       10000
00236'001750       1000
00237'000144       100
00240'000012       10
00241'000001       1
00242'000000       0
```

```
↑ 0009  MU000
                ; PROCEDURE DECBIN(ADDR,CUR,WORD);
                 ;          CALL:                    RETURN:
                ; AC0                               DESTROYED
                ; AC1     ADDR                      BINARY NUMBER
                ; AC2     CUR                       CUR
                ; AC3     LINK                      DESTROYED
  00243'055024 DB00: STA     3    SAVE,2      ; SAVE:=LINK;
  00244'045025       STA     1    SAVE1,2     ; SAVE1:=ADDR;
  00245'126400       SUB     1,1              ;
  00246'045026       STA     1    SAVE+2,2    ; WORD:=0;
  00247'025025 DB01: LDA     1    SAVE1,2     ;
  00250'011025       ISZ          SAVE1,2     ; ADDR:=ADDR+1;
  00251'006174       GETBYTE                  ;  GETBYTE(WORD,ADDR);
  00252'024135       LDA     1    .48         ;
  00253'034415       LDA     3    DB03        ;
  00254'116032       ADCZ#   0,3  SZC         ;    IF NUMBER<10 THEN
  00255'122423       SUBZ    1,0  SNC         ;     IF NUMBER>=0
  00256'000410       JMP          DB02        ;     BEGIN
  00257'025026       LDA     1    SAVE+2,2    ;
  00260'135120       MOVZL   1,3              ;
  00261'175120       MOVZL   3,3              ;
  00262'167120       ADDZL   3,1              ;
  00263'107000       ADD     0,1              ; WORD:=WORD*10+CHAR-48;
  00264'045026       STA     1    SAVE+2,2    ;    SAVE2:=WORD;
  00265'000762       JMP          DB01        ;   GOTO NEW;
  00266'025026 DB02: LDA     1    SAVE+2,2    ;
  00267'003024       JMP@         SAVE,2      ;

  00270'000072 DB03: 58                       ;

                ;   ******   END OF SYSTEM UTILITY PROCEDURES ******

                .END
```