```
000364                  STA         MIHBANK                 ;Set interrupt handler bank
000365                  LDA         #SIRCOUNT
000366                  LDX         SIRADDR
000367                  LDY         SIRADDR+1
000368                  JSR         ALLOCSIR                ;Allocate the ACIA
000369                  BCS         $020
000370
000371                  LDX         #DCB_LN-1               ;Copy Device Configuration Block
000372  ;  into device control parameters
000373
000374  $015            LDA         DCB,X
000375                  STA         CNTL_PARAM+1,X
000376                  DEX
000377                  BPL         $015
000378
000379                  LDA         #0
000380                  STA         IS_NEWLINE              ;Set newline mode to False
000381                  STA         NEWLINE                 ;Clear newline character
000382                  STA         RD_IMMEDIATE            ;Read immediate mode off
000383                  STA         IN_PROG                 ;XOFF in progress flag off
000384                  STA         SEND_XON                ;Send XON flag off
000385
000386                  JSR         CNTL00                  ;Set up ACIA
000387                  LDA         #TRUE                   ;  and clear STAT_REG, STAT_LATCH,
000388  ;  RTS_FALSE, NO_OUTPUT, CHAR_OUT,
000389  ;  IN_PROG1, DLYCNT and BRK_CNT
000390                  STA         OPENFLG                 ;Set serial port open
000391                  RTS
000392
000393  $020            LDA         #XNORESRC
000394                  JSR         SYSERR
000395                  .PAGE
000396  ;---------------------------------------------------------------------
000397  ;
000398  ;       RS232 Driver -- Close Request
000399  ;
000400  ;---------------------------------------------------------------------
000401
000402  RS_CLOSE        .EQU        *
000403                  ASL         OPENFLG                 ;Serial Port open?
000404                  BCS         $05                     ;  Yes
000405                  JMP         NOTOPEN
000406
000407  $05             LDA         OBUFCNT                 ;Wait for write completion
000408                  ORA         DLYCNT                  ; and delay complete
000409                  BNE         $05
000410
000411                  PHP                                 ;Save interrupt status
000412                  SEI                                 ;Disable interrupt system
000413                  SET_1MHZ
000414                  LDA         ACIACMD
000415                  AND         #0F0                    ;Disable Rcv/Xmit Interrupt
000416                  STA         ACIACMD                 ;DTR off, RTS off
000417                  LDA         ACIASTAT                ;Clear any prior interrupt
000418                  PLP                                 ;Restore interrupt status
000419
000420                  LDA         #SIRCOUNT
000421                  LDX         SIRADDR
000422                  LDY         SIRADDR+1
000423                  JSR         DEALCSIR                ;Deallocate the ACIA
000424                  RTS
000425
000426                  .PAGE
000427  ;---------------------------------------------------------------------
000428  ;
000429  ;       RS232 Driver -- Read Request
000430  ;
000431  ;---------------------------------------------------------------------
000432
000433  RS_READ         .EQU        *
000434                  BIT         OPENFLG                 ;Serial Port open?
000435                  BMI         $05
000436                  JMP         NOTOPEN
000437
000438  $05             LDA         ISCSMRPTR               ;Get CSMRPTR from driver storage
000439                  STA         ICSMRPTR                ;Put in temporary zero page
000440                  LDY         #00                     ;Prevent offset
000441                  STY         RETCNT                  ;Zero return count
000442                  STY         RETCNT+1
000443
000444                  LDA         #0FF                    ;One's complement count
000445                  EOR         REQCNT
000446                  STA         REQCNT
000447                  LDA         #0FF
000448                  EOR         REQCNT+1
000449                  STA         REQCNT+1
000450
000451  $010            INC         REQCNT                  ;Increment count
000452                  BNE         $015                    ;Is count zero ?
000453                  INC         REQCNT+1
000454                  BEQ         $099                    ;Yes, terminate
000455
000456  $015            LDA         IBUFCNT                 ;Is input buffer empty ?
```

```
000922                  JMP         RETURN                  ;RTS false, return
000923
000924  $002            LDA         STAT_REG                ;Check DSR and DCD status
000925                  AND         #60
000926                  BEQ         $003                    ;DSR and DCD true, continue
000927
000928                  LDA         DLYCNT                  ;DSR or DCD false, disable xmit int
000929                  BNE         $011                    ;  unless delay in progress
000930                  JMP         D_XMIT
000931
000932  $003            LDA         #BITON4                 ;Check xmit data reg empty
000933                  BIT         STAT_REG
000934                  BNE         $004                    ;Reg empty, continue
000935
000936                  JMP         E_XMIT                  ;Reg not empty, enable xmit interrupt
000937
000938  $004            BIT         IN_PROG                 ;XOFF need to be sent ?
000939                  BVC         $005                    ;No, continue
000940
000941                  LDA         #BITON7                 ;Yes, set XOFF in progress
000942                  STA         IN_PROG
000943                  LDA         CTLCHR1                 ;Send XOFF
000944                  JMP         $020
000945
000946  $005            BIT         SEND_XON                ;XON need to be sent ?
000947                  BPL         $010                    ;No, continue
000948
000949                  LDA         #0                      ;Yes, clear flags
000950                  STA         SEND_XON
000951                  STA         IN_PROG
000952                  LDA         CTLCHR2                 ;Send XON
000953                  JMP         $020
000954
000955  $010            LDA         DLYCNT                  ;Any transmit delay in progress ?
000956                  BEQ         $015                    ;No
000957
000958  $011            DEC         DLYCNT                  ;Yes, decrement count
000959                  JMP         E_XMIT
000960
000961  $015            LDX         OBUFCNT                 ;Is local output buffer count zero ?
000962                  BEQ         D_XMIT                  ;Yes, disable xmit interrupt and return
000963
000964                  BIT         NO_OUTPUT               ;Is output suspended ?
000965                  BMI         D_XMIT                  ;Yes, disable xmit interrupt and return
000966
000967                  BIT         PROTOCOL                ;Is, ENQ/ACK protocol mode set?
000968                  BVC         $018                    ;No, continue
000969
000970                  LDA         CHAR_OUT                ;Yes, check output char count
000971                  BNE         $016                    ;Count not yet exhausted, send char
000972
000973                  BIT         IN_PROG1                ;Check for ENQ in progress
000974                  BMI         D_XMIT                  ;Yes, disable xmit interrupt and return
000975
000976                  LDA         #BITON7                 ;No, set ENQ in progress
000977                  STA         IN_PROG1
000978                  LDA         CTLCHR1                 ;Send ENQ
000979                  JMP         $020
000980
000981  $016            DEC         CHAR_OUT                ;Decrement output char count
000982  $018            LDX         OCSMRPTR                ;No, get consumer pointer
000983                  LDA         OLOCBUF,X               ;Get character from buffer
000984                  DEC         OBUFCNT
000985                  INC         OCSMRPTR
000986
000987  $020            TAX
000988                  SET_1MHZ
000989                  STX         ACIADATA                ;Send character
000990
000991                  CPX         #ASC_CR                 ;Check for any delay
000992                  BNE         $022
000993                  LDA         CRDELAY
000994                  JMP         $024
000995
000996  $022            CPX         #ASC_LF
000997                  BNE         $023
000998                  LDA         LFDELAY
000999                  JMP         $024
001000
001001  $023            CPX         #ASC_FF
001002                  BNE         E_XMIT
001003                  LDA         FFDELAY
001004
001005  $024            STA         DLYCNT
001006
001007  E_XMIT          SET_1MHZ
001008                  LDA         ACIACMD                 ;Enable transmit interrupt
001009                  AND         #0F2
001010                  ORA         #05                     ;Set to [xxxx01x1]
001011                  STA         ACIACMD
001012
001013                  RTS                                 ;Return to user
001014
```

```
001015  D_XMIT          SET_1MHZ
001016                  LDA         ACIACMD                 ;Disable transmit interrupt
001017                  AND         #0F2
001018                  ORA         #09                     ;Set to [xxxx10x1]
001019                  STA         ACIACMD
001020  RETURN          RTS                                 ;Return to user
001021
001022  PRIME_OUT       .EQU        *                       ;Called by Read, Write and Control
001023  ;  request routines
001024
001025                  PHP                                 ;Save interrupt status
001026                  SEI                                 ;Disable interrupt system
001027                  BIT         RTS_FALSE
001028                  BMI         $010                    ;Return if RTS false
001029                  JSR         E_XMIT                  ;Enable transmit interrupt
001030                  SET_2MHZ
001031  $010            PLP                                 ;Restore interrupt status
001032
001033                  RTS                                 ;Return
001034
001035                  .END
001036

; ##############################################################################################################
; #   END OF FILE:  RS232.text
; #   LINES      :  1036
; #   CHARACTERS :  51017
; #   Formatter  :  Assembly Language Reformatter 1.0.2 (07 January 1998)
; #   Author     :  David T. Craig -- 71533.606@compuserve.com -- Santa Fe, New Mexico USA
; ##############################################################################################################
```

###