RS-232C Interface with COP800

National Semiconductor
Application Note 739
Michelle Giles
May 1991

INTRODUCTION

This application note describes an implementation of the RS-232C interface with a COP888CG. The COP888CG 8-bit microcontroller features three 16-bit timer/counters, MICROWIRE/PLUS™ Serial I/O, multi-source vectored interrupt capability, two comparators, a full duplex UART, and two power saving modes (HALT and IDLE). The COP888CG feature set allows for efficient handling of RS-232C hardware handshaking and serial data transmission/reception.

SYSTEM OVERVIEW

In this application, a COP888CG is connected to a terminal using the standard RS-232C interface. The serial port of the terminal is attached to the COP888CG interface hardware using a standard ribbon cable with DB-25 connectors on either end. The terminal keyboard transmits ASCII characters via the cable to the COP888CG interface. All characters received by the COP888CG are echoed back to the terminal screen. If the COP888CG detects a parity or framing error, it transmits an error message back to the terminal screen.

HARDWARE DESCRIPTION

The COP888CG features used in this application include the user programmable UART, the 8-bit configurable L PORT, and vectored interrupts. In addition to the COP888CG, the RS-232C interface requires a DS14C88 driver and a DS14C89A receiver. The DS14C88 converts TTL/CMOS level signals to RS-232C defined levels and the DS14C89A does the opposite. *Figure 1* contains a diagram of the COP888CG interface hardware.

The COP888CG is configured as data communications equipment (DCE) and the terminal is assumed to be data terminal equipment (DTE). The following RS-232C signals are used to communicate between the COP888CG (DCE) and the terminal (DTE):

RS-232C Signal Name	Signal Origin
TxD (Transmit Data)	DTE
RxD (Receive Data)	DCE
CTS (Clear To Send)	DCE
RTS (Request To Send)	DTE
DSR (Data Set Ready)	DCE
DTR (Data Terminal Ready)	DTE
DCD (Data Carrier Detect)	DCE

Five general purpose I/O pins on the COP888CG L PORT are used for the control signals CTS, DSR, DCD, RTS and DTR. Two additional L PORT pins are used for TxD and RxD. These two general purpose pins are configured for their alternate functions, UART transmit (TDX) and UART receive (RDX). According to the RS-232C interface standard, DCE transmits data to DTE on RxD and receives data from DTE on TxD. Therefore, the UART transmit data pin (TDX) is used for the RS-232C receive data signal (RxD) and the UART receive data pin (RDX) is used for the RS-232C transmit data signal (TxD). In this example, all handshaking between DCE and DTE is performed in hardware.

The terminal is setup to interface with the COP888CG by selecting the 9600 baud, 7 bits/character, odd parity and one stop bit options. The local echo back of characters is disabled to allow the COP888CG to perform the echo back function. The terminal is also configured to use the hardware control signals (CTS, DSR, RTS, DTR) for handshaking.

SOFTWARE DESCRIPTION

The software for this application consists of an initialization routine, several interrupt routines, and a disable routine. These routines handle RS-232C handshaking, transmitting and receiving of characters, error checking, and echoing back of received characters. *Figures 2* thru *5* contain flowcharts of the routines. The complete code is given at the end of this application note.

The initialization routine configures the UART, initializes the transmit/receive data buffer, and enables the 8-bit L PORT handling of RS-232C control signals. In this particular example, the UART is configured to operate at 9600 BAUD in full duplex, asynchronous mode. The framing format is chosen to be: 7 bits/character, odd parity, and one stop bit. Different baud rates, modes of operation, and framing formats may be selected by setting the ENUCMD, ENUICMD, BAUDVAL and PSRVAL constants located at the beginning of the code to alternative values. (Refer to the COP888CG data sheet or COP888 Family User's Manual for details on configuring the UART.) Each RS-232C control signal is assigned to an L PORT pin. Pins L0, L2, L5 and L6 are configured as outputs for the DCD, TxD, CTS and DSR signals, respectively. Pins L3, L4 and L7 are configured as inputs for TxD, RTS and DTR, respectively. The transmit/receive data buffer is a circular buffer whose location and size is selected by setting the START and END constants located at the beginning of the program. The initialization routine sets up the buffer based on these constants.

The interrupt routines respond to transmit buffer empty, receive buffer full, and L PORT interrupts. A generic context switching routine is used for entering and exiting all interrupts. This routine saves the contents of the accumulator, the PSW register and the B pointer before vectoring to the appropriate interrupt routine. It also restores the contents of saved registers before a return from interrupt is executed.

The UART transmitter interrupt is called when the transmit buffer empty flag (TBMT) is set. This routine checks for active RTS and DTR control signals. If both signals are active and there is data to be transmitted, a byte of data is loaded into the UART transmit buffer. Otherwise, the UART transmitter is disabled.

The L PORT interrupts are used to indicate an active-low transition of RTS and/or DTR. When both signals are active (the remote receiver is ready to accept data), this routine enables the UART transmitter.

The UART receiver interrupt routine is called when the receive buffer full flag (RBFL) is set. This routine reads the

MICROWIRE/PLUS™ is a trademark of National Semiconductor Corporation.

 TL/DD11110 RRD-B30M75/Printed in U. S. A.

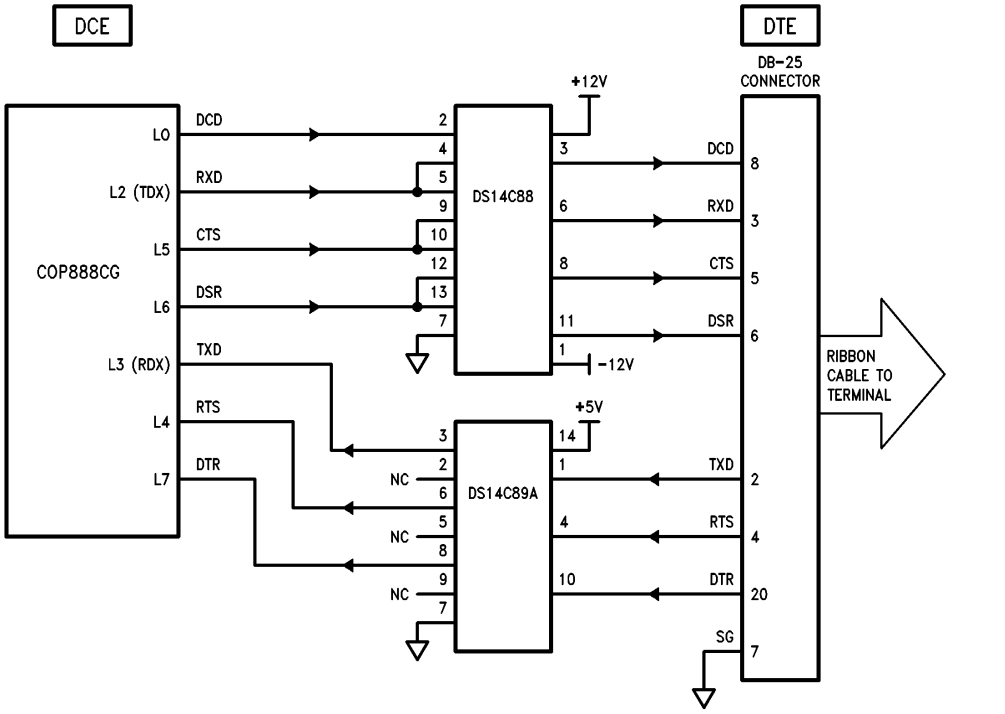
UART receive buffer and checks for errors. If no errors are detected, the incoming data is placed in the data buffer for echoing. If errors are detected, an error message is queued for transmission.

The receiver interrupt disables the remote transmitter by de-activating CTS whenever the transmit/receive data buffer is almost full. This action prevents the data buffer from overflowing. Note that CTS is turned off before the buffer is completely full to insure buffer space will exist for storing characters which are in the process of being sent when CTS is deactivated.

The disable routine clears the UART control registers, disables the L PORT interrupts, and de-activates the RS-232C control signals.

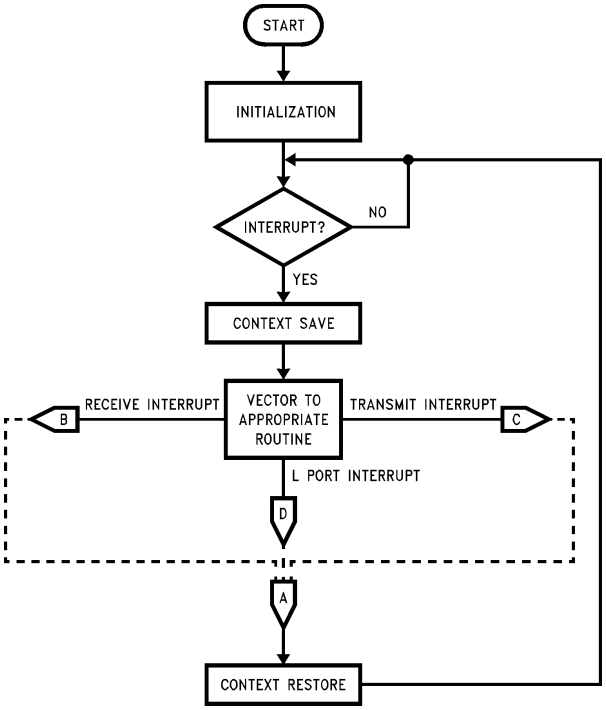
CONCLUSION

The user configurable UART, multiple external interrupt capabilities, and vectored interrupt scheme of the COP888CG microcontroller allow for an efficient implementation of the RS-232C interface standard. This application note shows how the COP888CG may be configured for connection to a terminal using these features. However, the code for this application can be easily adapted to other applications requiring different baud rates or framing formats, connection to a modem (DCE), separate transmit and receive buffers, incoming command decoding and/or handling of character strings. The versatility of the RS-232C standard and the COP888CG provides a means to develop practical solutions for many applications.



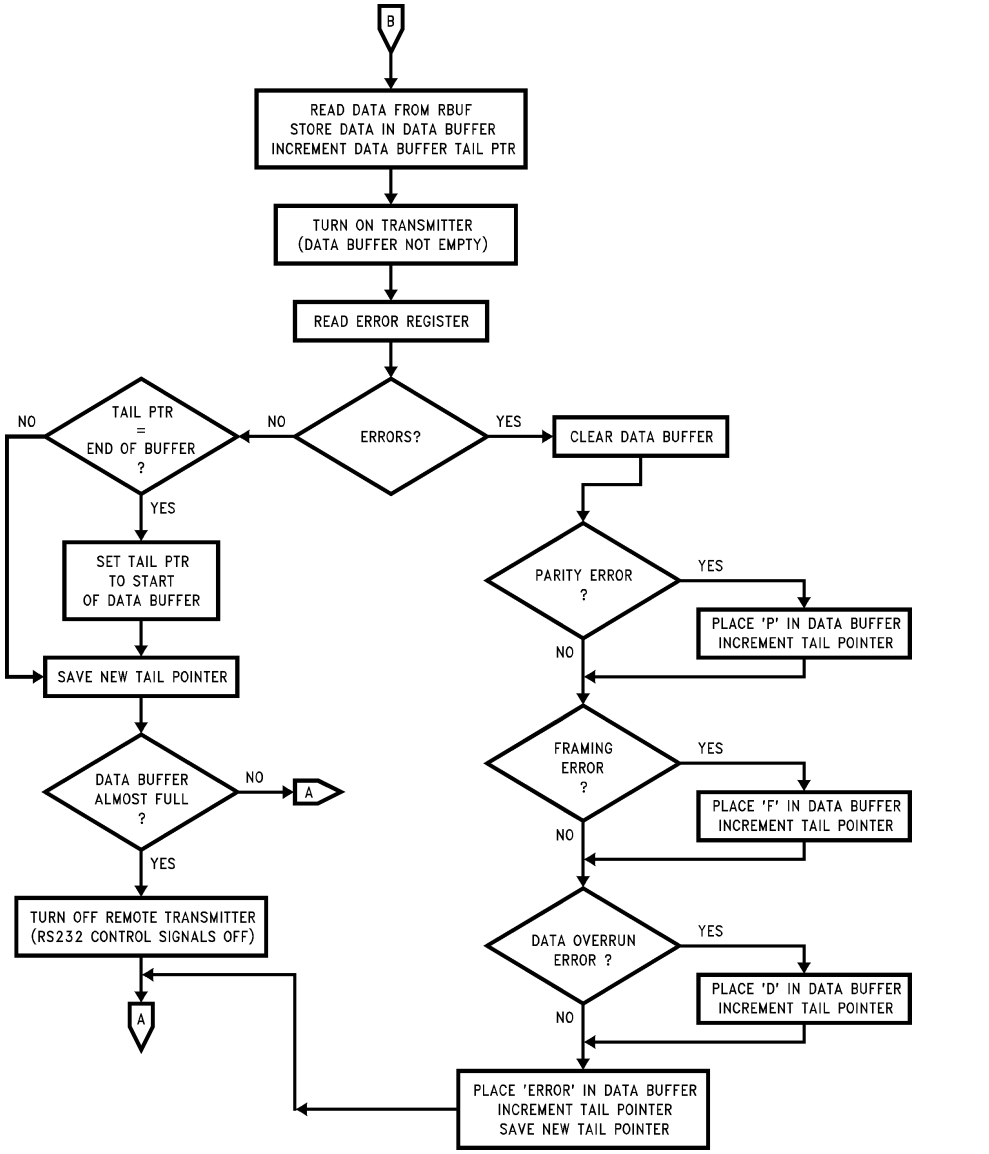
TL/DD/11110–1

FIGURE 1. Interface Diagram



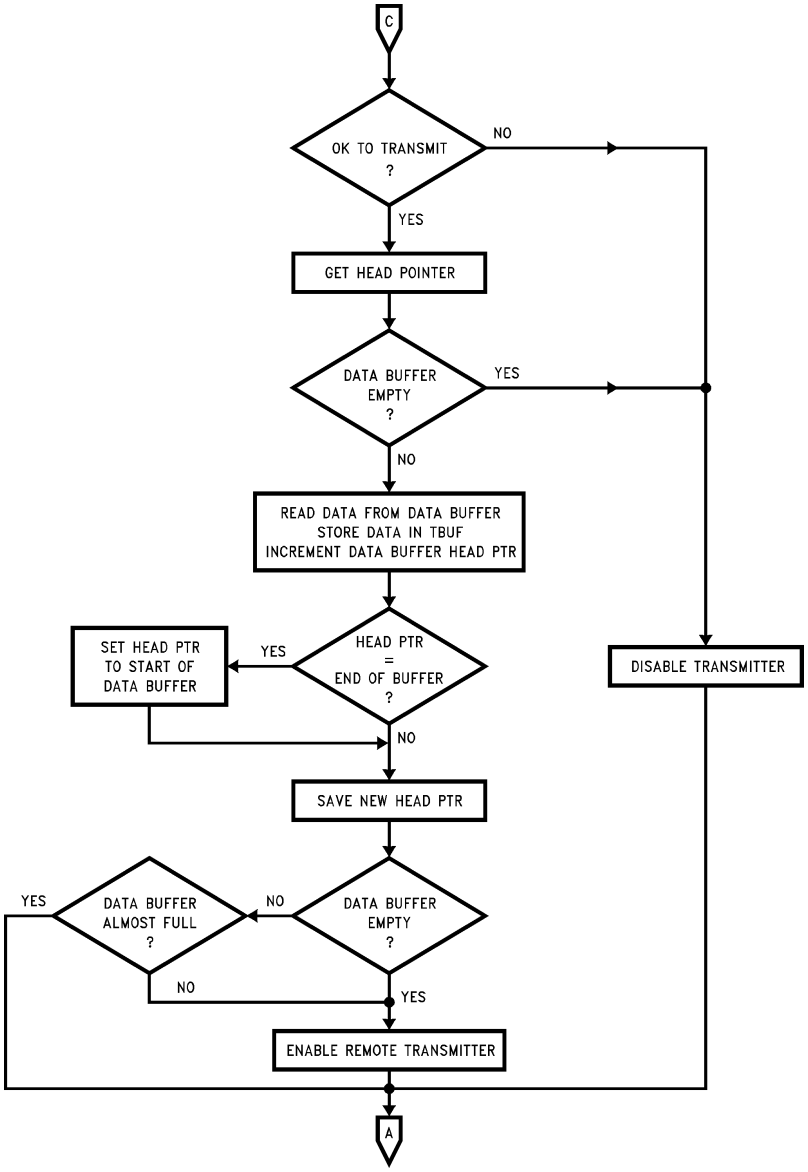
TL/DD/11110–2

FIGURE 2. Main Program Flow



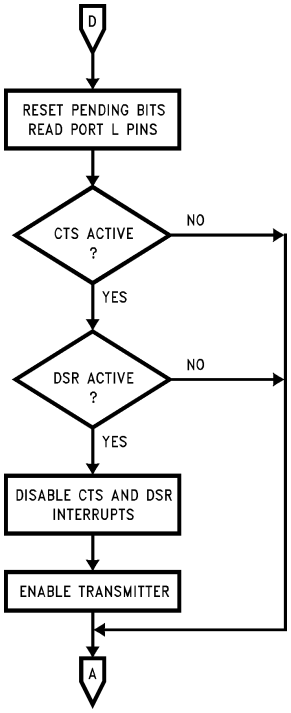
TL/DD/11110–3

FIGURE 3. Receiver Interrupt Routine



TL/DD/11110–4

FIGURE 4. Transmitter Interrupt Routine



TL/DD/11110–5

FIGURE 5. L Port Interrupt Routine

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER,REV:D1,12 OCT 88


  1                     ;The following set of routines uses the COP888CG UART and several I/O pins
  2                     ;to simulate an RS232 port interface. The code handles hardware control
  3                     ;signals, echo back of received characters, and error checking.  A single
  4                     ;routine called INIT initializes the UART and hardware control signals.
  5                     ;The transmitting and receiving of characters is handled in several
  6                     ;interrupt routines. The UART is disabled by calling the DISABLE routine.
  7                     ;The user must select values for several constants before compiling
  8                     ;this code.
  9                     ;
 10                     ;NOTES:
 11                     ;       * The COP tranmitter is enabled only when the transmit/receive
 12                     ;         buffer is not empty and the appropriate RS232 control signals
 13                     ;         from the remote receiver are present.
 14                     ;       * The COP receiver is always enabled.  the remote transmitter
 15                     ;       * The remote transmitter is disabled whenever the transmit/
 16                     ;         receive data buffer is full.
 17
 18                     ;Definition of Constants
 19     0089                    ENUCMD  =  089      ;Value to put in the ENU register
 20                                                 ;Selects bits per char and parity option
 21                                                 ;DEFAULT = 081  (7 bits/char and odd parity)
 22
 23     0020                    ENUICMD =  020      ;Value to put in the ENUI register
 24                                                 ;Selects number of stop bits, uart clock option,
 25                                                 ;sync/async option, xmit/rcv interrupt enable,
 26                                                 ;and TDX pin enable
 27                                                 ;DEFAULT = 023 ( 1 stop bit, internal BRG,
 28                                                 ;async operation, no interrupt, and TDX enabled)
 29
 30     0004                    BAUDVAL = 04        ;Baud rate divisor equals N - 1
 31     00C8                    PSRVAL  = 0C8       ;Baud rate prescalar
 32                                                 ; BR = FC/(16 * N*P) where
 33                                                 ;       FC = CKI frequency
 34                                                 ;       N  = Baud Divisor
 35                                                 ;       P  = Prescalar
 36                                                 ;GIVEN:         CALCULATE:   BAUDVAL:  PSRVAL:
 37                                                 ;CKI = 10MHz     N = 5
 38                                                 ;BR  = 9600      P = 13        04       0C8
 39                                                 ;
 40                                                 ;CKI = 10MHz     N = 10
 41                                                 ;BR  = 4800      P = 13        09       0C8
 42                                                 ;
 43                                                 ;See tables in users manual for translation
 44                                                 ;of N and P to BAUDVAL and PSRVAL
 45
 46     0010                    START   =  010      ;Beginning address of the xmit/rcv buffer
 47     001D                    END     =  01D      ;End address of the xmit/rcv buffer
 48     001E                    HEAD    =  01E      ;RAM address where current head of buffer stored
 49     001F                    TAIL    =  01F      ;RAM address where current tail of buffer stored
 50     000E                    SIZE    =  0E       ;Size of transmit/receive data buffer
 51
```

TL/DD/11110–6

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER, REV:D1, 12 OCT 88


  52      0000                  DCD     =  00       ;Bit position of DCD signal on L port pins
  53      0005                  CTS     =  05       ;Bit position of CTS signal on L port pins
  54      0007                  DTR     =  07       ;Bit position of DTR signal on L port pins
  55      0004                  RTS     =  04       ;Bit position of RTS signal on L port pins
  56      0006                  DSR     =  06       ;Bit position of DSR signal on L port pins
  57      0005                  ETDX    =  05       ;Bit position of TDX enable pin in ENUI
  58      0000                  TIE     =  00       ;Bit position of TX interrupt enable bit
  59      0001                  RIE     =  01       ;Bit position of RX interrupt enable bit
  60      0005                  PE      =  05       ;Bit position of parity error in ENUR
  61      0006                  FE      =  06       ;Bit position of framing error in ENUR
  62      0007                  DOE     =  07       ;Bit position of data overrun error in ENUR
  63
  64
  65
  66                            .INCLD  COP888.INC
  67
  68 0002 3008          MAIN:   JSR     INIT                ;INITIALIZE UART
  69 0004 FF                    JP      .                   ;DO OTHER TASKS
  70 0005 3044                  JSR     DISABLE             ;DISABLE UART
  71 0007 FF                    JP      .                   ;DO OTHER TASKS
  72
  73                    INIT:
  74 0008 9FEF                  LD      B,#PSW
  75 000A 68                    RBIT    GIE,[B]             ;DISABLE ALL INTERRUPTS
  76 000B BCBE00                LD      PSR,#00             ;UART OFF (POWERDOWN)
  77 000E BCD165                LD      PORTLC,#065         ;SET I/O
  78 0011 9FD0                  LD      B,#PORTLD           ;NOT READY TO RECEIVE
  79 0013 7E                    SBIT    DSR,[B]             ;   TURN OFF DATA SET READY
  80 0014 7D                    SBIT    CTS,[B]             ;   TURN OFF CLEAR TO SEND
  81 0015 68                    RBIT    DCD,[B]             ;   TURN ON DATA CARRIER DETECT
  82 0016 BC1E10                LD      HEAD,#START         ;INIT HEAD POINTER
  83 0019 BC1F10                LD      TAIL,#START         ;INIT TAIL POINTER
  84 001C 9FE8                  LD      B,#ICNTRL           ;CONFIGURE PORTL INTERRUPTS
  85 001E 6E                    RBIT    LPEN,[B]            ;  DISABLE PORTL INTERRUPTS
  86 001F BCC890                LD      WKEDG,#090          ;  SELECT FALLING EDGE FOR RTS AND DTR
  87 0022 BCC990                LD      WKEN,#090           ;  ENABLE RTS AND DTR INTERRUPT
  88 0025 BCCA00                LD      WKPND,#00           ;  CLEAR PORTL INTERRUPT PENDING FLAGS
  89 0028 7E                    SBIT    LPEN,[B]            ;  ENABLE PORT L INTERRUPTS
  90 0029 BCBA89                LD      ENU,#ENUCMD         ;SELECT BITS/CHAR AND PARITY OPTION
  91 002C BCBB00                LD      ENUR,#00            ;CLEAR ERROR BITS
  92 002F BCBC20                LD      ENUI,#ENUICMD       ;SELECT CLOCK, INTERRUPTS, STOPBITS
  93 0032 BCBD04                LD      BAUD,#BAUDVAL       ;SETUP BRG
  94 0035 9FBC                  LD      B,#ENUI
  95 0037 78                    SBIT    TIE,[B]             ;ENABLE TRANSMITTER INTERRUPT
  96 0038 79                    SBIT    RIE,[B]             ;ENABLE RECEIVER INTERRUPT
  97 0039 BCBEC8                LD      PSR,#PSRVAL         ;UART ON
  98 003C 9FD0                  LD      B,#PORTLD           ;READY TO RECEIVE
  99 003E 6E                    RBIT    DSR,[B]             ;  TURN ON DATA SET READY
 100 003F 6D                    RBIT    CTS,[B]             ;  TURN ON CLEAR TO SEND
 101 0040 9FEF                  LD      B,#PSW
 102 0042 78                    SBIT    GIE,[B]             ;ENABLE ALL INTERRUPTS
```

TL/DD/11110–7

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER,REV:D1,12 OCT 88


 103 0043 8E                    RET
 104
 105                    DISABLE:
 106 0044 BDEF68                RBIT    GIE,PSW             ;DISABLE INTERRUPTS
 107 0047 BCD061                LD      PORTLD,#061         ;TURN OFF HANDSHAKING SIGNALS
 108 004A BCBE00                LD      PSR,#00             ;UART POWERDOWN
 109 004D BCBA00                LD      ENU,#00             ;CLEAR UART CONTROL REGISTERS
 110 0050 BCBC00                LD      ENUI,#00
 111 0053 BCBB00                LD      ENUR,#00
 112 0056 9FC9                  LD      B,#WKEN             ;DISABLE RTS AND DTR INTERRUPTS
 113 0058 6C                    RBIT    RTS,[B]
 114 0059 6F                    RBIT    DTR,[B]
 115 005A BDEF78                SBIT    GIE,PSW             ;ENABLE INTERRUPTS
 116 005D 8E                    RET
 117
 118
 119                    ;INTERRUPT ROUTINES
 120
 121      00FF                  . = 0FF                     ;INTERRUPT START ADDRESS
 122 00FF 67                    PUSH    A                   ;CONTEXT SAVE
 123 0100 9DFE                  LD      A,B
 124 0102 67                    PUSH    A
 125 0103 9DEF                  LD      A,PSW
 126 0105 67                    PUSH    A
 127 0106 B4                    VIS
 128 0107 8C            REST:   POP     A                   ;CONTEXT RESTORE
 129 0108 9CEF                  X       A,PSW
 130 010A 8C                    POP     A
 131 010B 9CFE                  X       A,B
 132 010D 8C                    POP     A
 133 010E 8F                    RETI
 134
 135
 136                    ;PORT L INTERRUPTS
 137                    ;  The port L interrupts are used to indicate a return to active
 138                    ; state of the DTR and RTS signals from the remote receiver.
 139                    ; If both DTR and RTS are active, the remote receiver is ready
 140                    ; to accept data and the COP transmitter is enabled.
 141
 142                    LINT:                               ;PORT L INTERRUPT
 143 010F BCCA00                LD      WKPND,#00           ;RESET PENDING BITS
 144 0112 9DD2                  LD      A,PORTLP            ;READ PORT L PINS
 145 0114 6010                  IFBIT   #RTS,A              ;IF RTS (ACTIVE LOW) NOT PRESENT
 146 0116 06                    JP      NOTRDY              ;THEN REMOTE NOT READY TO RECEIVE
 147 0117 6080                  IFBIT   #DTR,A              ;IF DTR (ACTIVE LOW) NOT PRESENT
 148 0119 03                    JP      NOTRDY              ;THEN REMOTE NOT READY TO RECEIVE
 149 011A 9FBC          READY:  LD      B,#ENUI
 150 011C 78                    SBIT    TIE,[B]             ;RE-ENABLE TRANSMITTER INTERRUPT
 151 011D E9            NOTRDY: JP      REST                ;EXIT INTERRUPT
 152
 153
```

TL/DD/11110–8

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER,REV:D1,12 OCT 88


 154                    ;UART RECEIVE INTERRUPT
 155                    ;  The UART receive interrupt does the following:
 156                    ;       1. Reads the received data
 157                    ;       2. Checks for receiver errors
 158                    ;       3. If no errors detected, places the received data in
 159                    ;          the transmit/receive buffer and enables the transmitter.
 160                    ;       4. If errors detected, the transmit/receive buffer is cleared
 161                    ;          of ALL data and an error message is placed in the data buffer.
 162                    RCVINT:                                 ;RECEIVER INTERRUPT
 163 011E 9D1F                 LD      A,TAIL
 164 0120 9CFE                 X       A,B                     ;GET TAIL POINTER
 165 0122 9DB9                 LD      A,RBUF                  ;READ RECEIVED DATA
 166 0124 A2                   X       A,[B+]                  ;STORE RECEIVED DATA
 167 0125 9DBB                 LD      A,ENUR                  ;READ ERROR REGISTER
 168 0127 BDBC78               SBIT    TIE,ENUI                ;ENABLE TRANSMITTER INTERRUPT
 169 012A 60E0                 ANDSZ   A,#0E0                  ;CHECK FOR PE,DOE,FE
 170 012C 1A                   JP      ERROR                   ;THROW DATA AWAY IN BUFFER
 171 012D 9DFE                 LD      A,B                     ;LOAD ACC WITH NEW TAIL PTR
 172 012F 921E                 IFEQ    A,#END+1                ;IF END OF DATA BUFFER
 173 0131 9B10                 LD      A,#START                ; SET TAIL PTR TO START OF BUFFER
 174 0133 9C1F                 X       A,TAIL                  ;SAVE TAIL PTR
 175 0135 9D1E                 LD      A,HEAD                  ;IS DATA BUFFER FULL?
 176 0137 A1                   SC
 177 0138 BD1F81               SUBC    A,TAIL                  ;A = HEAD - TAIL
 178 013B 89                   IFNC                            ;IF BORROWED (TAIL ) HEAD)
 179 013C 940E                 ADD     A,#SIZE                 ;THEN ADD BUFFER SIZE TO RESULT
 180 013E 9303                 IFGT    A,#03                   ;IF DATA BUFFER NOT FULL
 181 0140 2107                 JMP     REST                    ;  THEN EXIT INTERRUPT
 182 0142 BDD07D        RXOFF:  SBIT    CTS,PORTLD              ;  ELSE TURN OFF REMOTE TRANSMITTER
 183 0145 2107                 JMP     REST                    ;EXIT INTERRUPT
 184                    ERROR:
 185 0147 BC1E10               LD      HEAD,#START             ;CLEAR BUFFER
 186 014A 9F10                 LD      B,#START                ;POINT TO START OF BUFFER
 187 014C 6020                 IFBIT   PE,A
 188 014E 9A50                 LD      [B+],#'P'               ;P = PARITY
 189 0150 6040                 IFBIT   FE,A
 190 0152 9A46                 LD      [B+],#'F'               ;F = FRAMING
 191 0154 6080                 IFBIT   DOE,A
 192 0156 9A44                 LD      [B+],#'D'               ;D = DATA OVERRUN
 193 0158 9A20                 LD      [B+],#020               ;BLANK SPACE
 194 015A 9A45                 LD      [B+],#'E'
 195 015C 9A52                 LD      [B+],#'R'
 196 015E 9A52                 LD      [B+],#'R'
 197 0160 9A4F                 LD      [B+],#'O'
 198 0162 9A52                 LD      [B+],#'R'
 199 0164 9A0A                 LD      [B+],#0A                ;LINE FEED
 200 0166 9A0D                 LD      [B+],#0D                ;CARRIAGE RETURN
 201 0168 9DFE          OUTERR: LD      A,B                     ;SAVE NEW TAIL PTR
 202 016A 9C1F                 X       A,TAIL
 203 016C 2107                 JMP     REST
 204
```

TL/DD/11110–9

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER,REV:D1,12 OCT 88


 205
 206                    ;UART TRANSMIT INTERRUPT
 207                    ;  The UART transmit interrupt does the following:
 208                    ;       1. Checks for RTS and DTR signals (OK to transmit?)
 209                    ;       3. If OK to transmit and buffer not empty, transmits data.
 210                    ;       4. If not OK to transmit or buffer empty, disables transmitter.
 211
 212                    XMITINT:                        ;TRANSMITTER INTERRUPT
 213 016E 9DD2                  LD      A,PORTLP
 214 0170 6090                  ANDSZ   A,#090          ;IS IT OK TO TRANSMITT?
 215 0172 219C                  JMP     IDLE            ;NO:  GO TURN OFF TRANSMITTER
 216 0174 9D1E                  LD      A,HEAD          ;YES: GET PTR TO DATA
 217 0176 BD1F82                IFEQ    A,TAIL          ;IF DATA BUFFER EMPTY
 218 0179 219C                  JMP     IDLE            ;THEN TURN OFF TRANSMITTER
 219 017B 9CFE                  X       A,B             ;ELSE
 220 017D AA                    LD      A,[B+]          ;GET TRANSMIT DATA
 221 017E 9CB8                  X       A,TBUF          ;SEND TRANSMIT DATA
 222 0180 9DFE                  LD      A,B             ;LOAD ACC WITH NEW HEAD PTR
 223 0182 921E                  IFEQ    A,#END+1        ;IF END OF DATA BUFFER
 224 0184 9810                  LD      A,#START        ;  SET HEAD PTR TO START OF BUFFER
 225 0186 9C1E                  X       A,HEAD          ;SAVE HEAD PTR
 226 0188 9D1E                  LD      A,HEAD          ;IS DATA BUFFER FULL?
 227 018A BD1F82                IFEQ    A,TAIL          ;IF BUFFER EMPTY
 228 018D 09                    JP      NFULL           ;   THEN NOT FULL
 229 018E A1                    SC                      ;   ELSE CHECK HOW FULL
 230 018F BD1F81                SUBC    A,TAIL          ;A = HEAD - TAIL
 231 0192 89                    IFNC                    ;IF BORROWED (TAIL > HEAD)
 232 0193 940E                  ADD     A,#SIZE         ;THEN ADD BUFFER SIZE TO RESULT
 233 0195 9303                  IFGT    A,#03           ;IF DATA BUFFER NOT FULL
 234 0197 BDD06D        NFULL:  RBIT    CTS,PORTLD      ;   THEN TURN ON REMOTE TRANSMITTER
 235 019A 2107                  JMP     REST            ;   ELSE EXIT INTERRUPT
 236 019C 9FBC          IDLE:   LD      B,#ENUI
 237 019E 68                    RBIT    TIE,[B]         ;DISABLE TRANSMITTER INTERRUPT
 238 019F 2107                  JMP     REST            ;EXIT INTERRUPT
 239
 240                    ;Software Trap
 241                    ;
 242 01A1 B5            SFTINT: RPND
 243 01A2 2000                  JMP     00              ;RESTART
 244
 245                    ;VECTOR INTERRUPT TABLE
 246
 247      01E2                  .=01E2
 248 01E2 010F                  .ADDRW  LINT            ;L PORT INTERRUPT
 249      01EC                  .=01EC
 250 01EC 016E                  .ADDRW  XMITINT         ;TRANSMITTER INTERRUPT
 251 01EE 011E                  .ADDRW  RCVINT          ;RECEIVER INTERRUPT
 252      01FE                  .=01FE
 253 01FE 01A1                  .ADDRW  SFTINT          ;SOFTWARE INTERRUPT/TRAP
 254                            .END
```

TL/DD/11110–10

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER, REV:D1, 12 OCT 88

SYMBOL TABLE


B       00FE          BAUD    00BD          BAUDVA 0004          CNTRL   00EE  *
CTS     0005          DCD     0000          DISABL 0044          DOE     0007
DSR     0006          DTR     0007          END    001D          ENU     00BA
ENUCMD  0089          ENUI    00BC          ENUICM 0020          ENUR    00BB
ERROR   0147          ETDX    0005  *       FE     0006          GIE     0000
HEAD    001E          ICNTRL  00E8          IDLE   019C          INIT    0008
LINT    010F          LPEN    0006          MAIN   0002  *       NFULL   0197
NOTRDY  011D          OUTERR  0168  *       PE     0005          PORTLC  00D1
PORTLD  00D0          PORTLP  00D2          PSR    00BE          PSRVAL  00C8
PSW     00EF          RBUF    00B9          RCVINT 011E          READY   011A  *
REST    0107          RIE     0001          RTS    0004          RXOFF   0142  *
SFTINT  01A1          SIZE    000E          SP     00FD          START   0010
TAIL    001F          TBUF    00B8          TIE    0000          WKEDG   00C8
WKEN    00C9          WKPND   00CA          X      00FC          XMITIN  016E
```

TL/DD/11110–11

```
NATIONAL SEMICONDUCTOR CORPORATION
COP800 CROSS ASSEMBLER, REV:D1, 12 OCT 88

MACRO TABLE


   NO WARNING LINES

   NO ERROR LINES

  267  ROM BYTES USED

SOURCE CHECKSUM = 6884
OBJECT CHECKSUM = 096B

INPUT   FILE C:UART.MAC
LISTING FILE C:UART.PRN
OBJECT  FILE C:UART.LM
```

TL/DD/11110–12

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

National Semiconductor Corporation
2900 Semiconductor Drive
P.O. Box 58090
Santa Clara, CA 95052-8090
Tel: 1(800) 272-9959
TWX: (910) 339-9240

National Semiconductor GmbH
Livry-Gargan-Str. 10
D-82256 Fürstenfeldbruck
Germany
Tel: (81-41) 35-0
Telex: 527649
Fax: (81-41) 35-1

National Semiconductor Japan Ltd.
Sumitomo Chemical
Engineering Center
Bldg. 7F
1-7-1, Nakase, Mihama-Ku
Chiba-City,
Ciba Prefecture 261
Tel: (043) 299-2300
Fax: (043) 299-2500

National Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: (852) 2737-1600
Fax: (852) 2736-9960

National Semiconductores Do Brazil Ltda.
Rue Deputado Lacorda Franco
120-3A
Sao Paulo-SP
Brazil 05418-000
Tel: (55-11) 212-5066
Telex: 391-1131931 NSBR BR
Fax: (55-11) 212-1181

National Semiconductor (Australia) Pty, Ltd.
Building 16
Business Park Drive
Monash Business Park
Nottinghill, Melbourne
Victoria 3168 Australia
Tel: (3) 558-9999
Fax: (3) 558-9998