8251 Programmable/Communication Interface

As an example of a serial interface device let us consider Intel's 8251 A programmable communication interface. The 8251A is diagrammed in Figure 4.7. It is capable Page **1** of **11**being programmed for asynchronous or synchronous communication. The data in buffer and data-out buffer registers share the same port address. For input, the serial bit stream arriving on the R x D pin is shifted into the receiver shift register and then the data bits are transferred to the data-in buffer register, where they can be input by the CPU. Conversely, on output the data bits put in the data-out buffer register by the CPU are transferred to the transmitter shift register and, along with the necessary synchronization bits, are shifted out through the T x D pin. Among other things the contents of the mode register, which are initialized by the executing program, determine whether the 8251A is in asynchronous mode or synchronous mode and the format of the characters being received and transmitted. The control register, which is also set by the program, controls the operation of the interface, and the status register makes certain information available to the executing program. Clearly, the sync character registers are for storing the sync characters needed for synchronous communication.

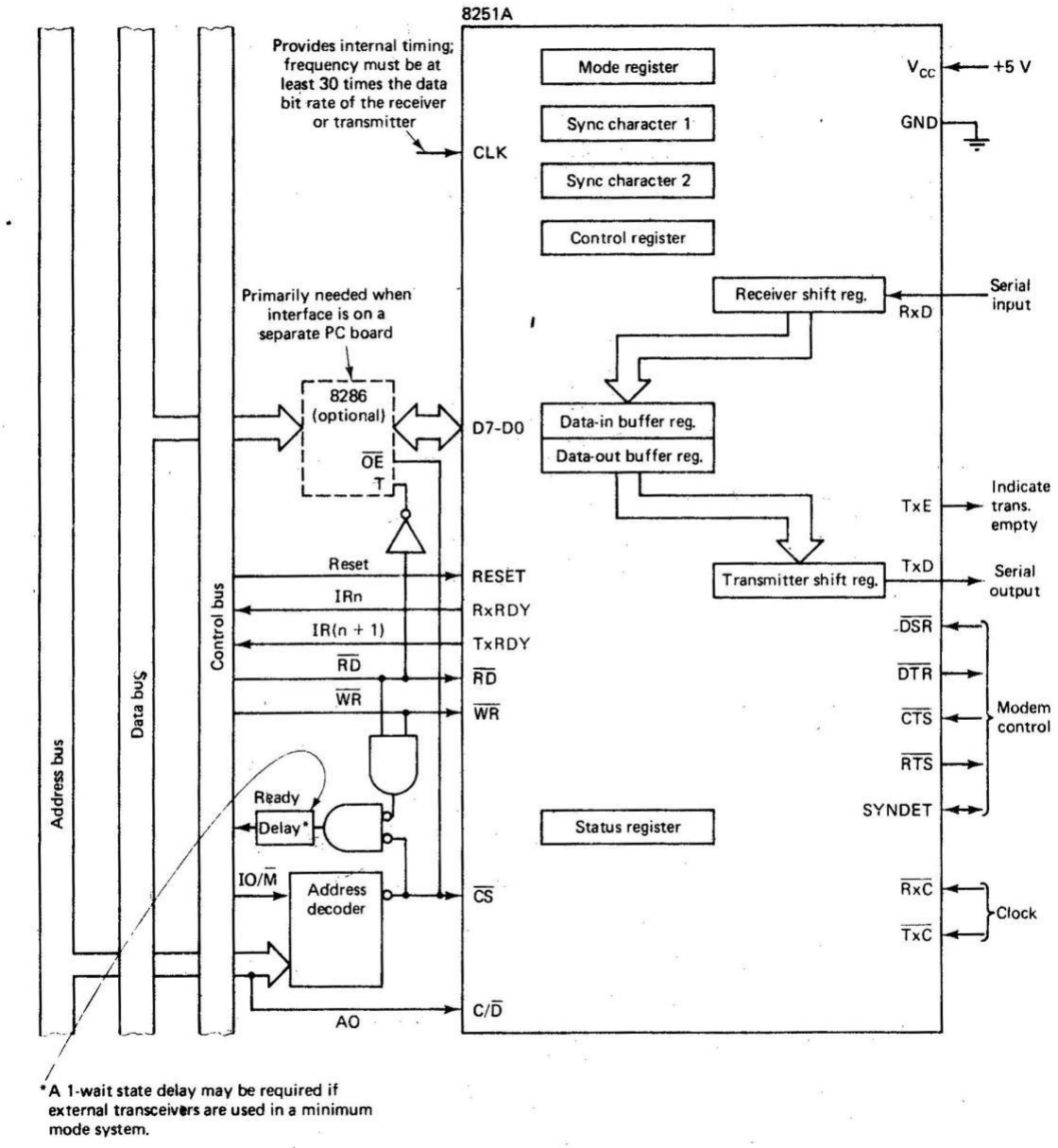


Figure 4.7: 8251 A Serial Communication Interface

Even though all seven of the registers on the left side of Figure 4.7 can be accessed by the processor, the 8251A is associated with only two port addresses. The C/D pin is connected to the address line AO and AO differentiates the two port addresses. The 8251A internally interprets the C/D, RD, and WR signals as follows:

C/D(=AO)	RD	WR	
0	0	1	Data input from the data-in buffer
0	1	0	Data output to the data-out buffer
1	0	1	Status register is put on data bus
1	1	0	Data bus is put in mode, control, or sync character register

where 1 means that the pin is high and 0 means that it is low. All other combinations cause the three-state D7-DO pins to go into their high-impedance state.

Whether the mode, control, or sync character register is selected depends on the accessing sequence. A flowchart of the sequencing is given in Figure 4.8. After a hardware reset or a command is given with its reset bit set to 1, the next output with A0=1 (i.e., with C/D=1, RD=1, and WR=0) is directed to the mode register. The formats of the mode register for both the asynchronous and synchronous cases are defined in Figure 4.9. If the two LSBs of the mode are zero, then the interface is put in its synchronous mode and the MSB determines the number of sync characters. In the synchronous mode, the next 1 or 2 bytes output with A0 = 1 become the sync characters. If the two LSBs of the mode are not both 0, then the 8251A enters its asynchronous mode. In either case, all subsequent bytes prior to another reset go to the control register if A0=1 and the data-out buffer register if A0 = 0.

In the synchronous mode the baud rates of the transmitter and receiver, which are the shift rates of the shift registers, are the same as the frequencies of the signals applied to TxC and RxC, respectively, but in the asynchronous mode the three remaining possible combinations for the two LSBs in the mode register dictate the baud rate factor. The relationship between the frequencies of the TxC and RxC clock inputs and the baud rates of the transmitter and receiver is:

Clock frequency = Baud rate factor x Baud rate

If 10 is in the LSBs of the mode register and the transmitter and receiver baud rates are to be 300 and 1200, respectively, then the frequency applied to TxC should be 4800 Hz and the frequency at RxC should be 19.2 kHz. In both the asynchronous and synchronous modes, bits 2 and 3 indicate the number of data bits in each character, bit 4 indicates whether or not there is to be parity bit, and bit 5 specifies the type of parity (odd or even). For the asynchronous mode the two MSBs indicate the number of stop bits, but for the synchronous mode bit 6 determines whether the SYNDET pin is to be used as an input or as an output and, as mentioned above, bit 7 indicates the number of sync characters.

If the SYNDET pin is used as an output it becomes active when a bit-for-bit match has been found between the incoming bit stream and the sync character(s). If the search for sync characters is conducted by an external device, then SYNDET can be used to input a signal, indicating that a match has been found by the external device. The pin also has a meaning during asynchronous operation, but in this case it can only be an output. This output is called the break detect signal and goes high whenever a character consisting of all 0s is received.

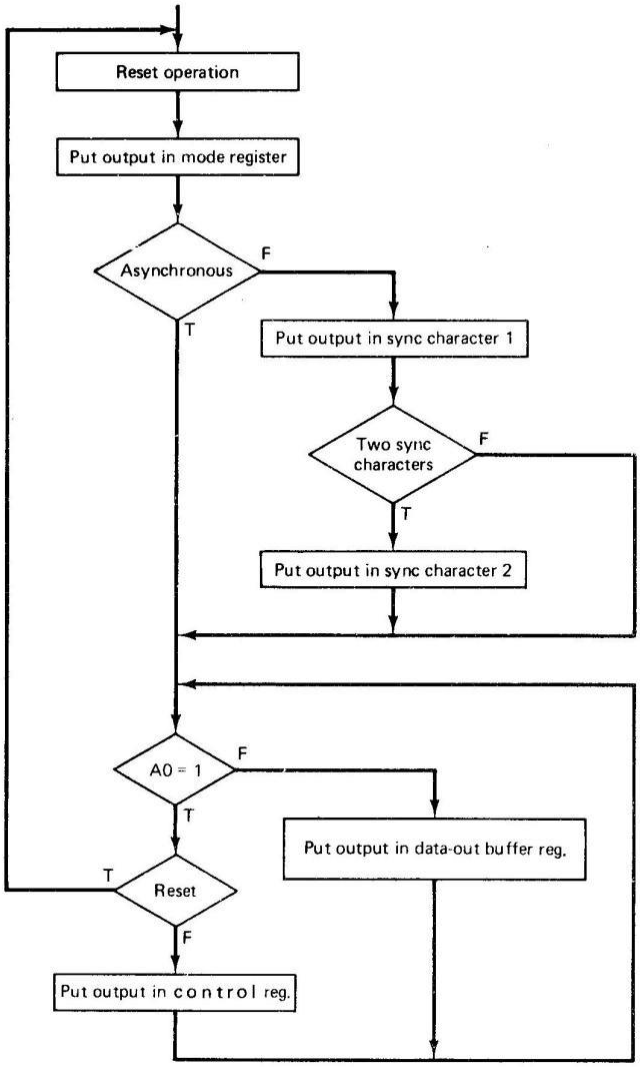
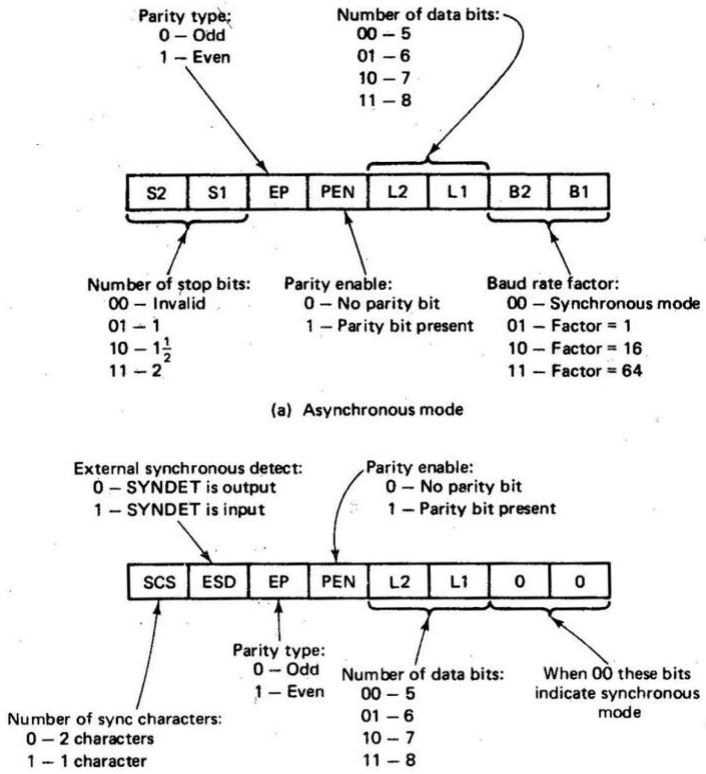


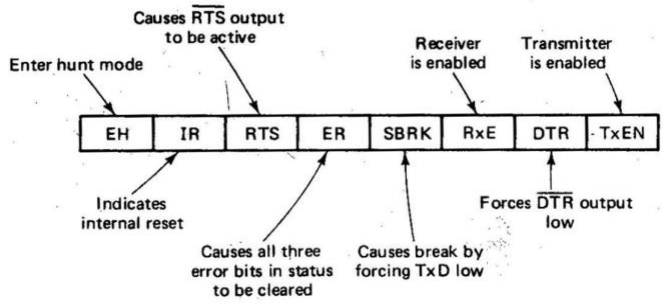
Figure 4.8: Flowchart of the disposition of output

The format for the control register is given in Figure 4.10. Bit 0 of this register must be 1 before data can be output and bit 2 must be 1 before data can be received. Programmed answering of a modem is accomplished by setting bit 1 to 1 since this forces the DTR pin to 0 and the complement of DTR is normally connected to the CD line from the modem. Bit 3 equal to 1 forces TxD to 0, thus causing break characters to be transmitted. Setting bit 4 to 1 causes all the error bits in the status register to be cleared (the bits that are set when framing,

overrun, and parity errors occur). Bit 5 is used for sending a Request to Send signal to a modem. If the complement of the RTS pin is connected to a modem's CA line, then a 1 put in bit 5 will cause the CA line to go high. Setting bit 6 causes the 8251 A to be reinitialized and the reset sequence to be reentered (i.e., a return is made to the top of the flowchart shown in Figure 4.8 and the next output will be to the mode register). Bit 7 is used only with the synchronous mode. When set, it causes the 8251A to begin a bit-by-bit search for a sync character or sync characters.

Figure 4.9: Format of the mode register

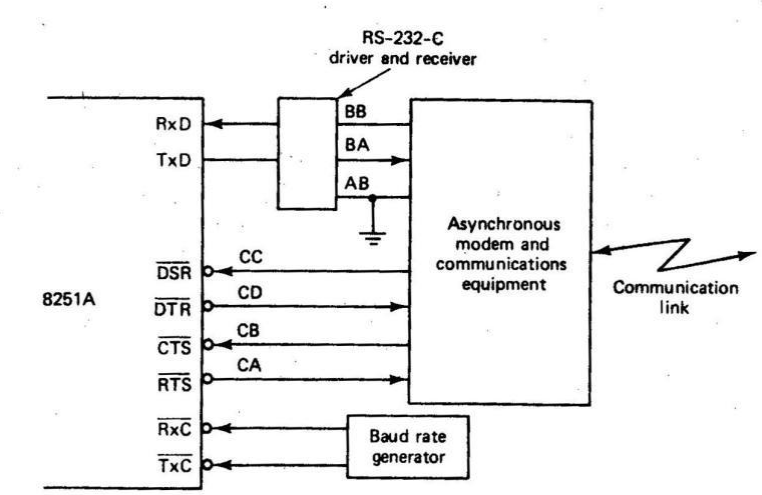




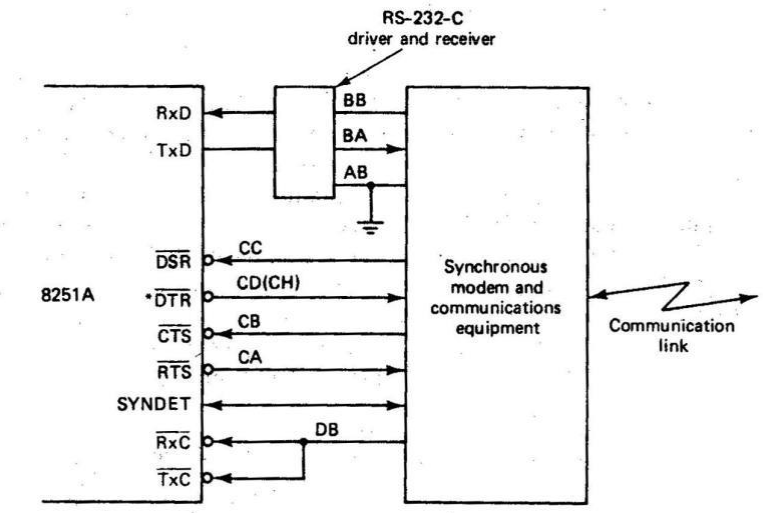
Note: In all cases action is taken when the bit is set to 1.

Figure 4.10: Format of the control register

Typical connections to modems for asynchronous and synchronous transmissions are shown in Figure 4.11. With regard to the synchronous connections, it is assumed that the timing is controlled by the modem and its related communications equipment. Also, if this equipment is used to detect the sync character(s) at the beginning of a received message, then it can inform the 8251A of its success over the SYNDET line. On the other hand, if 8251A searches for the sync character(s), then it can use the SYNDET line as an output to tell the modem that the sync character(s) has been found. To satisfy the RS-232 -C standard, drivers and receivers are needed to convert the TTL-compatible signals at TxD and RxD to the proper voltage levels.



(a) Asynchronous modem connection



*May also be used for rate selection – modem line CH.

(b) Synchronous modem connection

NOTE: Control line interfacing depends on modem specifications.

Figure 4.11: 8251 A modem connections

A program sequence which initializes the mode register and gives a command to enable the transmitter and begin an asynchronous transmission of 7-bit characters followed by an even-parity bit and 2 stop bits is:

```
MOV  AL, 11111010B

OUT  51H, AL

MO   AL, 00110011B
OUT  51H, AL
```

This sequence assumes that the mode and control registers are at address 511-1 and the clock frequencies are to be 16 times the corresponding baud rates. The sequence:

```
MOV  AL, 00111000B
OUT  51H, AL
MOV  AL, 16H
OUT  51H, AL
OUT  51H, AL
MOV  AL, 10010100B
OUT  51H, AL
```

would cause the same 8251A to be put in synchronous mode and to begin searching for two successive ASCII sync characters. As before, the characters are to consist of 7 data bits and an even parity bit, but there will, of course, be no stop bits. The format of the status register is given in Figure 4.12. Bits 1, 2, and 6 reflect the signals on the RxRDY, TxE, and SYNDET pins. TxRDY indicates that the data-out buffer is empty. Unlike the TxRDY pin, this bit is not affected by the CTS input pin or the TxEN control bit. RxRDY indicates that a character has been received and is ready to be input to the processor. Either the TxRDY and RxRDY bits

can be used for programmed I/O or the signals on the corresponding pins can be connected to interrupt request lines to provide for interrupt I/O. The TxRDY bit is automatically cleared when a character- is made available for transmitting and the RxRDY bit is automatically cleared when the character that set it is input by the processor. Bit 2 indicates that the transmitter shift register is waiting to be sent a character from the data-out buffer register. During synchronous transmissions, while this bit is set, the transmitter will take its data from the sync character registers until data are put in the data-out buffer register. Bits 3, 4, and 5 indicate parity, overrun, and framing errors, respectively. When an error is detected, the bit having the corresponding error type will be set to 1. If the complement of the DSR pin is connected to the Data Set Ready (CC) line, then bit 7 reflects the state of the modem and is 1 when the modem is turned on and is in its data mode.

Figure 4.13 gives a typical program sequence which uses programmed I/O to input 80 characters from the 8251A, whose data buffer register's address is 0050, and put them in the memory buffer beginning at LINE. The inner loop continually tests the RxRDY bit until it is set by a character being put in the data-in buffer register. Then the newly arrived character is moved to the buffer and the error bits are checked. If the present character arrived before the previous character was input or a parity or framing error occurred during transmission, then the input ceases and a call is made to an error routine that would presumably examine the individual error bits, print an appropriate message, and clear the error bits.

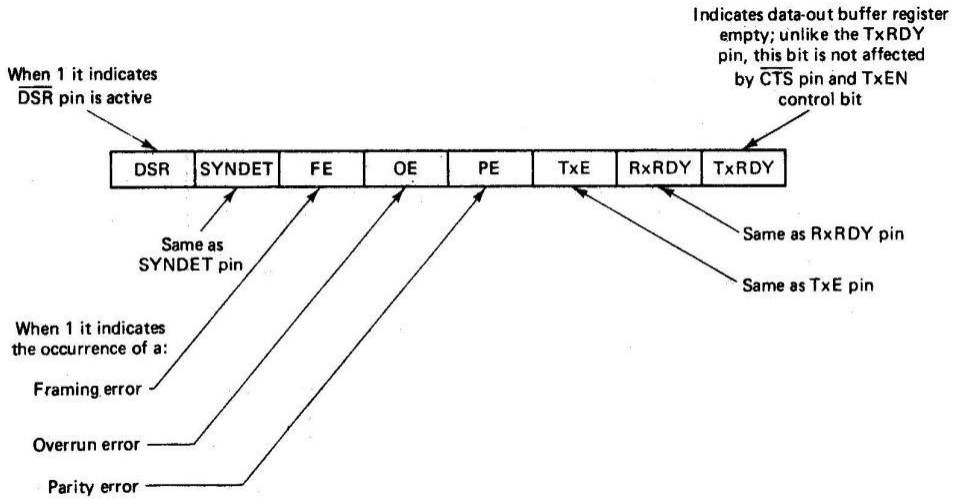


Figure 4.12: Format of the status register

```
        MOV   AL,00110101B       ;ENABLE TRANSMITTER AND RECEIVER
        OUT   51H;AL             ;AND CLEAR ERROR BITS
        MOV   DI,0               ;INITIALIZE INDEX
        MOV   CX,80              ;PUT COUNT IN CX
BEGIN:  IN    AL,51H             ;WAIT FOR INPUT
        TEST  AL,02H
        JZ    BEGIN
        IN    AL,50H             ;INPUT CHARACTER AND
        MOV   LINE[DI],AL        ;PUT IN LINE BUFFER
        INC   DI
        IN    AL,51H             ;CHECK ERROR
        TEST  AL,00111000B       ;BITS AND
        JNZ   ERROR              ;IF NO ERROR IS FOUND
        LOOP  BEGIN              ;CONTINUE INPUTTING
        JMP   SHORT EXIT
ERROR:  CALL  NEAR PTR ERR_ROUT  ;ELSE CALL ERR_ROUT
EXIT:    .
         .
         .
```

Figure 4.13: Inputting a line of characters through an 8251A

Because inputting a character automatically resets the RxRDY bit, unless another character is received before the inner loop is reentered, the inner loop must cycle until the RxRDY bit is reset to 1 by the next incoming character. If the incoming characters have fewer than 8 bits, the unused MSBs in the data buffer register are always zeroed. Also, the parity bit is not passed to the processor and checks for parity errors can only be made by examining the parity error bit in the status register. On output, if a character is less than 8 bits long, the unneeded MSBs in the data-out buffer register are ignored