# TRASMITTING AND RECEIVING A CHARACTER

BY

SUBATHRA S

# TRANSMITTING AND RECEIVING A CHARACTER

## OBJECTIVE

To initialize 8253 and 8251A and to check the transmission and reception of a character

## APPARATUS REQUIRED

- 8085 Microprocessor trainer kit
- 8251 Programmable communication interface
- 8253 Programmable interval timer
- Power supply

## DESCRIPTION

The program first initializes 8253 in to give an output clock frequency of 150KHz at channel 0 which will give a 9600 baud rate of 825551A. The 8251Ais initialized to a dummy mode command. The internal reset to 8251A is then provided since the 8251A is in the command mode now. Then the 8251A is interfaced as said with data 4E and 37.

The program after initializing will read the status register and check for 8251A. If the transmitted buffer is empty then it will send A1 to the serial port and then check for a character in the receiver buffer. If some character is present, then it is received and stored at location 4200. If the serial port is not proper, the program will be in a constant loop either in the transmission mode or in the reception.

## ALGORITHM

1.Initialize 8253 and give it to clock output as input to 8251A.

2.By resetting the accumulator reset 8251A in command instruction format us well as in mode instruction format.

3.The condition of instruction format 8251A in mode instruction format is

a.8bit data b. No parity

c.16x baud rate factor d.1 stop bit

4.The condition for initializing 8251A in command instruction format is

a. Reset error flag

b. Enable transmission

c. Make RTS and DTR active low

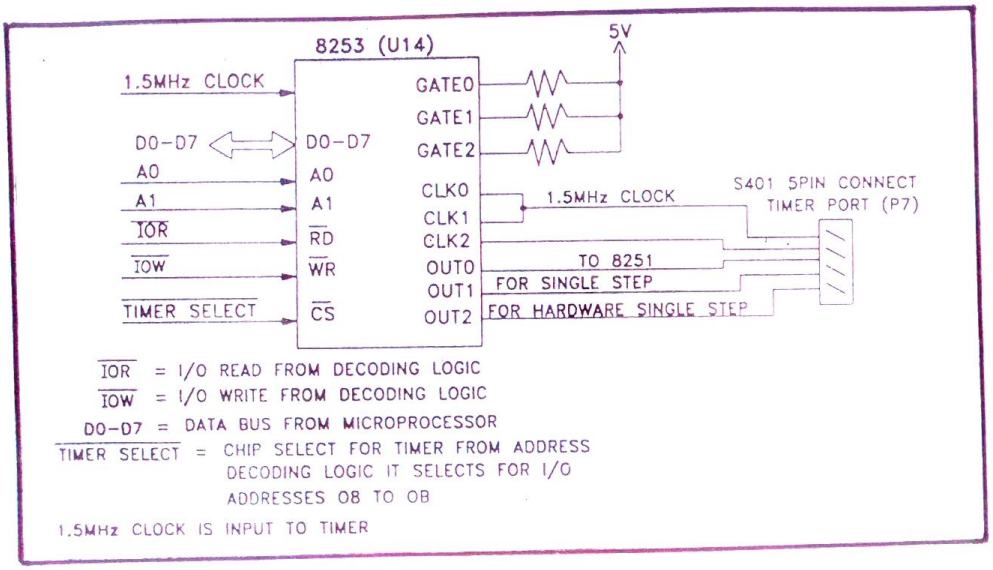
5.Check the status of transmission empty in 8251A if the transmission is empty then transmit the data

6. Check the status of receiver ready in 8251A of the receiver is ready then receive the data

7.Store the received data in the specified location

8.Stop the execution

BASIC BLOCK DIAGRAM OF 8253 INTERFACE




## CONTORL WORD FORMAT 8253

<table>
<tr><th colspan="2">DATA BIT</th><th>DESCRIPTION</th></tr>
<tr><td>$D_0$</td><td>BCD</td><td>BINARY/BCD</td></tr>
<tr><td>$D_1$</td><td>M0</td><td rowspan="3">MODE
<table>
<tr><th>M2</th><th>M1</th><th>M0</th><th>MODE</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>2</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>3</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>5</td></tr>
</table></td></tr>
<tr><td>$D_2$</td><td>M1</td></tr>
<tr><td>$D_3$</td><td>M2</td></tr>
<tr><td>$D_4$</td><td>RL0</td><td rowspan="2">READ/LOAD
<table>
<tr><th>RL1</th><th>RL0</th><th></th></tr>
<tr><td>0</td><td>0</td><td>Latch</td></tr>
<tr><td>0</td><td>1</td><td>LSB</td></tr>
<tr><td>1</td><td>0</td><td>MSB</td></tr>
<tr><td>1</td><td>1</td><td>LSB/MSB</td></tr>
</table></td></tr>
<tr><td>$D_5$</td><td>RL1</td></tr>
<tr><td>$D_6$</td><td>SC0</td><td rowspan="2">
<table>
<tr><th>SC0</th><th>SC1</th><th>CH#</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>2</td></tr>
<tr><td>1</td><td>1</td><td>X</td></tr>
</table></td></tr>
<tr><td>$D_7$</td><td>SC1</td></tr>
</table>

## COMMAND INSTRUCTION FORMAT OF 8251A

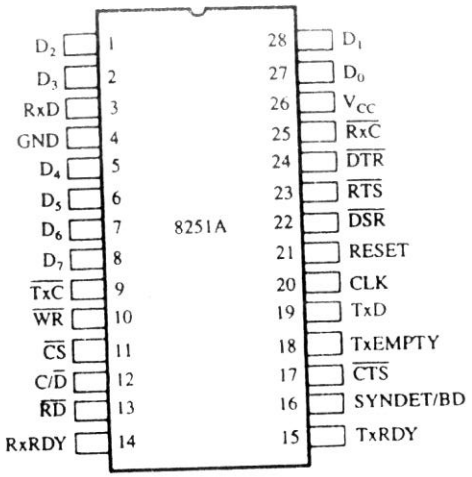
| DATA BIT | | DESCRIPTION |
|---|---|---|
| $D_0$ | TXEN | TRANSMIT ENABLE<br>1 – ENABLE ; 0 – DISABLE |
| $D_1$ | DTR | DATA TERMINAL READY<br>'HIGH' WILL FORCE DTR OUTPUT TO 0 |
| $D_2$ | RXE | RECEIVE ENABLE<br>1 – ENABLE ; 0 – DISABLE |
| $D_3$ | SBRK | SEND BREAK CHARACTER<br>1 = FORCE TXD "LOW"<br>0 = NORMAL OPERATION |
| $D_4$ | ER | ERROR RESET<br>1 = RESET ERROR FLAGS OF, PE, OE, FE |
| $D_5$ | RTS | REQUEST TO SEND "HIGH " WILL FORCE RTS OUPUT TO 0 |
| $D_6$ | IR | INTERNAL RESET "HIGH" RETURNS 8251A TO MODE INSTRUCTION FORMAT |
| $D_7$ | EH | ENTER HUNT MODE<br>1 – ENABLE A SEARCH FOR SYNCRONOUS CHARACTER<br>( HAS NO EFFECT IN ASYNCRONOUS MODE ) |

## STATUS WORD FORMAT 8251A

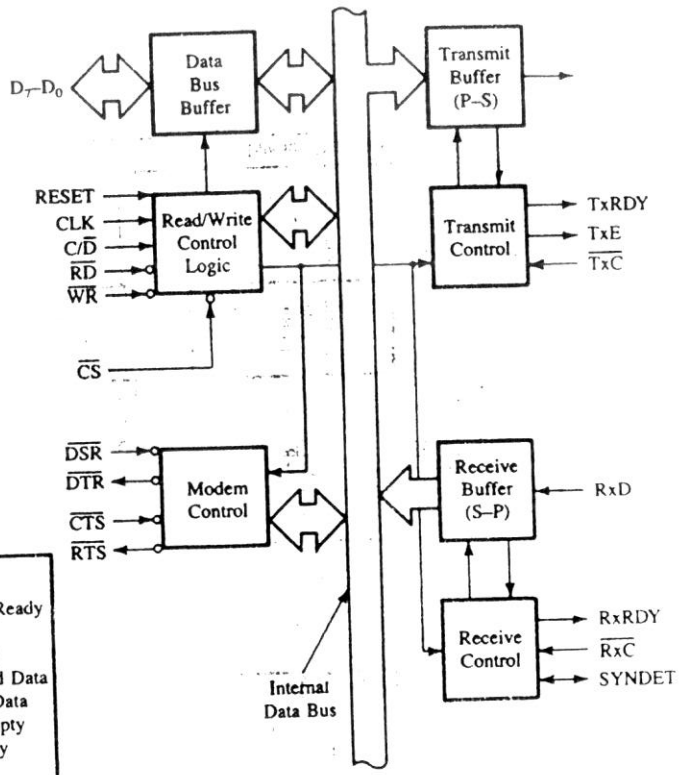
| DATA BIT | | DESCRIPTION |
|---|---|---|
| $D_0$ | TXRDY | TRANSMITTER READY<br>TXRDY status bit has different meanings from the TXRDY output pin. The former is not conditioned by CTS & TXEN (i.e) TXRDY status bit = DB buffer empty TXRDY pin out = DB buffer empty. (CTS – 0) (TXEN – 1). |
| $D_1$ | RXRDY | RECEIVER READY<br>This bit indicates that the 8251A contains a character that is ready to be input to the CPU |
| $D_2$ | TXEMPTY | TRANSMITTER EMPTY<br>When the 8251A has no character to transmit this bit will go high |
| $D_3$ | PE | PARITY ERROR<br>The PE flag is set when a parity error is detected. It is reset by the ER bit of the command instruction. PE does not inhibit operation of the 8251A |
| $D_4$ | OE | OVERFLOW ERROR<br>The OE flag is set when the CPU does not read a character before the next one becomes available. OE is reset by the ER bit of the command instruction. OE does not inhibit the operation of 8251A however the previously overrun character is lost. |
| $D_5$ | FE | FRAMING ERROR(Asynchronous mode only)<br>The FE flag is set when a valid stop bit is not detected at end of every character. It is reset by the ER BIT of the command instruction. FE does not inhibit the operation of 8251A. |
| $D_6$ | SYNDET | SYNC DETECT<br>This pin is used in synchronous mode for syndet and is used in asynchronous mode for break detect. |
| $D_7$ | DSR | DATA SET READY<br>Indicates that the DSR is at zero level. |

# THE 8251A : BLOCK DIAGRAM, PIN CONFIGURATION & DESCRIPTION

Pin Configuration




Block Diagram




Pin Names

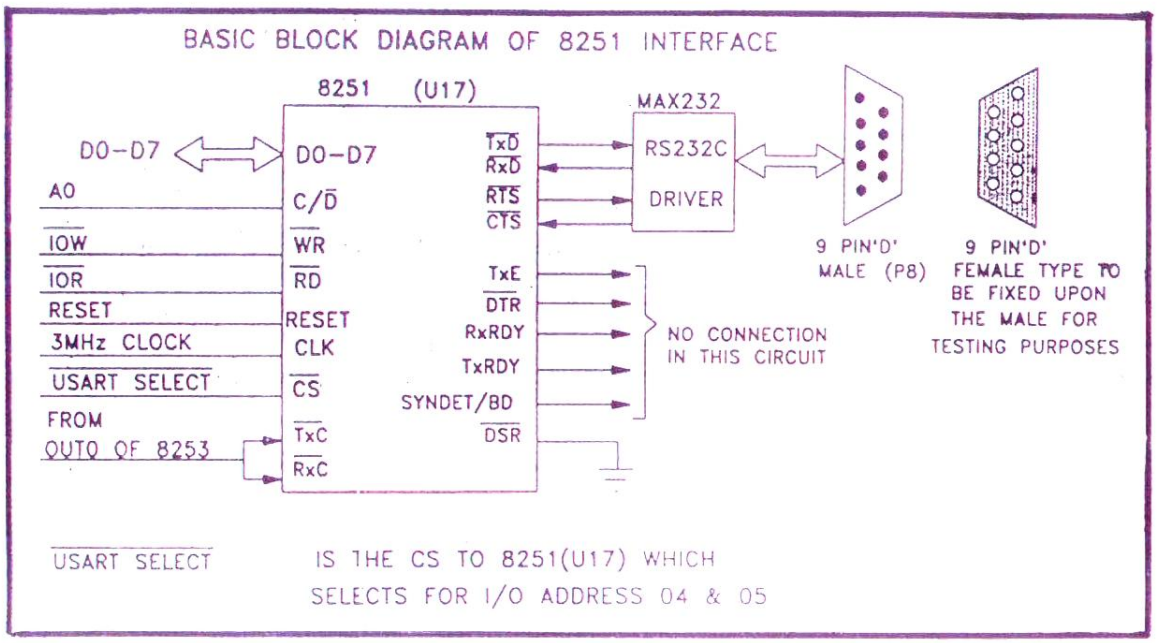
| | |
|---|---|
| $D_7$–$D_0$ | Data Bus (8 bits) |
| C/D | Control or Data is to be Written or Read |
| RD | Read Data Command |
| WR | Write Data or Control Command |
| CS | Chip Enable |
| CLK | Clock Pulse (TTL) |
| RESET | Reset |
| TxC | Transmitter Clock |
| TxD | Transmitter Data |
| RxC | Receiver Clock |
| RxD | Receiver Data |
| RxRDY | Receiver Ready |
| TxRDY | Transmitter Ready |

| | |
|---|---|
| DSR | Data Set Ready |
| DTR | Data Terminal Ready |
| SYNDET/BD | Sync Detect/ Break Detect |
| RTS | Request to Send Data |
| CTS | Clear to Send Data |
| TxE | Transmitter Empty |
| $V_{CC}$ | +5 Volt Supply |
| GND | Ground |

# ASYNCRONOUS MODE OF 8251A

<table>
<tr><th colspan="2">DATA BIT</th><th>DESCRIPTION</th></tr>
<tr><td>$D_0$</td><td>B1</td><td rowspan="2">Baud rate Factor
<table>
<tr><th>B2</th><th>B1</th><th></th></tr>
<tr><td>0</td><td>0</td><td>Synchronous mode</td></tr>
<tr><td>0</td><td>1</td><td>1X</td></tr>
<tr><td>1</td><td>0</td><td>16X</td></tr>
<tr><td>1</td><td>1</td><td>64X</td></tr>
</table></td></tr>
<tr><td>$D_1$</td><td>B2</td></tr>
<tr><td>$D_2$</td><td>L1</td><td rowspan="2">Character Length
<table>
<tr><th>L2</th><th>L1</th><th></th></tr>
<tr><td>0</td><td>0</td><td>5 bits</td></tr>
<tr><td>0</td><td>1</td><td>6 bits</td></tr>
<tr><td>1</td><td>0</td><td>7 bits</td></tr>
<tr><td>1</td><td>1</td><td>8 bits</td></tr>
</table></td></tr>
<tr><td>$D_3$</td><td>L2</td></tr>
<tr><td>$D_4$</td><td>PEN</td><td>Parity Enable<br>1 – ENABLE<br>0 – DISABLE</td></tr>
<tr><td>$D_5$</td><td>EP</td><td>Even parity Generation Check<br>0 – ODD<br>1 – EVEN</td></tr>
<tr><td>$D_6$</td><td>S1</td><td rowspan="2">No of Stop bit
<table>
<tr><th>S2</th><th>S1</th><th></th></tr>
<tr><td>0</td><td>0</td><td>Invalid</td></tr>
<tr><td>0</td><td>1</td><td>1 bit</td></tr>
<tr><td>1</td><td>0</td><td>1.5 bits</td></tr>
<tr><td>1</td><td>1</td><td>2 bits</td></tr>
</table></td></tr>
<tr><td>$D_7$</td><td>S2</td></tr>
</table>

# CIRCUIT DIAGRAM



# ASSEMBLY LANGUAGE PROGRAM

| ADDRESS | LABEL | MNEMONICS | OPCODE/OPERAND |
|---|---|---|---|
| 4100 | | **MVI A,$36_H$** | 3E 36 |
| 4102 | | **OUT TMRCNT** | D3 0B |
| 4104 | | **MVI A,$0A_H$** | 3E 0A |
| 4106 | | **OUT TMRCHO** | D3 08 |
| 4108 | | **XRA A** | AF |
| 4109 | | **OUT TMRCHO** | D3 0A |
| 410B | | **XRA A** | AF |
| 410C | | **OUT UATCNT** | D3 05 |
| 410E | | **MVI A,$04_H$** | 3E 40 |
| 4110 | | **OUT UATCNT** | D3 05 |
| 4112 | | **MVI A,$4E_H$** | 3E 4E |
| 4114 | | **OUT UATCNT** | D3 05 |
| 4116 | | **MVI A,$37_H$** | 3E 37 |
| 4118 | | **OUT UATCNT** | D3 05 |
| 411A | **LOOP** | **IN UATCNT** | DB 05 |
| 411C | | **ANI $04_H$** | E6 04 |
| 411E | | **JZ LOOP** | CA 1C C1 |
| 4121 | | **MVI A,$41_H$** | 3E 41 |
| 4123 | | **OUT UATDAT** | D3 04 |
| 4125 | **LOOP1** | **IN UATCNT** | DB 05 |
| 4127 | | **ANI $02_H$** | E6 02 |
| 4129 | | **JZ LOOP1** | CA 27 C1 |
| 412C | | **IN UATDAT** | DB 04 |
| 412E | | **STA $4200_H$** | 32 00 42 |
| 4131 | | **HLT** | 76 |

# PROGRAM TRACE

<table>
<tr><th>LABEL</th><th>MNEMONICS</th><th>DESCRIPTION</th></tr>
<tr><td></td><td>MVI A,$36_H$</td><td>$36_H$ is moved to accumulator.
REGISTERS
<table>
<tr><td>A</td><td>36</td><td>XX</td><td>F</td></tr>
<tr><td>B</td><td>XX</td><td>XX</td><td>C</td></tr>
<tr><td>D</td><td>XX</td><td>XX</td><td>E</td></tr>
<tr><td>H</td><td>XX</td><td>XX</td><td>L</td></tr>
</table>
➢ INITIALISATION OF 8253<br>
Initializing the COUNTER 0 of the PIT 8253 in MODE 3 by writing the control word as $36_H$. A square wave was generated similar to clock pulse.<br><br>
CONTROL WORD FORMAT OF 8253
<table>
<tr><td rowspan="2">COMMENT</td><td colspan="2">SELECT COUNTER</td><td colspan="2">READ/ LOAD</td><td colspan="3">MODE</td><td>BCD/ BINARY COUNT</td></tr>
<tr><td>SC1</td><td>SC0</td><td>RL1</td><td>RL0</td><td>M2</td><td>M1</td><td>M0</td><td>COUNT</td></tr>
<tr><td>DATA BITS</td><td>$D_7$</td><td>$D_6$</td><td>$D_5$</td><td>$D_4$</td><td>$D_3$</td><td>$D_2$</td><td>$D_1$</td><td>$D_0$</td></tr>
<tr><td></td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>OBSERVE</td><td colspan="2">C0UNTER 0</td><td colspan="2">LSB/MSB</td><td colspan="3">MODE 3</td><td>BINARY COUNT</td></tr>
</table>
The 8253 give its clock output as input to 8251A.<br>
The 8253 gives an output clock frequency=150 kHz at counter 0 which will give a baud rate 9600 baud rate of 8251A.<br><br>
<u>NOTE:</u><br>
❖ BAUD is the number of signal units/second.<br>
❖ BAUD RATE is a way of specifying the rate at which data bits are shifted in or out of a serial device.<br>
❖ BAUD RATE for a device such as the 8251A is defined as one over the time per bit.<br>
❖ Commonly used BAUD RATES are 110,300,1200,2400,4800,9600 and 19,200Bd.</td></tr>
<tr><td></td><td>OUT TMRCNT</td><td>Write the transmitter control word.<br><u>HINTS:</u><br>Opcode used for TMRCNT is $0B_H$ .</td></tr>
<tr><td></td><td>MVI A,$0A_H$</td><td>Lower order byte of the count is loaded in to Accumulator as $0A_H$.
REGISTERS
<table>
<tr><td>A</td><td>0A</td><td>XX</td><td>F</td></tr>
<tr><td>B</td><td>XX</td><td>XX</td><td>C</td></tr>
<tr><td>D</td><td>XX</td><td>XX</td><td>E</td></tr>
<tr><td>H</td><td>XX</td><td>XX</td><td>L</td></tr>
</table>
The baud rate of 8251A is 9600 baud/sec & frequency is 150 kHz. The 8253 frequency is 1500 kHz.<br>
Clock baud rate<br>
Count = 1500/150 = $10_{10}$ = $OA_H$<br>
Hence we are applying $OA_H$ to accumulator<br>
• In order to synchronize the 8251A & 8253<br>
• 8253 to produce an output of 150 kHz clock for initializing 8251A.</td></tr>
</table>

<table>
<tr><td></td><td>OUT<br>TMRCHO</td><td>Load TIMER CHANNEL 0 with lower order byte.<br><u>HINTS:</u><br>❑ Opcode used for TMRCHO is $08_H$.<br>❑ After loading both LSB/MSB, the square wave will be generated.</td></tr>
<tr><td></td><td>XRA A</td><td>Higher order byte of the count is loaded in to Accumulator as $00_H$.<br>REGISTERS<br>A | 00 | XX | F<br>B | XX | XX | C<br>D | XX | XX | E<br>H | XX | XX | L</td></tr>
<tr><td></td><td>OUT<br>TMRCHO</td><td>Load TIMER CHANNEL 0 with higher order byte.<br>At the end of the count, it generates a pulse that can be used to interrupt the microprocessor.</td></tr>
<tr><td></td><td>XRA A</td><td>➢ RESETTING THE 8251A<br>By clearing the accumulator, reset the 8251A in COMMAND INSTRUCTION FORMAT as well as in MODE INSTRUCTION FORMAT.<br>REGISTERS<br>A | 00 | XX | F<br>B | XX | XX | C<br>D | XX | XX | E<br>H | XX | XX | L</td></tr>
<tr><td></td><td>OUT<br>UATCNT</td><td>Universal Synchronous Asynchronous Receiver Transmitter Count 8251A.</td></tr>
<tr><td></td><td>MVI A,$40_H$</td><td>REGISTERS<br>A | 40 | XX | F<br>B | XX | XX | C<br>D | XX | XX | E<br>H | XX | XX | L<br><br>COMMAND INSTRUCTION FORMAT OF 8251A<br><br>
<table>
<tr><th colspan="2">INSTRUCTION BIT</th><th>BIT</th><th>DESCRIPTION</th></tr>
<tr><td>Enter Hunt Mode</td><td>EH</td><td>0</td><td>Does not enter Hunt Mode</td></tr>
<tr><td>Internal Reset</td><td>IR</td><td>1</td><td>Internal reset high returns 8251A to mode instruction format, will not force o/p to zero</td></tr>
<tr><td>Request To Send</td><td>RTS</td><td>0</td><td>Will not force RTS O/P to 0</td></tr>
<tr><td>Error Reset</td><td>ER</td><td>0</td><td>Does not Reset Error flags OF, PE, OE, FE</td></tr>
<tr><td>Send Break Character</td><td>SBRK</td><td>0</td><td>Normal operation</td></tr>
<tr><td>Receive Enable</td><td>RXE</td><td>0</td><td>Receive Disable</td></tr>
<tr><td>Data Terminal Ready</td><td>DTR</td><td>0</td><td>Will not force DTR O/P to 0</td></tr>
<tr><td>Transmit Enable</td><td>TXEN</td><td>0</td><td>Disable Transmit</td></tr>
</table></td></tr>
</table>

| | | |
|---|---|---|
| | OUT UATCNT | Universal Synchronous Asynchronous Receiver Transmitter Count 8251A. |
| | MVI A,$4E_H$ | **REGISTERS**<br>**A** **4E** XX F<br>B XX XX C<br>D XX XX E<br>H XX XX L<br>➢ INTIALIZATION OF 8251A<br>The condition of instruction 8251A in mode instruction format is as follows<br>▪ 8 bit data<br>▪ no parity<br>▪ 16X baud rate factor<br>▪ 1 stop bit<br>**ASYNCRONOUS MODE OF 8251A** |

| No of Stop bit | | Even parity Generation Check | Parity Enable | Character Length | | Baud rate Factor | |
|---|---|---|---|---|---|---|---|
| S2 | S1 | EP | PEN | L2 | L1 | B2 | B1 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 bit | | 0-odd | 0-Disable | 8 bit | | 16X | |

| | | |
|---|---|---|
| | OUT UATCNT | Universal Synchronous Asynchronous Receiver Transmitter Count 8251A. |
| | MVI A,$37_H$ | **REGISTERS**<br>**A** **37** XX F<br>B XX XX C<br>D XX XX E<br>H XX XX L<br>The condition of initializing 8251A in command instruction format is as follows<br>▪ Reset error flag<br>▪ Enable transmission & reception<br>▪ Make RTS & DTS active low |

**COMMAND INSTRUCTION FORMAT OF 8251A**

| HEX CODE / INSTRUCTION | EH | IR | RTS | ER | SBRK | RXE | DTR | TXEN |
|---|---|---|---|---|---|---|---|---|
| $37_H$ | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 |
| | 3 | | | | 7 | | | |

| INSTRUCTION BIT | | BIT | DESCRIPTION |
|---|---|---|---|
| Enter Hunt Mode | EH | 0 | Does not enter Hunt Mode |
| Internal Reset | IR | 0 | No internal Reset |
| Request To Send | RTS | 1 | Request to send 'High' will force RTS O/P to 0 |
| Error | ER | 1 | Reset Error flags OF, PE, OE, FE |

<table>
<tr><td></td><td></td><td>
<table>
<tr><td>Reset</td><td></td><td></td><td></td></tr>
<tr><td>Send Break Character</td><td>SBRK</td><td>0</td><td>Normal operation</td></tr>
<tr><td>Receive Enable</td><td>RXE</td><td>1</td><td>Receive enable</td></tr>
<tr><td>Data Terminal Ready</td><td>DTR</td><td>1</td><td>Data terminal ready force DTR O/P to 0</td></tr>
<tr><td>Transmit Enable</td><td>TXEN</td><td>1</td><td>Transmit enable</td></tr>
</table>
</td></tr>
<tr><td></td><td>OUT UATCNT</td><td>Universal Synchronous Asynchronous Receiver Transmitter Count 8251A.</td></tr>
<tr><td>LOOP</td><td>IN UATCNT</td><td>Input $37_H$ to accumulator.<br>REGISTERS<br>A | 37 | XX | F<br>B | XX | XX | C<br>D | XX | XX | E<br>H | XX | XX | L<br>➢ Check 8251A TxEMPTY and then send the data 41 to 8251A</td></tr>
<tr><td></td><td>ANI $04_H$</td><td>And immediate the accumulator content with $04_H$.<br>$37_H$ => 0011 0111<br>$04_H$ => 0000 0100<br>-----------<br>0000 0100 => $04_H$<br>-----------<br>STATUS WORD FORMAT
<table>
<tr><td>HEX CODE / INSTRUCTION</td><td>DSR</td><td>SYNDET</td><td>FE</td><td>OE</td><td>PE</td><td>TxEMPTY</td><td>RxRDY</td><td>TxRDY</td></tr>
<tr><td>$04_H$</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td></td><td colspan="4">0</td><td colspan="4">4</td></tr>
</table>
REGISTERS<br>A | 04 | XX | F<br>B | XX | XX | C<br>D | XX | XX | E<br>H | XX | XX | L</td></tr>
<tr><td></td><td>JZ LOOP</td><td>Here, if ANI instruction gives non-zero value implies the Tx is empty (i.e. TxEMPTY = enabled) then transmits the data to 8251A, otherwise, loop until it is empty.</td></tr>
<tr><td></td><td>MVI A,$41_H$</td><td>INPUT DATA to be transmitted.<br>REGISTERS<br>A | 41 | XX | F<br>B | XX | XX | C<br>D | XX | XX | E<br>H | XX | XX | L</td></tr>
<tr><td></td><td>OUT UATDAT</td><td>Output the data.</td></tr>
<tr><td>LOOP1</td><td>IN UATCNT</td><td>Input the control/status word.</td></tr>
</table>

| | | |
|---|---|---|
| | | **REGISTERS**<br>**A** **00** XX F<br>B XX XX C<br>D XX XX E<br>H XX XX L<br>➢ Check 8251A RxRDY and hence get the data and store at 4200 |
| | ANI $02_H$ | **STATUS WORD FORMAT**<br>HEX CODE / INSTRUCTION: DSR SYNDET FE OE PE TxEMPTY RxRDY TxRDY<br>$02_H$: 0 0 0 0 0 0 1 0<br>0 2<br>RxRDY bit indicates that 8251Acontains a character that is ready to be input to CPU.<br>**REGISTERS**<br>**A** **02** XX F<br>B XX XX C<br>D XX XX E<br>H XX XX L |
| | JZ LOOP1 | Here, if<br>RxRDY = 0 => it is not ready, so go on looping until it is not zero<br>RxRDY = 1 => it is ready, so stop looping. |
| | IN UATDAT | Data is input in to the receiver.i.e. Transmitted data is being received.<br>**REGISTERS**<br>**A** **41** XX F<br>B XX XX C<br>D XX XX E<br>H XX XX L |
| | STA $4200_H$ | Store the received data in the specified location.<br>**REGISTERS**<br>**A** **41** XX F<br>B XX XX C<br>D XX XX E<br>H XX XX L<br>**MEMORY**<br>41FD xx<br>41FE xx<br>41FF xx<br>4**200** **41**<br>4201 xx<br>The program will be in a constant loop either in the transmission mode or in the reception mode. |
| | HLT | Stop the execution. |

## VERIFICATION

Check at location 4200 using substitute command. If a content of location is $41_H$ ,then the serial port is OK. Else check the loop back connector and verify the status of RST*, CST*, TXD and RXD using an oscilloscope.

| MODE | ADDRESS | DATA |
|---|---|---|
| *Input* | 4124 $_H$ | 41 $_H$ |
| *Output* | 4200 $_H$ | 41 $_H$ |

## REFERENCE

1. Ramesh S.Gaonkar, Microprocessor Architecture, Programming, and Applications, Fourth Edition, Penram International Publishing (India), 2000.

2. S.Subathra, “Advanced Microprocessor Laboratory”, Record work, Adhiparashakthi Engineering College, Melmaruvathur, October 2002

3. S.Subathra, “Programming in 8085 Microprocessor and its applications – An Innovative Analysis”, Technical Report, Adhiparashakthi Engineering College, Melmaruvathur, March 2003

4. Micro-85 EB, User Manual, Version – 3.0, CAT #M85 EB-002, VI Microsystems Pvt. Ltd., Chennai.