Lecture 12

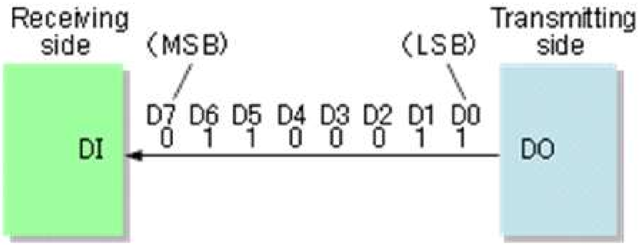
Serial Input/Output,
8251 USART

Topics

- Serial Input/Output Interfacing
- 8251 USART Chip

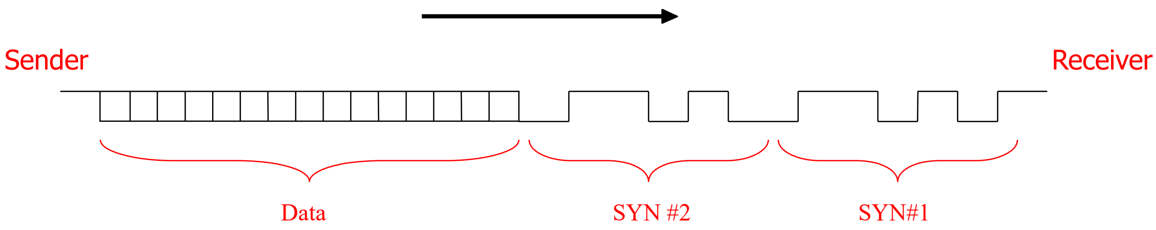
Serial Input/Output Interfacing

- Serial communication between interface devices requires one wire.
- Bits are transferred one at a time.
- There are two types of serial transfer.
  - Synchronous serial transfer
  - Asynchronous serial transfer
- Example :
  Two serial interface devices are connected.
  The MSB (Most Significant Bit) is sent first.



Synchronous Serial Transfer

- In synchronous serial transfer, two units share a common clock frequency.
- The transmitter and the receiver can setup transmission rate and synchronize their clocks.
- Transmission is Message-based.
  - The sender doesn't transmit characters as they occur.
  - Sender stores the message in a buffer.
  - Synchronization occurs at the beginning of a message.
  - A message contains multiple characters (bytes).



Synchronous Serial Transfer

- The following methods use synchronous serial transfer.
  - I2C (Inter-Integrated Circuit)
  - SPI (Serial Peripheral Interface)
- Used for communication between integrated circuits.

Asynchronous Serial Transfer

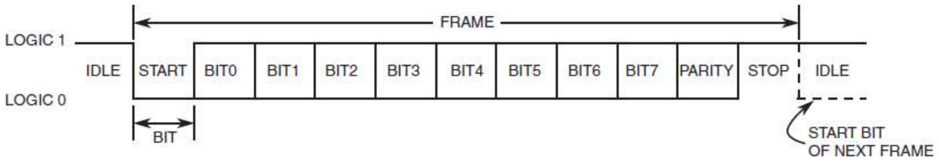
- In asynchronous serial transfer, the transmission is Frame-based.
- Each character (8-bit data) is transmitted as a separate frame.
- The recevier device must be able to recognize when transmission starts, and when transmission ends.
- The following methods use asynchronous serial transfer.
  - RS-232 (Recommended Standard 232)
  - UART (Universal Asynchronous Receiver-Transmitter)

Synchronous & Asynchronous Serial Transfer

- The methods below can use both synchronous and asynchronous serial transfer.
  - USART (Universal Synchronous Asynchronous Receiver-Transmitter)
  - USB (Universal Serial Bus)
- The transfer configuration is done by using a control command.
- Used for communication with a computer.

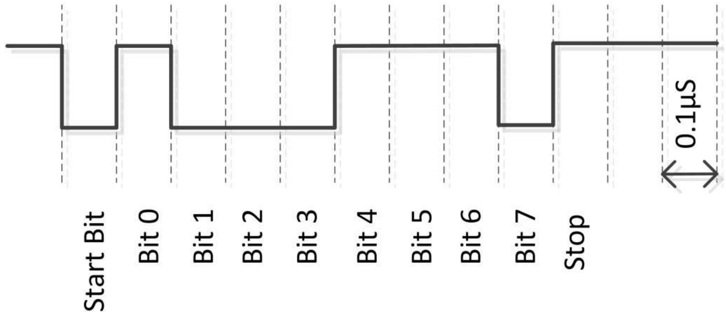
Descriptions of Frame Bits

- Special bits are added at both ends of the character code frame.
- Each character code frame consists of following parts.
  - Start bit: always 0, indicates beginning of a character
  - Information bits: data (8 bits)
  - Parity bit: 1 or 0, depending on the Parity method (Odd/Even).
    - Parity (equality) bit is a checking bit (verification) appended to data bits to make the sum of all the data bits, including the checking bit itself.
    - Parity bit can be either odd (1) or even (0).
  - Stop bit: always 1



Asynchronous Transfer Rules for a Frame

- When a character is not being sent, the transmisson line is kept in the logical 1 state.
- Character transmission is detected from the start bit (0).
- Information bits follow the start bit.
- Transmitter calculates the parity bit (0 or 1) and transmits it.
- One or two stop bits are sent.
- Parameters of the serial example below are:
  8 Data bits, 1 Stop bit, No Parity bit



Speed Rate of Transmission

- There needs to be an agreement on how long each bit stays on the line.

- Speed parameter: The rate of transmission is usually measured in bits per second (Baud).

- Baud Rate = Bits/Second
  (bps : bits per second)

- Baud Rate is the standard transmission speed unit in serial communication. It is equivalent to the bps.

Example Table for Baud Rates

Baud (bps)	Number of stop bits	Frames / Second	Bit time (milli seconds)
1200	1	120	0.83
2400	1	240	0.42
4800	1	480	0.21
9600	1	960	0.10
19200	1	1920	0.05

Example calculation of bit time for 1200 bps:
Bit time = (1 / 1200) * 10^3
= 0.83 ms

Example: Calculating Bit Rate and Data Rate

Suppose the Baud Rate is 19200 bps in a serial communication.

QUESTIONS

1) Calculate how many milli seconds one bit should stay on the line. (Bit Rate)

2) Calculate how many milli seconds one frame should stay on the line. (Data Rate)

3) Calculate how many frames can be transferred in one second.

Example: Calculating Bit Rate and Data Rate

ANSWERS

1) Bit Rate calculation:

Time for one bit is = 1/19200 seconds
= 0.00005 seconds
= 0.00005 * 10^3 milli seconds
= 0.05 milli seconds

2) Data Rate calculation :

Assume there is 1 Start Bit, 1 Stop bit, 8 data bits,
and no parity bits in the frame (Total length of frame is 10 bits).

Time for one frame is = Bit rate * Frame length
= 0.05 * 10 bits = 0.5 milli seconds

3) Calculation of frames per second :

19200 bits / 10 bits = 1920 frames per second
≈ 2K frames per second

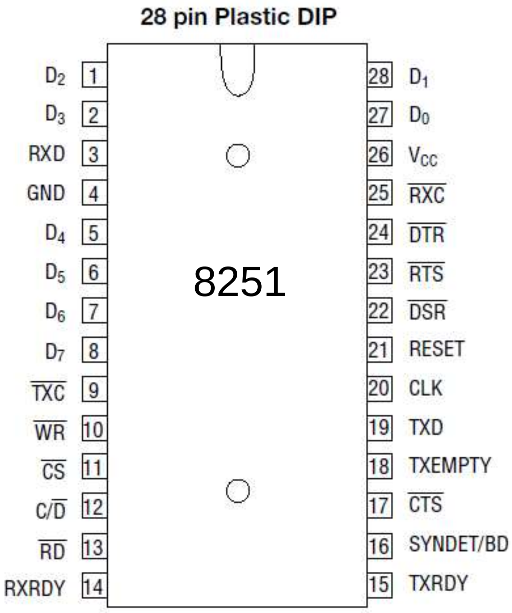
Topics

- Serial Input/Output Interfacing
- 8251 USART Chip

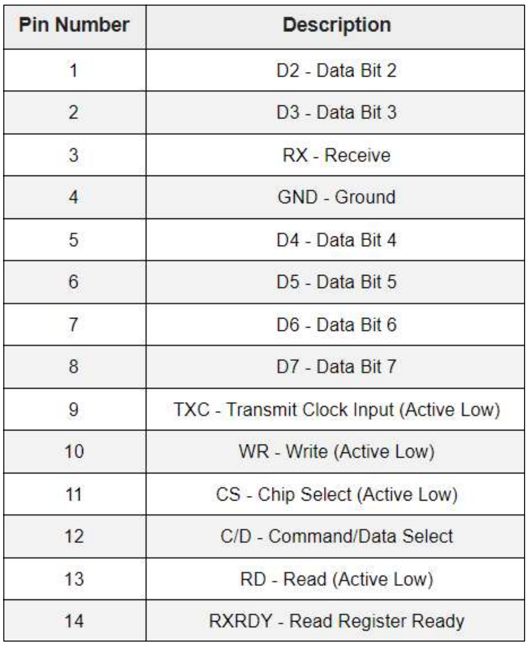
8251 Universal Synchronous Asynchronous Receiver Transmitter (USART)

- Intel 8251 Universal Synchronous Asynchronous Receiver Transmitter (USART) is an interface chip used for serial communication.
- Used between the microprocessor and a peripheral device to transmit serial data.
- 8251 chip takes data serially from peripheral devices (outside devices) and converts into parallel data. After converting the data into parallel form, it transmits data to microprocessor (8086 CPU).
- 8251 chip receives parallel data from microprocessor and converts it into serial form. After converting data into serial form, it transmits serial data to outside device (peripheral).
- Synchronous communication speed is up to 64 Kbaud.
- Asynchronous communication speed is up to 38 Kbaud.
- Allows transmitting/receiving operations under buffered configuration.
- Supports error detection (parity, overrun, and framing).

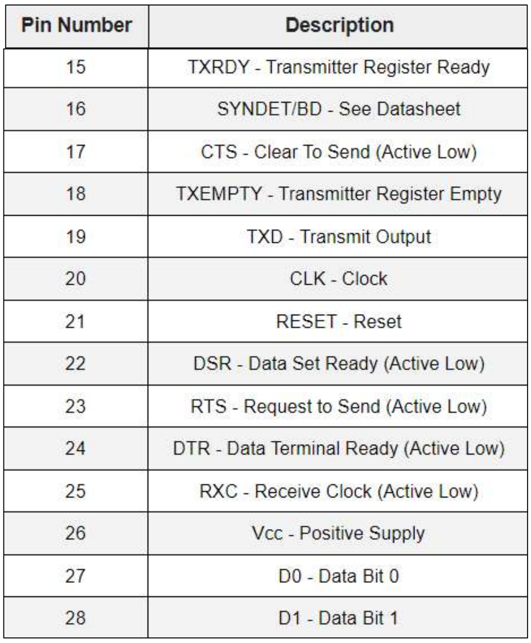
8251 USART Pins



8251 Pin Descriptions



Pin Number	Description
1	D2 - Data Bit 2
2	D3 - Data Bit 3
3	RX - Receive
4	GND - Ground
5	D4 - Data Bit 4
6	D5 - Data Bit 5
7	D6 - Data Bit 6
8	D7 - Data Bit 7
9	TXC - Transmit Clock Input (Active Low)
10	WR - Write (Active Low)
11	CS - Chip Select (Active Low)
12	C/D - Command/Data Select
13	RD - Read (Active Low)
14	RXRDY - Read Register Ready



Pin Number	Description
15	TXRDY - Transmitter Register Ready
16	SYNDET/BD - See Datasheet
17	CTS - Clear To Send (Active Low)
18	TXEMPTY - Transmitter Register Empty
19	TXD - Transmit Output
20	CLK - Clock
21	RESET - Reset
22	DSR - Data Set Ready (Active Low)
23	RTS - Request to Send (Active Low)
24	DTR - Data Terminal Ready (Active Low)
25	RXC - Receive Clock (Active Low)
26	Vcc - Positive Supply
27	D0 - Data Bit 0
28	D1 - Data Bit 1

Pin Functions

- Data Bus : Two-directional, tri-state, 8-bit Data Bus.
  Pins allow transfer of bytes between the CPU and the 8251.
- RD (Read) : A low on this input allows the CPU to read data or status bytes from 8251.
- WR (Write) : A low on this input allows the CPU to write data or command word to the 8251.
- CLK (Clock) : The CLK input is used to generate internal device timing.
- RESET : A high on this input forces the 8251 into an idle mode. The device will remain at idle until a new set of control words is written into the 8251.
- C/D (Control / Data) : The input in conjuction with the WR and RD inputs, informs the 8251 that the word on the Data Bus is either a data character control word or status information.

8251 Registers

- The 8251 USART chip contains following registers.
- Each register is 8 bits.
    - Data buffer register
    - Control register / Status register
      (Control register and Status register are shared.)
- To implement serial communication with 8086 CPU, the Assembly program must inform the 8251 chip all details such as communication mode, baud rate (in case of asynchronous mode), number of stop bits, parity etc.
- Prior to data transfer, a set of control bits must be loaded into the Control register of 8251 chip.

Control Register

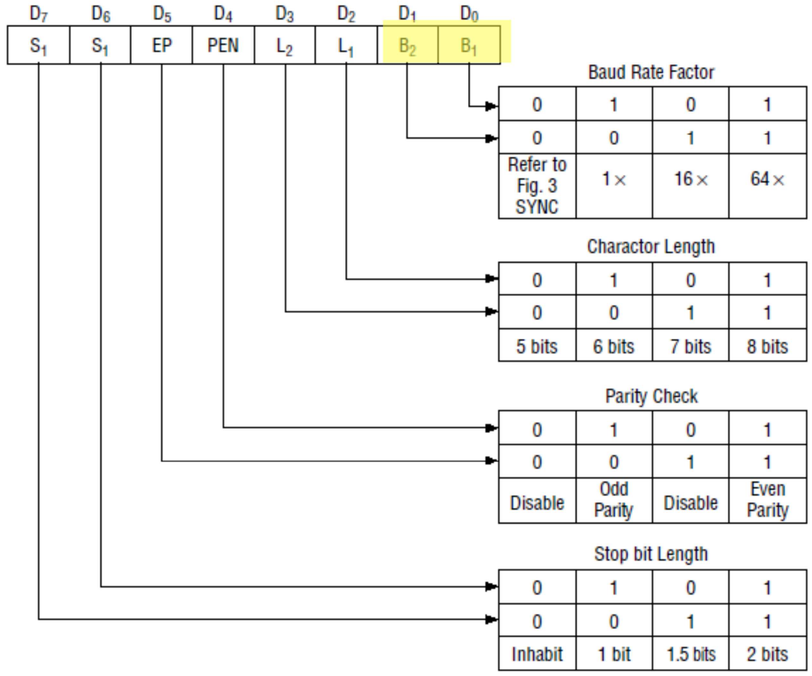
- The 8251 chip is configured by the Control Word Register.
- Control register is used twice :
    - First usage corresponds to Mode Instruction format.
    - Second usage corresponds to Command Instruction format.
- The content of control word register determines followings.
    - Synchronous or asynchronous operation.
    - Baud rate
    - Number of bits per character
    - Number of stop bits
    - Type of parity

Mode Instruction

- Mode instruction is used for setting the function of 8251.
- Writing of a control word after resetting is recognized as a "mode instruction."
- Items set by mode instruction are as follows:
  - Synchronous / Asynchronous mode selection
(When D0=0 and D1=0, then mode is Synchronous.)
(Otherwise, mode is Asynchronous.)
  - Stop bit length (Asynchronous mode only)
  - Character length (number of bits)
  - Parity bit
  - Baud rate (Asynchronous mode only)
  - Internal/external synchronization (Synchronous mode only)
  - Number of synchronous characters (Synchronous mode only)

Configuration of Mode Instruction (Asynchronous)

When both D1 and D0 bits are zero, then Synchronous mode is selected. Otherwise, Asynchronous is selected.



D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
S_1	S_1	EP	PEN	L_2	L_1	B_2	B_1

Baud Rate Factor

0	1	0	1
0	0	1	1
Refer to Fig. 3 SYNC	1×	16×	64×

Charactor Length

0	1	0	1
0	0	1	1
5 bits	6 bits	7 bits	8 bits

Parity Check

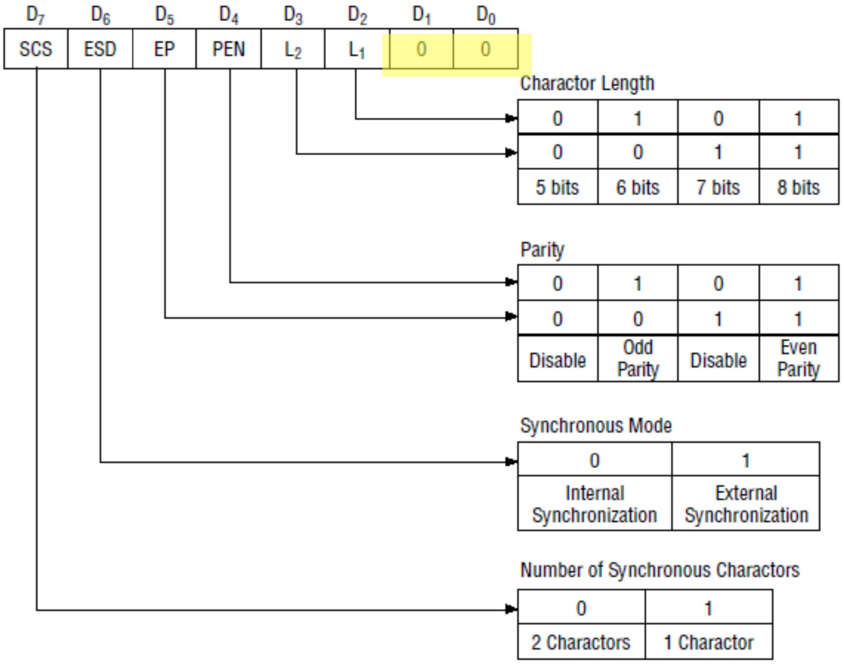
0	1	0	1
0	0	1	1
Disable	Odd Parity	Disable	Even Parity

Stop bit Length

0	1	0	1
0	0	1	1
Inhabit	1 bit	1.5 bits	2 bits

Configuration of Mode Instruction (Synchronous)

When both D1 and D0 bits are zero, then Synchronous mode is selected. Otherwise, Asynchronous is selected.



D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SCS	ESD	EP	PEN	L_2	L_1	0	0

Charactor Length

0	1	0	1
0	0	1	1
5 bits	6 bits	7 bits	8 bits

Parity

0	1	0	1
0	0	1	1
Disable	Odd Parity	Disable	Even Parity

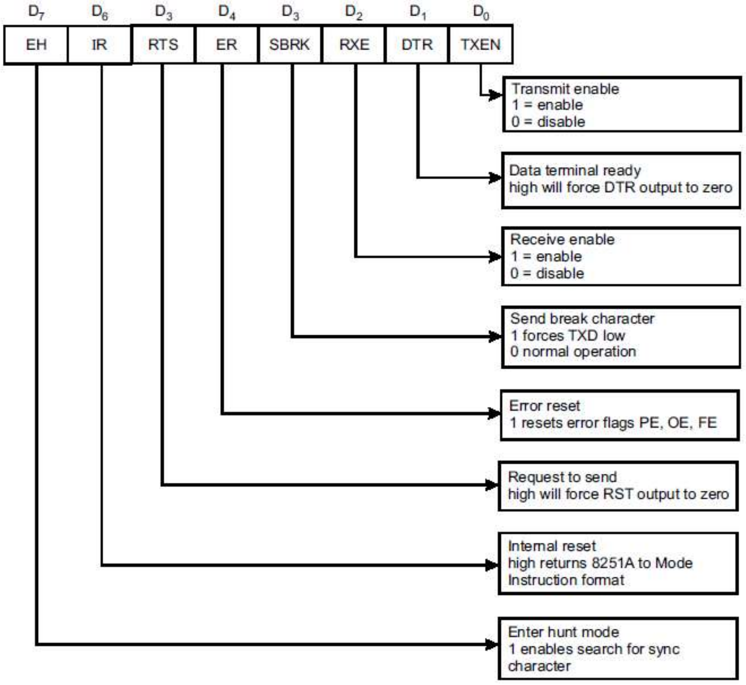
Synchronous Mode

0	1
Internal Synchronization	External Synchronization

Number of Synchronous Charactors

0	1
2 Charactors	1 Charactor

Configuration of Command



D_7	D_6	D_3	D_4	D_3	D_2	D_1	D_0
EH	IR	RTS	ER	SBRK	RXE	DTR	TXEN

Transmit enable
1 = enable
0 = disable

Data terminal ready
high will force DTR output to zero

Receive enable
1 = enable
0 = disable

Send break character
1 forces TXD low
0 normal operation

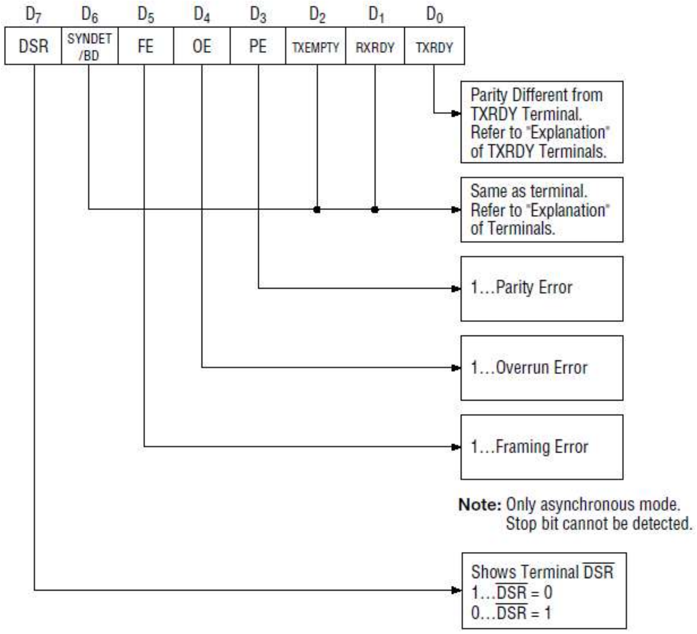
Error reset
1 resets error flags PE, OE, FE

Request to send
high will force RST output to zero

Internal reset
high returns 8251A to Mode Instruction format

Enter hunt mode
1 enables search for sync character

Configuration of Status Word



D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
DSR	SYNDET /BD	FE	OE	PE	TXEMPTY	RXRDY	TXRDY

Parity Different from TXRDY Terminal. Refer to "Explanation" of TXRDY Terminals.

Same as terminal. Refer to "Explanation" of Terminals.

1…Parity Error

1…Overrun Error

1…Framing Error

Note: Only asynchronous mode. Stop bit cannot be detected.

Shows Terminal $\overline{DSR}$
1…$\overline{DSR}$ = 0
0…$\overline{DSR}$ = 1

Modem Control Signals

- DSR (Data Set Ready) : The input signal is used to test modem conditions such as Data Set Ready.
- DTR (Data Terminal Ready) : The output signal is used to tell modem that Data Terminal is ready.
- RTS (Request to Send) : The output signal is asserted to begin transmission.
- CTS (Clear to Send) : A low on the input enables the 8251 to transmit serial data if the TxE bit in the command byte is set to 1.

Transmitter Signals

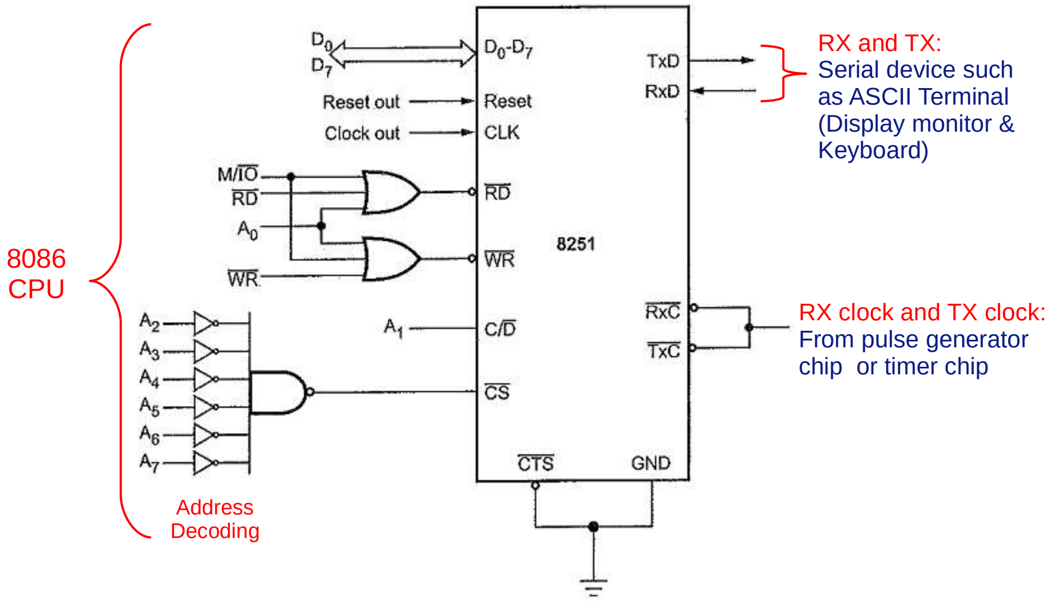
- TxD (Transmit Data) : The output signal outputs a composite serial stream of data on the falling edge of TxC.
- TxRDY (Transmitter Ready) : The output signal indicates the CPU that the transmitter is ready to accept a data character.
- TxE (Transmitter Empty) : The output signal indicates that the transmitter has no character to transmit.
- TxC (Transmitter Clock) : The clock input controls the rate at which the character is to be transmitted.

Receiver Signals

- RxD (Receiver data) : The input receives a composite serial stream of data on the rising edge of RxC.
- RxRDY (Receiver Ready) : The output indicates that the 8251 contains a character that is ready to be input to the CPU.
- RxC (Receiver Clock) : The clock input controls the rate at which the character is to be received.

Example: Interfacing of 8251 in Isolated I/O

- The 8251 serial interface chip is connected to the 8086 CPU chip.
- A0-A7 address lines from CPU are used for the Chip Selection ($\overline{CS}$) of 8251.
- RD and WR signals are activated when M/$\overline{IO}$ signal is low, indicating I/O bus cycle.
- Only lower data bus (D0 – D7) from CPU is used, because 8251 is 8-bit device.
- Reset out signal from Clock generator chip is connected to the Reset signal of the 8251.



8086 CPU

Address Decoding

RX and TX: Serial device such as ASCII Terminal (Display monitor & Keyboard)

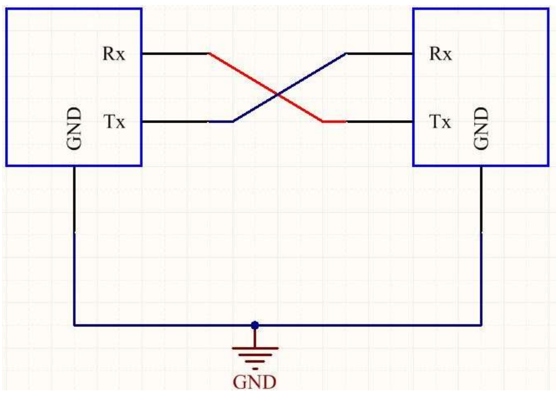
RX clock and TX clock: From pulse generator chip or timer chip

USART and Serial Device Connections

- The following diagram shows connections between 8251 USART chip and a Serial Device such as an ASCII screen.

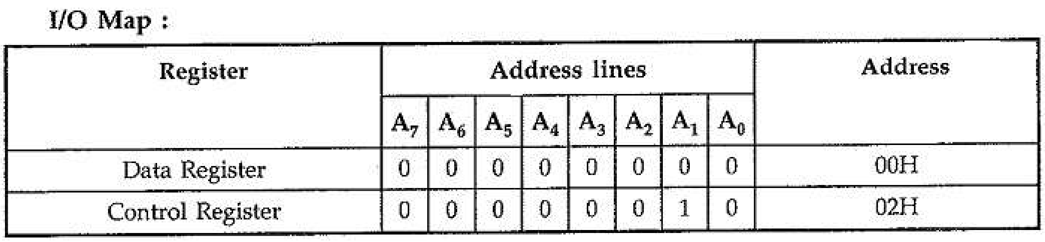
8251 USART chip

Serial Device



8251 Port Addresses

The followings are register addresses (port addresses) of 8251 chip.



I/O Map :

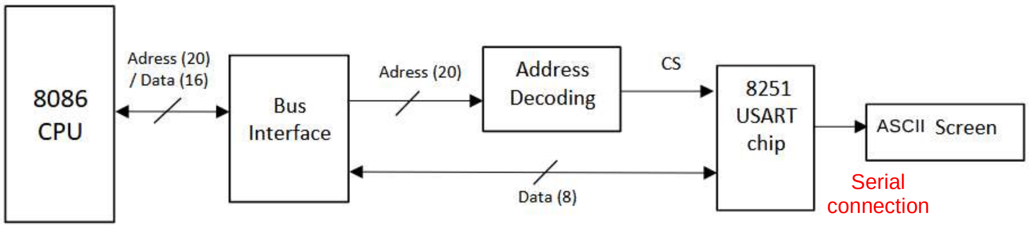
Register	Address lines								Address
	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
Data Register	0	0	0	0	0	0	0	0	00H
Control Register	0	0	0	0	0	0	1	0	02H

Application Example

- Write a 8086 Assembly program to send ASCII characters of a string variable, to the data port of 8251 USART chip.
- Suppose a serial device (such as ASCII screen) is connected to the 8251 chip.
- The register addresses of 8251 chip are as follows.
  - Data register address : 00h
  - Control register address : 02h
- Configure the 8251 chip with the specifications below.
  - Mode : Asynchronous
  - Transmitter/Receiver direction : Transmitter
  - Parity: Even
  - Number of stop bits: 2
  - Character length : 8 bits
  - Baud rate : 1 K bits per second

8251 Chip and 8086 CPU Connection

- Data connection between CPU and 8251 interface chip is parallel.
- Data connection between 8251 chip and the ASCII screen is serial.



Serial connection

Parallel connection

Asynchronous Mode Configuration

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hex Value
1	**1**	**1**	**1**	**1**	**1**	**0**	**1**	**FD**
2 Stop Bits		Even Parity Enabled		Character Length is 8 Bits		Baud Rate is 1 Kbps		

Command Word for Transmission

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Hex Value
0	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**01**
EH	IR	RTS	ER	SBRK	RxE	DTR	TxEN	

Program

Part1

```
; 8251 USART chip port (register) addresses:
DATA_PORT    EQU  00h
KONTROL_PORT EQU  02h

.model small
.data
Cumle  DB   'Testing'  ; String to send to the serial device (ASCII screen)

.code

.STARTUP

MOV SI, OFFSET Cumle   ; SI points to string variable
MOV CL, 7              ; Length of string is in CL

MOV AL, 0FDh           ; Load Asynchronous mode word to D0–D7
OUT KONTROL_PORT, AL   ; Mode is selected

MOV AL, 01h            ; Load Command word to D0–D7
OUT KONTROL_PORT, AL   ; Command is selected (transmit enabled)

BEKLE:
IN AL, KONTROL_PORT    ; Read status of 8152 chip
AND AL, 01H            ; Check the transmitter enable bit
JZ BEKLE               ; If bit is zero then wait for transmitter to be ready
```

Program

Part2

```
MOV AL, [SI]           ; If transmitter is ready, send next byte of string data
OUT DATA_PORT, AL      ; One character of string is transmitted
INC SI                 ; Point to next byte in string
DEC CL                 ; Decrement counter
JNZ BEKLE              ; If CL is not zero, continue for next byte in string

.EXIT
END                    ; End of file
```