LECTURE NOTES (E- CONTENTS) for

B.Sc. III Electronics (2021-22)

Semester: V Paper- V DSE 1005E1

Linear Integrated Circuits, 8051 Microcontroller Interfacing and Embedded C

Section II: 8051 Microcontroller Interfacing and Embedded C

Unit- 3: Serial Communication in 8051

Prepared and Circulated for

B .Sc. III Electronics Students

BY

Dr. C. B. Patil

Associate Professor, Department of Electronics

Vivekanand College (Autonomous), Kolhapur

(For Private Circulation only)

UNIT 3

Serial Communication in 8051

Introduction

In this chapter the serial communication facility of 8051 is discussed. The 8051 has an on chip UART (Universal Asynchronous Reception Transmission) port used for serial data transmission and reception with different speeds (baud rates). Two pins of the 8051 are available for serial communication. The RXD(P3.0) pin is used for serial data reception and the TXD (P3.1)pin is used for serial data reception.

Modes of Data Transmission/Reception

There are two basic modes of data transmission and reception: parallel and serial.

In parallel mode, more than one bits are transmitted/received at a time. Therefore, it speed of data transmission/reception is high. It uses more number of lines. This mode is useful for short range data transmission/reception. Using P0, P1, P2 and P3 8051 transmit/receive parallel 8-bit data.

Serial communication on the other hand, uses only one or two data lines to transfer data. In serial communication the data is sent one bit at a time. It is generally used for long distance communication. Using pins TXD and RXD, 8051 transmit/receive serial data.

Serial communication uses two methods for data transmission/reception.

1) Asynchronous data transmission: used for character-oriented transmissions

2) Synchronous data transmission: used for block-oriented data transfers

Asynchronous serial communication and data framing

In the asynchronous method, each character is placed between start and stop bits. This process is called as data framing. The start bit is always one bit, but the stop bit can be one or two bits. The start bit is always a 0 (low) and the stop bit(s) is 1 (high). For example, look at Fig.1. in which the ASCII character “A” (8-bit binary 0100 0001) is framed between the start bit and a single stop bit. Here LSB is sent out first.

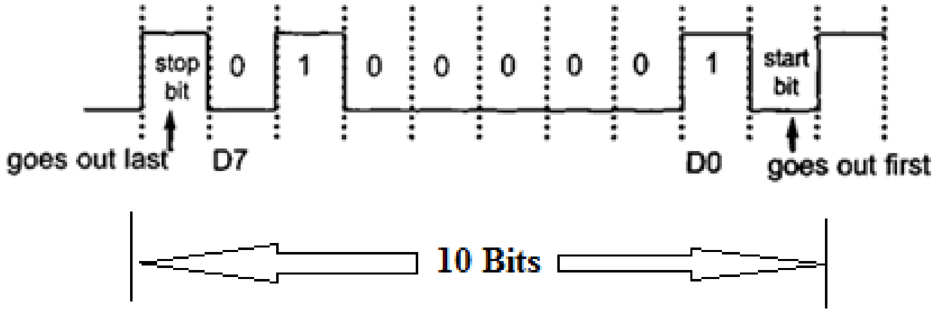


Fig. 1. Asynchronous data framing

Baud Rate

In serial communication, data is transmitted 1 bit at a time. The number of bits transmitted per second known as Baud Rate and its unit is bps (bit/sec). The standard baud rate are defined by the IBM PCXT and it is 110, 150, 300, 600, 1200, 2400, 4800, 9600 etc. The 8051 can be useful to transfer and receive serial data at different baud rates using software instructions. The timer1 mode2 (8-bit auto reload) mode of 8051 is used to set the baud rate in serial communication.

Steps for calculate baud rate for serial communication in 8051

- The 8051 microcontroller transfers and receives data serially at different baud rates. The baud rate in the 8051 is programmable and can be set using Timer 1 mode 2.

- To generate standard baud rate, external crystal of the frequency 11.0592 MHz is used. So it can generate a clock frequency of 11.0592 MHz.
- The machine cycle is calculated by dividing clock frequency by 12. Here machine cycle is 921.6 KHz.
- Before it is used by Timer 1 to set the baud rate, the 8051 UART circuitry divides the machine cycle frequency of 921.6 kHz by 32. Therefore, 921.6 kHz divided by 32 gives baud frequency of 28,800 Hz.
- When Timer 1 is used to set the baud rate it must be programmed in mode 2 (8-bit auto reload). To get baud rates compatible with PC, we must load TH1 with any one values for the given baud rate as shown in table 1.

Sr. No.	Baud Rate	TH1(Decimal)	TH1(Hex)	Baud Frequency
1	9600	-3	FDH	28,800/3
2	4800	-6	FAH	28,800/6
3	2400	-12	F4H	28,800/12
4	1200	-24	E8H	28,800/24

Table 1: TH1 values and Baud Rate

- These values of baud rate are compatible with PC's COM port.

SBUF register

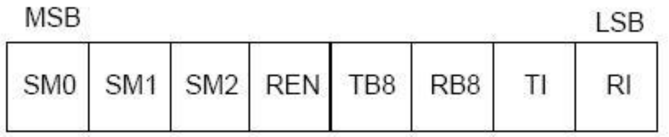
It is an important 8-bit special function register used for serial communication in the 8051. During transmission of the serial data it must be placed in SBUF register. Similarly during reception of the data, SBUF holds received data. The SBUF register can be accessed like any other register in the 8051. E.g. 1) SBUF = ‘A’ ; Load SBUF register with ASCII of ‘A’ character

2) mybyte = SBUF ;save value from SBUF into variable mybyte

During transmitting data via TXD pin SBUF frames the data by adding start and stop bit. During reception of the data using RXD pin SBUF deframes the data by eliminating start and stop bit. The framing and deframing of the data is depends on the selected mode of serial communication.

SCON register

The SCON **(serial control) register** is an 8-bit bit as well as byte addressable register of 8051. It is used during programming of serial communication. Fig 2. shows the SCON register internal bit structure.



MSB							LSB
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Fig. 2: SCON register

SM0, SM1

SM0 and SM1 are bit number D7 and D6 of the SCON register respectively. These two bits determine the modes of serial data transmission/reception. Table 1. 2 shows SM0, SM1 bits and selected mode name

SM0	SM1	Mode	Name
0	0	Mode 0	Shift Register
0	1	Mode1	8-bit Standard UART 1 start and one stop bit
1	0	Mode2	9-bit UART (Baud rate Fixed)
1	1	Mode3	9-bit UART (Baud rate Variable)

Table 2: bits of SCON register

The mode 1 is standard and it is mostly widely used mode than other.

SM2

SM2 is the D5 bit of the SCON register. It is multiprocessing communication bit. This bit enables/disables the multiprocessing capability of the 8051. It is mostly useful in mode 2 and mode 3 serial communication mode.

REN

REN (receive enable) is D4 bit of the SCON register. This bit is important in serial data reception. If it is '1', it allows to receive data through the RXD pin. If it is '0' the data reception disabled. If we want both transmission and reception REN must be '1'.

TB8*

The TB8 is bit D3 of SCON register. If we select mode 2 or mode 3 it is 8th bit, because total 9 bits are transmitted in mode 2 and mode 3.

RB8*

The RB8 is bit D2 of SCON register. If we select mode 2 or mode 3 it stored 8th bit ,because total 9 bits are received in mode 2 and mode 3.

TI

TI (transmit interrupt) is bit D1 of the SCON register. If this bit is set to '1'indicates data transmission is over and it is ready to transfer next byte of data. For next data transmission it must be cleared to '0'.

RI

RI (receive interrupt) is the D0 bit of the SCON register. If this bit is set to '1' indicates data reception is over. After removing start and stop data is placed in SBUF register and it is ready to receive next byte of data. For next data reception it must be clear to '0'.

*In mode 2 and mode 3 the total 11 bits (1 start bit +1 bit TB8/RB8+ 8 bit SBUF+ 1 stop bit) are transmitted/received.

Modes in serial communication

The mode of serial communications are set by using SM0 and SM1 bit of SCON register (Please see table 1.2). There are four modes in serial communication data transmission and reception in 8051 which are given below.

a) Mode 0: It is a Shift Register Mode. It is half duplex mode. In this RXD pin is used for data transmission and reception. The TXD pin is used as a clock out pin. The total 8 bits are transmitted/received. The baud frequency is fixed and it is F/12. Where F is the crystal frequency.

b) Mode 1: It is an 8-bit standard UART mode. It is full duplex mode. In this RXD pin is used for data reception. The TXD pin is used data transmission. The total 10 bits (1 start bit + 8 bit data + 1 stop bit) are transmitted/received. The baud frequency is variable. The timer 1 mode 2 (8-bit auto reload) is used to generate baud rate.

c) Mode 2: It is a 9-bit standard UART mode (baud frequency fixed). The total 11 bits [1 start bit+TB8/RB8+ 8 bit data(SBUF) +1 stop bit] are transmitted/received. The baud frequency is fixed and it is F/32 or F/64. Where F is crystal frequency.

c) Mode 3: It is a 9-bit standard UART mode (baud frequency variable). The total 11 bits (1 start bit+TB8/RB8+ 8 bit data (SBUF)+1 stop bit) are transmitted/received in this mode. The baud frequency is variable. The timer 1 mode 2 (8-bit auto reload) is used to generate baud rate.

RS232 standards

A RS232 standard is an interfacing standard developed by the Electronics Industries Association (EIA) for the compatibility among all the serial data communication equipments. Now it is most widely used serial I/O interfacing standard. This standard is used in PCs and numerous types of equipment's. But this standard is not compatible with input and output voltage levels of TTL. The logic levels of the RS232 is as follow,

- ‘1’ is represented by -3 to -25 V,
- ‘0’ bit is +3 to +25 V,
- undefined -3 to +3V

For this reason, we cannot connect any RS232 standard equipment to the microcontroller system and vice versa. It is required to convert voltage levels TTL into RS232 and vice versa. A MAX232, converts the TTL logic levels into the RS232 voltage levels and vice versa. MAX232 chip is commonly referred as line drivers. It converts RS232 into TTL and vice versa.

MAX232

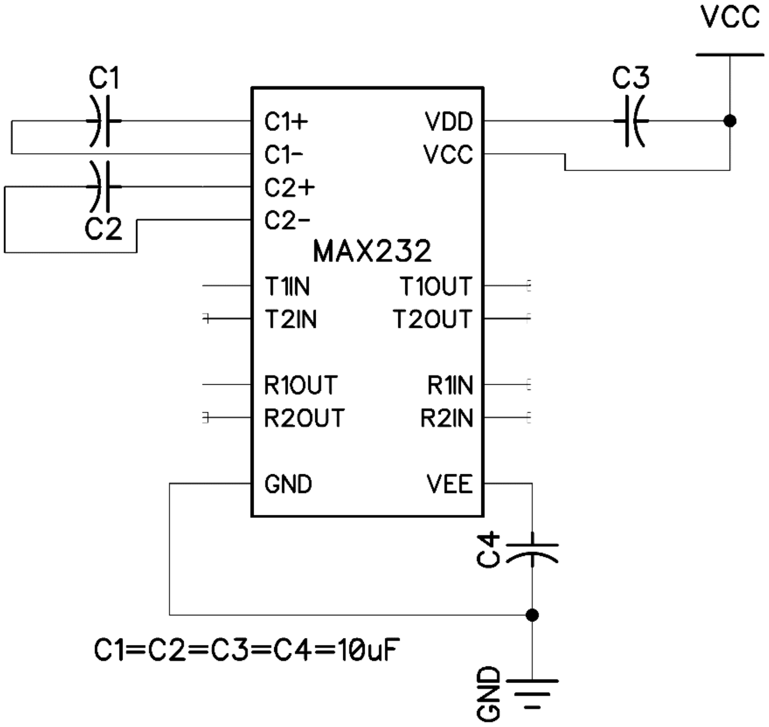


Fig. 3. Pin Connection MAX232

The RS232 standards are not compatible with TTL. So, a line driver required which can convert RS232 voltage levels and vice versa. The MAX232 is a line driver which can converts RS232 to TTL and vice versa. The pin connection of the MAX232 is shown in fig 1.2. It is 16 pin IC. It has two set of line driver for transmitting and receiving. T1 and T2 for the transmission and R1 and R2 for the reception. It converts RS232 into TTL and vice versa. The interfacing of MAX232 with 8051 microcontroller is shown in fig.4.

The serial port TXD pin of the 8051 is connected to the T1IN pin of the MAX232 and the serial port RXD pin of the 8051 is connected to the R1OUT pin of the MAX232. The MAX232 pins T1OUT and R1INis connected to DB9 connector pin 2 and 3 respectively. Through this connector we can connect a serial port of computer. It also required four capacitor C1, C2, C3 and C4 ranging from 1µF to 22 µF. MAX232 required only +5V supply.

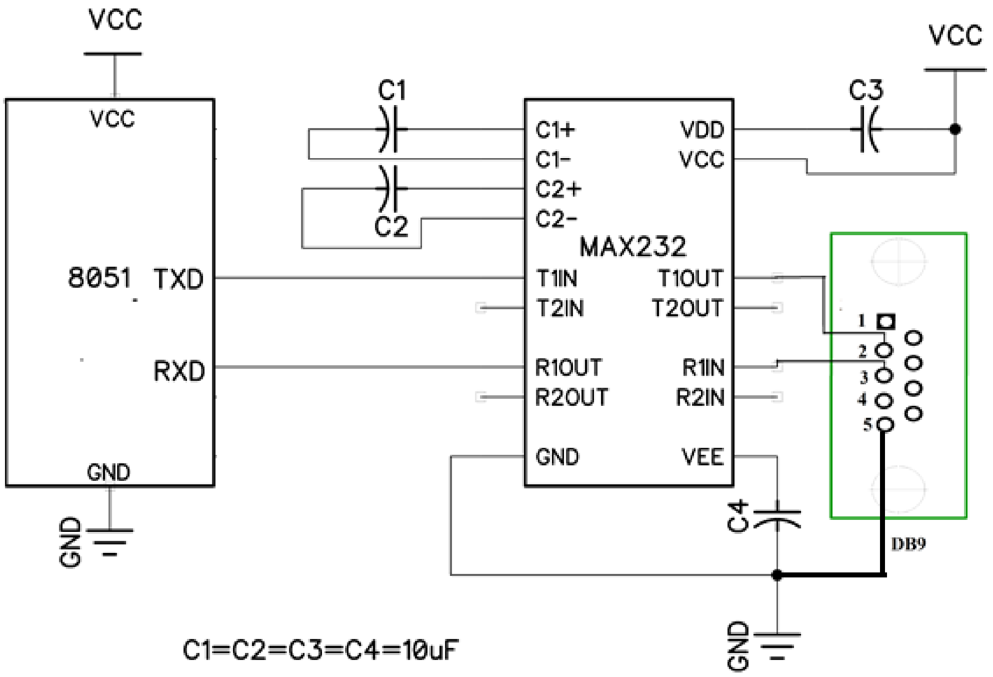


Fig 4: Interfacing MAX232 with 8051

Steps for programming the 8051 to transfer data serially

The following steps must be taken during serial data transmission,

1. Load the TMOD register with the value 20H. It uses Timer1 in mode 2 (8-bit auto-reload) to set the baud rate.
2. Load the TH1 register with the values shown in table 1 to set the baud rate for serial data transmission.

3. Load the SCON register with the value 50H, indicating serial mode 1 (8-bit standard UART mode) and reception is enabled.
4. Start the Timer 1 by setting TR1 bit to ‘1’.
5. Load the SBUF register with character byte to be transferred serially.

1. Monitor the TI flag bit using instruction ***while (TI==0);*** to see transmission is over. [TI flag is set to ‘1’ during the transfer of the stop bit indicating that the last character is transmitted and it is ready to transfer the next character].

6. Clear the TI flag using instruction ***TI=0;*** for next data transmission.
7. Goto step number 5 for next data transmission.

Ex. 1 Write a C program for 8051 to transfer the letter “A” serially at 4800 baud continuously. Use 8-bit data and 1 stop bit.

Solution:

```
#include <reg51.h>
void main(void)
{
TMOD=0x20;      //use Timer 1, mode 2
TH1=0xFA;       //4800 baud rate
SCON=0x50;      // serial mode 1 and reception is enabled
TR1=1;          //start the timer
while (1)       //repeat forever
 {
SBUF=‘A’;       //place value in buffer
while (TI==0);  // monitor the TI flag bit
TI=0;           // clear the TI flag
}
}
```

Ex. 2 Write a program to transfer the message “YES” serially at 9600 baud, 8-bit data, with 1 stop bit.

```
#include <reg51.h>
void SerTx(unsigned char);       //user defined function named as SexTx
void main(void)
{
TMOD=0x20;          //use Timer 1, mode 2
TH1=0xFD;           //9600 baud rate
SCON=0x50;          // serial mode 1 and reception is enabled
TR1=1;              //start timer
while (1)           //repeat forever
{
SerTx(‘Y’);         //call SexTx function
SerTx(‘E’);         //call SexTx function
SerTx(‘S’);         //call SexTx function
}
}
void SerTx(unsigned char x)
{
SBUF=x;             //place value in buffer
while (TI==0);      // monitor the TI flag bit (wait until transmitted)
TI=0;               // clear the TI flag
}
```

Steps for programming the 8051 to Receive data serially

The following steps must be taken during serial data reception,

1. Load the TMOD register with the value 20H. It uses Timer1 in mode 2 (8-bit auto-reload) to set the baud rate.
2. Load the TH1 register with the values shown in table 2 to set the baud rate for serial data transmission.
3. Load the SCON register with the value 50H, indicating serial mode 1 (8-bit standard UART mode) and reception is enable.
4. Start the Timer 1 by setting TR1 bit to ‘1’.

5. Monitor the RI flag bit using instruction ***while (RI==0);*** to check reception is over.
6. If RI arises means SBUF has entire character byte which is received serially. [When stop bit is received, the 8051 makes RI= ‘1’ indicating that entire character or byte is received and it is in SBUF register. User can copy the contents of SBUF register into the safe place like memory or register]. Store the SBUF at safe place.
7. Clear the RI flag using instruction ***RI=0;*** for next data reception.
8. Goto step number 5 for next data transmission.

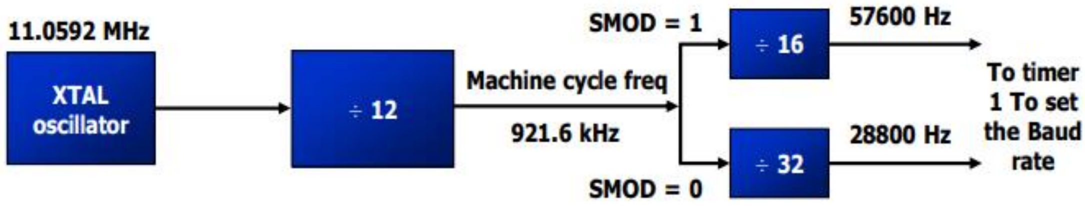
Ex. 3 Write a program to receive serially at 4800 baud, 8-bit data, with 1 stop bit. After receiving send it continuously to P0.

```
#include <reg51.h>
void main(void)
{
unsigned char mybyte;
TMOD=0x20;     //use Timer 1, mode 2
TH1=0xFA;      //4800 baud rate
SCON=0x50;     // serial mode 1 and reception is enabled
TR1=1;         //start timer
while (1)      //repeat forever
{
while (RI==0); //wait to receive
mybyte=SBUF;   //save value
P1=mybyte;     //send value to port
RI=0;          // clear the RI flag
}
}
```

Doubling the baud rate using 8051

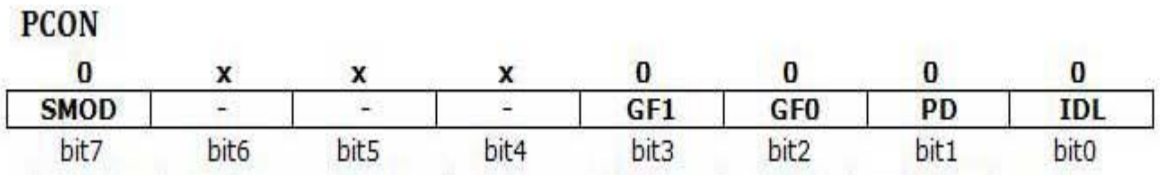
There are two methods are used to increase the baud rate of serial communication.

1. First is by increasing frequency of the crystal but is not feasible because IBM PC baud rate can't exactly achieved.(e.g. 12 MHz/12=1 MHz and 1MHz/32=31250 Hz no correct value for baud rate 9800)
2. Another way to increase baud rate is use of PCON register.



PCON Register

It is an 8 bit special function register. It is byte addressable register and useful to doubling the baud rate as well power control in CHMOS controller. Fig 1.5 shows the internal structure of the PCON register.



PCON

0	x	x	x	0	0	0	0
SMOD	-	-	-	GF1	GF0	PD	IDL
bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0

Fig.5 PCON register

SMOD

It is D7 bit of PCON register. As soon power is ON D7 bit is zero and we use usual baud rate set by Timer 1 mode 2. But if we make D7 bit=1 by software then baud rate of 8051 doubles as shown in Table.

Crystal frequency	Machine cycle frequency	SMOD	Serial communication frequency	If Value of TH1=-6 then baud rate=
11.0592MHz	11.0592/12=921.6Khz	=0	921.6/32=28800Hz	28800/6=4800
11.0592MHz	11.0592/12=921.6Khz	=1	921.6/16=57600Hz	57600/6=9800

Table 3: Doubling the Baud rate

If we set D7 =1, the internal circuitry for serial communication divides machine cycle frequency by 16 instead of 32. Hence baud rate will doubles for same value of TH1.

Following program for doubling baud rate for same crystal frequency.

```
MOV A,PCON      ;Copy of PCON into accumulator
SETB ACC.7      ;Make D7 bit of Accu.high
MOV  PCON,A     ;Make D7 of PCON =1
```

GF0 and GF1 are the general flag 0 and 1.

PD is a power down bit and **IDL** is an ideal mode bit. Both the bits are used in CHMOS processor only.