

8051 Uart/Serial Communication Tutorial

In 8051, we make use of Timer 1 to generate the required baud rate. Following are the registers that are need to be configured to communicate over UART.

- TMOD
- SCON
- TH1
- TL1
- TCON

TMOD: This register is used to set the mode of Timer0 and Timer1. It is also used to select whether the timers are used as Timer or Counter.

SCON: Serial Control register has various functions like.. it has flags for Framing error, Transmit interrupt and receive interrupt. Its used to select the serial port mode, to enable or disable the reception etc.

TCON: This register has various flag and control bits e.g. Timer overflow flags, interrupt edge flags, timer control bits to start/stop the timer.

TH1 & TL1: Timer registers for Timer 1 determines the baudrate of UART.

More information on the above registers can be found in the 8051 Hardware manual.

Initializing UART in 8051

In Assembly:

```
1. Serial_Init:
2.           ;Set timer 1 mode to 8-bit Auto-Reload
3.           mov TMOD,#20H
4.           ;Enable reception
5.           ;Set Serial port mode to 8-bit UART
6.           mov SCON,#50H
7.           ;Set baudrate to 9600 at 11.0592MHz
8.           mov TH1,#0FDH
9.           mov TL1,#0FDH
10.          ;Start Timer
11.          setb TR1
12.          ret
```

in C we can do this as...

```

1. #include <reg51.h>.
2. void serial_init(){
3.     TMOD = 0x20;
4.     SCON = 0x50;
5.     TH1  = 0xFD;
6.     TL1  = 0xFD;
7.     TR1  = 1;
8. }

```

To Send data to the serial port we just have to move the data in SBUF (serial buffer register) and wait for the Transmit Interrupt flag to be set. While receiving we wait for the Receive interrupt flag to be set and read the data from SBUF register. This can be done as shown below...

```

1. Serial_Send:
2.     ;wait for last data to be
3.     ;sent completely
4.     jnb TI,Serial_Send
5.     ;clear the transmit interrupt flag
6.     clr TI
7.     ;Then move the data to send in SBUF
8.     mov SBUF,A
9.     ret
10. Serial_Read:
11.     ;Wait for Receive interrupt flag
12.     jnb RI,Serial_Read
13.     ;If flag is set then clear it
14.     clr RI
15.     ;Then read data from SBUF
16.     mov A,SBUF
17.     ret

```

```

1. void serial_send(unsigned char dat){
2.     while(!TI);
3.     TI = 0;
4.     SBUF = dat;
5. }
6. unsigned char serial_read(){
7.     while(!RI);
8.     RI = 0;
9.     return SBUF;
10. }

```

