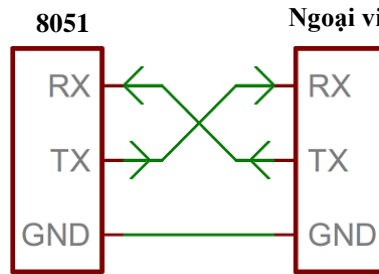


## SERIAL



### 1. Giới thiệu

- Serial là chuyển truyền nối tiếp, sử dụng 3 chân: Tx(truyền), Rx(nhận), GND(đồng bộ GND cho bên truyền và bên nhận)
  - o Chân Tx của VDK được nối với chân Rx của Ngoại vi. Và ngược lại.
- Gói tin truyền cơ bản, gồm: bit Start, 8 bit data, bit Stop
  - o Tức mỗi lần truyền/nhận được 1 byte (8 bit).

### 2. Tốc độ truyền (baudrate)

- Serial có 4 mode, chúng ta sẽ sử dụng mode 1.
- Trong mode 1, Serial sẽ sử dụng Timer 1 để tạo xung truyền/nhận dữ liệu.  
(Nên, Timer 1 sẽ không dùng cho mục đích khác)
- Tốc độ truyền Serial (baudrate) được cấu hình dựa trên giá trị nạp vào TH1 của Timer1, bạn xem bảng sau:

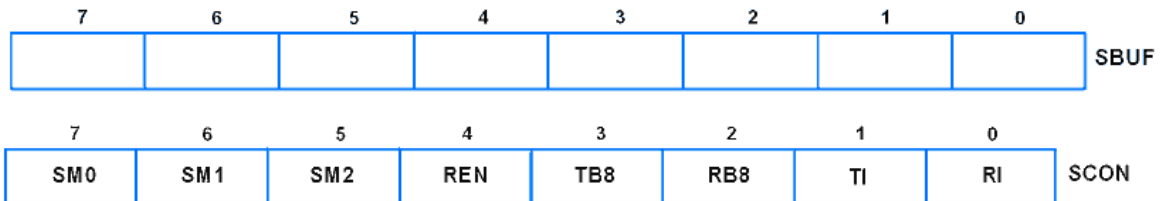
Baud Rate	TH1 (Hex)
9600	FD
4800	FA
2400	F4
1200	E8

- (\*)Lưu ý: Khi dùng Serial, bạn phải sử dụng thạch anh **11.0592MHz**

### 3. Hàm gửi/nhận Serial

- Để gửi 1 byte dữ liệu, bạn chỉ cần đặt dữ liệu vào thanh ghi SBUF (có sẵn)
- Để nhận 1 byte dữ liệu,
  - o Bạn không thể biết khi nào thì dữ liệu đến, vì vậy bạn cần viết hàm Ngắt nhận cho Serial (khi có dữ liệu đến – thì hàm Ngắt được gọi)
  - o Để đọc dữ liệu, bạn tiến hành đọc từ thanh ghi SBUF và lưu vào biến bạn cần.

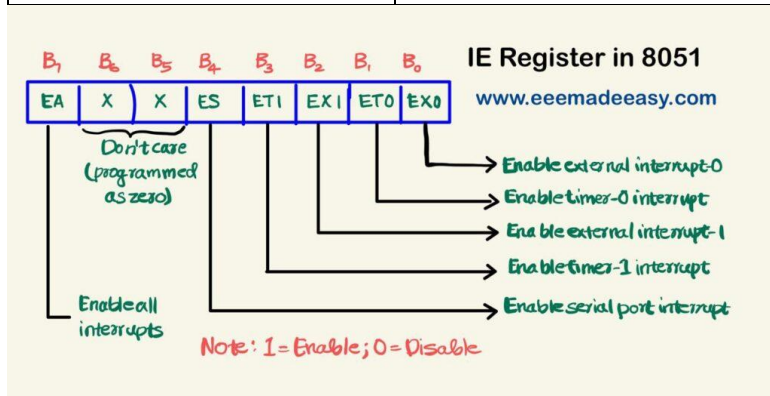
### 4. Các thanh ghi liên quan Serial



Bit 7:6 - SM0:SM1: Serial Mode Specifier

Mode	SM0	SM1	Mode
0	0	0	1/12 of Osc frequency shift register mode fixed baud rate
1	0	1	8-bit UART with timer 1 determined baud rate
2	1	0	9-bit UART with 1/32 of Osc fixed baud rate
3	1	1	9-bit UART with timer 1 determined baud rate

Bit	Ý nghĩa
REN=1	Cho phép nhận
TI	=0: mặc định =1: khi truyền xong
RI	=0: mặc định =1: khi có dữ liệu mới
SM2=0	Hiện tại không quan tâm
TB8, RB8	Hiện tại không quan tâm

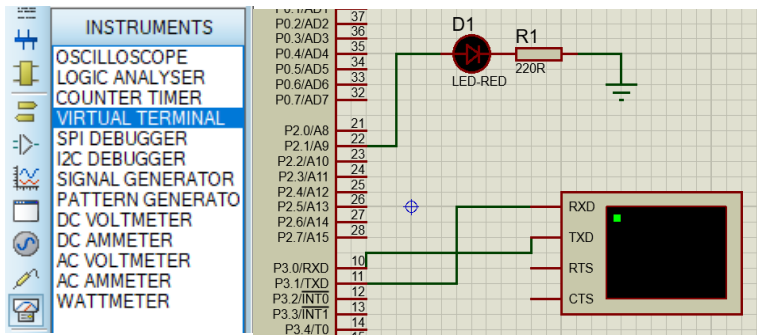


## 5. Chương trình mẫu

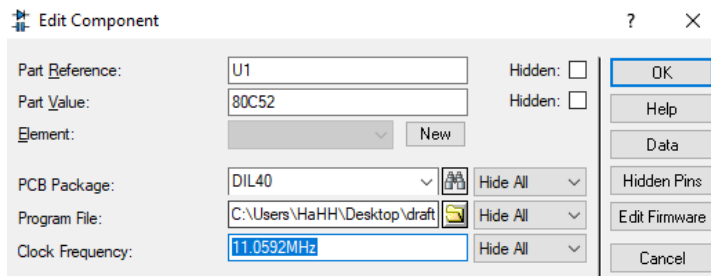
```
1  #include <REGX52.H>
2
3  sbit LED = P2^1;
4  unsigned char flag = 0;
5  unsigned char revData;
6
7  void serial_init(void)
8  {
9      /* fcrystal=11.0592MHz */
10     /*Config Timer1 (Don't use Timer1's Interrupt*/
11     TMOD = 0x20;    // Timer 1, 8bit, auto-reload
12     TH1 = 0xFD;    // 9600 baud rate
13     SCON = 0x50;    // Serial: Use Timer1, Receive Enable
14     TR1 = 1;        // Start Timer 1
15
16     EA = 1;         /* Enable global interrupt */
17     ES = 1;         /* Enable serial interrupt */
18 }
19
20 void serial_Send(unsigned char x)
21 {
22     SBUF = x;        //put a character in buffer, the system automatically sends it.
23     while(TI==0);    //wait until transmitting completely. (TI=1: Finish sending)
24     TI = 0;          //Need to clear TI Flag by software.
25 }
26
27 void serial_ISR(void) interrupt 4
28 {
29     if (RI==1) {     //Interrupt when receiving data
30         revData = SBUF; /* received data saves to revData */
31         RI = 0;       /* Need to clear RI flag */
32
33         flag=1;
34     }
35     //if (TI==1) ...only use when need interrupt for sending
36 }
37
38 void main(void)
39 {
40     LED = 0;         //turn off led
41     serial_init();    //config Serial
42     while(1)
43     {
44         if(flag==1)
45         {
46             if(revData=='1')
47                 LED = 1;
48             else
49                 LED = 0;
50
51             serial_Send('0');
52             serial_Send('K');
53
54             flag=0;
55         }
56         //Do some other codes....
57     }
58 }
```

## 6. Cách chạy chương trình mẫu với Proteus

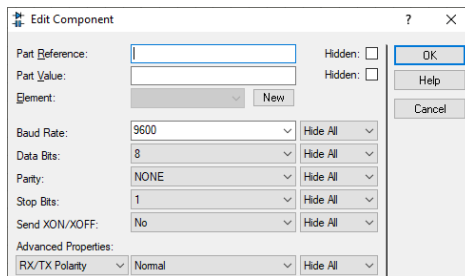
- Sơ đồ kết nối (Sử dụng Terminal làm ngoại vi giao tiếp VDK)



- Cấu hình tần số thạch anh là 11.0592MHz

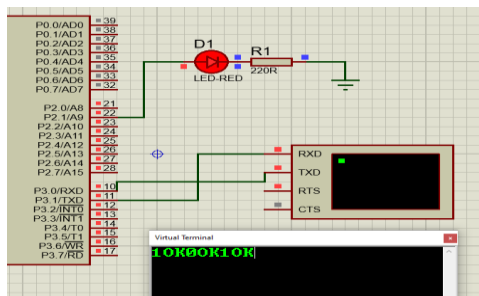


- Cấu hình tốc độ baudrate cho Terminal



- Cách chạy:

- Khi chạy, Terminal hiện cửa sổ màu đen, Right-click vào cửa sổ đen chọn “echo...” (Khi không thấy tắt cửa sổ đen, thì bấm chạy rồi vào Debug sẽ thấy)
- Rồi gõ 1 ký tự từ bàn phím rồi xem kết quả.



## 7. Câu hỏi

1. Gọi lên bảng, chỉ 1 câu lệnh trong chương trình mẫu và yêu cầu giải thích.

## 8. Bài tập áp dụng

1. Chạy lại chương trình mẫu.
  2. Viết chương trình gửi mã điều khiển 3 led từ Terminal:
    - a. '1' : Bật led 1
    - b. '2' : Tắt led 1
    - c. '3' : Bật led 2
    - d. '4' : Tắt led 2
    - e. "5x" : Nhấp nháy led 3, với x là số lần nhấp nháy  $T = 500\text{ms}$  ( $x = 1..9$ )
- Yêu cầu: 8051 gửi lại lên Terminal: "Led1 on" khi nhận '1'. Hay "Led3 blink" khi nhận "5x".
  - Chú ý: SBUF là thanh ghi 8 bit. Mỗi lần chỉ nhận được 1 byte/kí tự.  
(Tức là: Mỗi lần có 1 kí tự đến, thì CT nhảy vào hàm interrupt 1 lần)
  - Nên (không bắt buộc): Viết thêm hàm SendString() để VĐK gửi được 1 chuỗi dữ liệu.
  - Có thể: Tận dụng hàm sprintf() (#include <stdio.h>) (nếu thấy phù hợp)