**Serial Communication between 8051 microcontroller and**

**Peripheral device**.

Device Name:

1. Device: at89c51
2. Instruments > Virtual Terminal

Keil

1. Code:

#include <reg51.h>

void main()

{

SCON=0x50;

TMOD=0x20;

TH1=-3;

TR1=1;

SBUF='B';

while(TI==0);

TI=0;

}

1. Options for Target:

A screenshot of a computer

AI-generated content may be incorrect.

This code is written in C for the 8051 microcontroller, using its built-in registers and features. Let's go through the code step by step to explain each part:

**1. #include <reg51.h>**

This line includes the header file for the 8051 microcontroller. The reg51.h file defines the special function registers (SFRs) for the 8051 architecture. These registers are used to interact with hardware peripherals like timers, serial communication, and others.

**2. void main()**

This is the main function where the execution of the program begins.

**3. SCON=0x50;**

This sets the Serial Control register (SCON). The 8051 microcontroller has a serial communication feature, and SCON is used to control the serial port. The value 0x50 sets the register as follows:

* **Bit 7 (SM0)**: 0 (Serial mode)
* **Bit 6 (SM1)**: 1 (Serial mode)
* **Bit 5 (SM2)**: 0 (Multiprocessor communication mode)
* **Bit 4 (REN)**: 1 (Receiver enable, allowing the serial port to receive data)
* **Bit 3-0 (TB8, RB8, TI, RI)**: These bits are used for data transmission and reception; TI is the Transmit Interrupt flag, and RI is the Receive Interrupt flag. Here, these bits are not directly set by 0x50.

**4. TMOD=0x20;**

This sets the Timer Mode register (TMOD) to 0x20. The TMOD register controls the modes of the two timers in the 8051 microcontroller. Here, the value 0x20 configures Timer 1 in mode 2 (auto-reload mode), where the timer will automatically reload its value when it overflows. This is typically used to generate a baud rate for serial communication.

**5. TH1=-3;**

This line sets the high byte of Timer 1 to -3. In the 8051, Timer 1 operates in auto-reload mode, and TH1 (the high byte of Timer 1) is used to set the timer's reload value. The value -3 (which is equivalent to 0xFD in hexadecimal) is used to generate a specific baud rate for serial communication. In combination with TL1 (the low byte of Timer 1, which is implicitly set), it determines the baud rate.

**6. TR1=1;**

This starts Timer 1 by setting the TR1 bit to 1. Timer 1 will now begin counting, and it will overflow when the timer reaches the value set in TH1 and TL1.

**7. SBUF='A';**

This sends the character 'A' (ASCII code 65) through the serial port. The SBUF register is the serial buffer register. When you write data to SBUF, it is placed in the transmit buffer and is then sent serially.

**8. while(TI==0);**

This line waits until the transmission is complete. The TI (Transmit Interrupt) flag is set by hardware when the serial port has finished transmitting the byte in SBUF. This line makes the program wait until TI is set, which indicates that the character has been transmitted.

**9. TI=0;**

This clears the Transmit Interrupt flag (TI), resetting it for the next transmission.