

Bit	Bit Name	Description
SCON.7	SM0	$0 \Rightarrow$ 8-bit data; $1 \Rightarrow$ 9-bit data
SCON.6	SM1	$0 \Rightarrow \left(\text{Baud-rate} = \frac{F_{clk}}{12} \text{ for SM0}=0; \frac{2^{SMOD} \cdot F_{clk}}{64} \text{ for SM0}=1 \right);$ $1 \Rightarrow \text{Baud-rate} = \frac{2^{SMOD} \cdot F_{clk}}{12 \times 32 \times (\text{No. of Timer1 machine cycles})}$
SCON.5	SM2	For multi-processor communication (If SM2=1, hardware asserts RI only if RB8=1, when serial data is received)
SCON.4	REN	Receive enable
SCON.3	TB8	9th transmit data bit in 9-bit UART mode
SCON.2	RB8	9th received data bit in 9-bit UART mode
SCON.1	TI	Transmit interrupt flag
SCON.0	RI	Receive interrupt flag
PCON.7	SMOD	Double baud-rate for $\{SM0, SM1\} = \{0,1\}, \{1,0\}$ or $\{1,1\}$

1. In the example discussed in class for multi-processor communication, the P3.0 (RxD) and P3.1 (TxD) pins of 4 microcontrollers A, B, C and D are all connected together to enable serial communication between any set of processors. There is a bug in the programs on two of the processors such that they start transmitting simultaneously. Will this bug result in a short circuit (high current leading to potential damage to the chip)? Give the reason(s). [1]

Solution

There will be no short circuit current. Logic ‘1’ is provided through a weak pull-up resistor (i.e. a large resistor). Therefore, even if one Tx is sending a ‘0’ and the other one sending a ‘1’, only a small current can flow through the large resistor, and resultant logic state is ‘0’.

2. Assume that you’re not allowed to use the serial interface in the 8051 microcontroller. However, you still use the microcontroller to implement a UART (using some other port pins). [2]
- (a) Will you be able to implement the UART and support all the data rates that are otherwise supported by a dedicated 8051 serial interface, assuming that no other processes are running on the 8051 chip (if not, which rates cannot be supported)?
- (b) Will there be a synchronization problem (that cannot be resolved) in the supported baud rates of the Tx or the Rx or both (explain briefly)?

Solution

- (a) One cannot support data rates in which bit period is not integer multiple of machine cycle periods. That is, $\text{Baud-rate} = \frac{2^{SMOD} \cdot F_{clk}}{64}$ can’t be supported.
Also, $\frac{F_{clk}}{12}$ can’t be supported (because it requires the program to update bits every machine cycle, which is not possible as there are always some overheads. However this is not a “UART” mode, hence even if you didn’t mention it, you don’t lose points.

- (b) All synchronization problems in the micro-controller can be resolved as there are multiple machine cycle periods in one bit period of the UART for the supported baud-rates – minimum number of machine cycles is 16, which gives flexibility in the program for synchronization.
3. An 8051 has been configured to transmit and receive data in an 8-bit UART mode. Write the sequence of steps that must be taken in the serial interface ISR to ensure that the serial interface and the microcontroller work properly (optional steps that may vary from program to program need not be mentioned). [2]

Solution

- (a) Check if TI is 1 or RI is 1, based on it jump to Tx or Rx instructions.
- (b) Rx instructions:
- i. Copy SBUF to a register.
 - ii. Anything else if required.
 - iii. Clear RI flag.
 - iv. Return from ISR (RETI).
- (c) Tx instructions:
- i. Clear TI flag.
 - ii. Copy next byte to be transmitted to SBUF (if another byte available for transmission).
 - iii. Anything else if required.
 - iv. Return from ISR (RETI).