

Embedded C Programming
Laboratory 9
Programs on Serial Interrupts

Task 1:

Write a C program using serial interrupts to do the following:

- (a) Use the timer 1 8-bit auto reload mode and set the baud rate at 9600 for serially communication.**
- (b) Configure the SCON register for serial communication.**
- (c) Start the timer 1.**
- (d) Enable the IE register for serial interrupt.**

Ask the program to give an acknowledgment 'ACK' serially whenever the user presses an enter key and toggle the pins at port 0 from 00H to FFH. (HINT: Value for ENTER key in decimal is 13)

Solution:

(a) (i) TMOD – Timer 0 8-bit auto reload: 0000 0010

Therefore, TMOD = 0100 0010 = 0x20;

(ii) TH = 0xFD

(b) SCON = 0101 0000 = 0x50

(c) TR1 = 1

(d) IE = 1001 0000 = 0x90

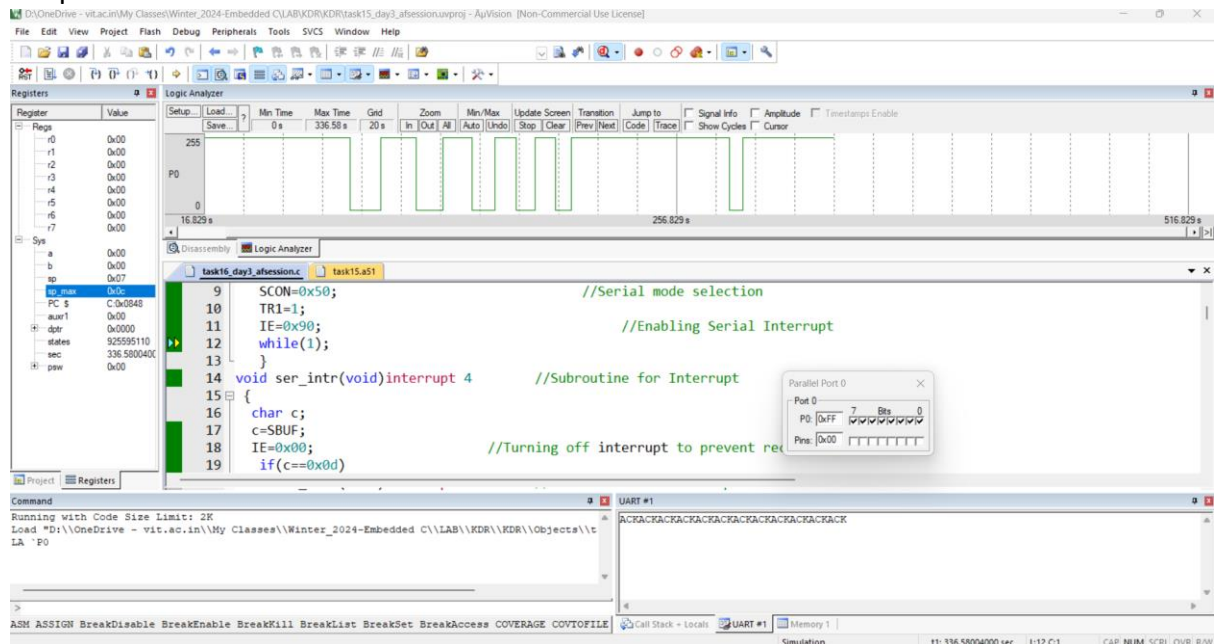
```
#include <reg51.h>
void main()
{
    TMOD=0x20;           //Choosing Timer mode
    TH1=0xFD;            //Selecting Baud Rate
    SCON=0x50;           //Serial mode selection
    TR1=1;
    IE=0x90;             //Enabling Serial Interrupt
    while(1);
}
void ser_intr(void)interrupt 4    //Subroutine for Interrupt
{
    char c;
    c=SBUF;
    IE=0x00;             //Turning off interrupt temporarily to prevent recursion
    if(c==0x0d)
    {
```

```

P0=~P0;
SBUF='A';           //Sending back "ACK" as Acknowledgement
while(TI==0); //waits until TI is set to indicate 'A' has been transmitted
TI=0;
SBUF='C';
while(TI==0);
TI=0;
SBUF='K';
while(TI==0);
TI=0;
}
RI=0;
IE=0x90;           //Reactivating the interrupt
}

```

Output:



Task 2:

Write a C program using serial interrupts to do the following:

- Use the timer 0 8-bit auto reload mode and set the baud rate at 4800 for serially communication.
- Configure the SCON register for serial communication.
- Start the timer 1.
- Enable the IE register for serial interrupt.

Ask the program to give an acknowledgment 'RECEIVED' serially whenever the user presses the space bar key and toggle the pins at port 2 from 00H to FFH. (HINT: Value for SPACEBAR key is 0x20)

```
#include <reg51.h>
```

```

void main()
{

    TMOD=0x20;           //Choosing Timer mode
    TH1=0xFD;           //Selecting Baud Rate
    SCON=0x50;           //Serial mode selection
    TR1=1;
    IE=0x90;             //Enabling Serial Interrupt
    while(1);
}

void ser_intr(void) interrupt 4    //Subroutine for Interrupt
{
    char c;
    c=SBUF;
    IE=0x00;             //Turning off interrupt to prevent recursion
    if(c==0x20)
    {
        P2=~P2;
        SBUF='R';        //Sending back "ACK" as Acknowledgement
        while(TI==0);
        TI=0;
        SBUF='E';
        while(TI==0);
        TI=0;
        SBUF='C';
        while(TI==0);
        TI=0;
        SBUF='E';        //Sending back "ACK" as Acknowledgement
        while(TI==0);
        TI=0;
        SBUF='I';
        while(TI==0);
        TI=0;
        SBUF='V';
        while(TI==0);
        TI=0;
        SBUF='E';
        while(TI==0);
        TI=0;
        SBUF='D';
        while(TI==0);
        TI=0;
        SBUF=' ';
        while(TI==0);
    }
}

```

Output



(a) Receive data serially and send it to P0

(c) Make timer 0 generate a square wave of 5 kHz frequency on P0.1

(Hint: Use timer 1 to serially transmit, use timer 0 to generate a square wave)

(i) **TMOD – Timer 0 mode 2 ; Timer 1 mode 2**

(ii) TH1 = 0xF6 - 4800 baud rate

(iv) $TH0 = ?$

$\frac{1}{2}$ of the time for the high and low pulse = 100 μ s

Total clock to generate delay of $100\mu\text{s} = 100/1.085 = 92$

Therefore, TH0 = 0xA4

(v) IE : Timer 1, so, 1001 0010

Therefore, IE = 0x92;

(vi) TR1 = 1

(vii) TR0 = 0;

```
#include <reg51.h>
sbit WAVE =P0^1;
void timer0(void) interrupt 1
{
WAVE=~WAVE; //toggle pin
}
void ser_intr(void) interrupt 4 //Subroutine for Interrupt
{
if(TI==1)//If TI flag is set, make it as '0'
{
TI=0;
}
else
{
char c;
c=SBUF;
IE=0x00; //Turning off interrupt to prevent recursion
P0=SBUF;

while(RI==0);
RI=0;
}
IE=0x90; //Reactivating the interrupt
}
void main()
{
unsigned char x;

P1=0xF6;
TH1=0xFA;
TH0=0xA4;
TMOD=0x22;
SCON=0x50;
TR0=1;
TR1=1;
IE=0x92; //enable interrupt for timer 0
while (1)
{
```

```

x=P1;
SBUF = x;
P2=x;
}
}

```

Output:

