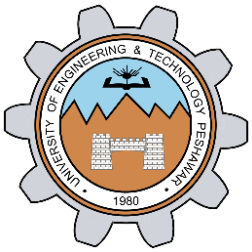
MICROPROCESSOR BASED SYSTEM DESIGN

Final term



Spring 2021

CSE307 MBSD

Submitted by: **Shah Raza**

Registration No. : **18PWCSE1658**

Class Section: **B**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: _____

Submitted to:

Dr. Bilal Habib

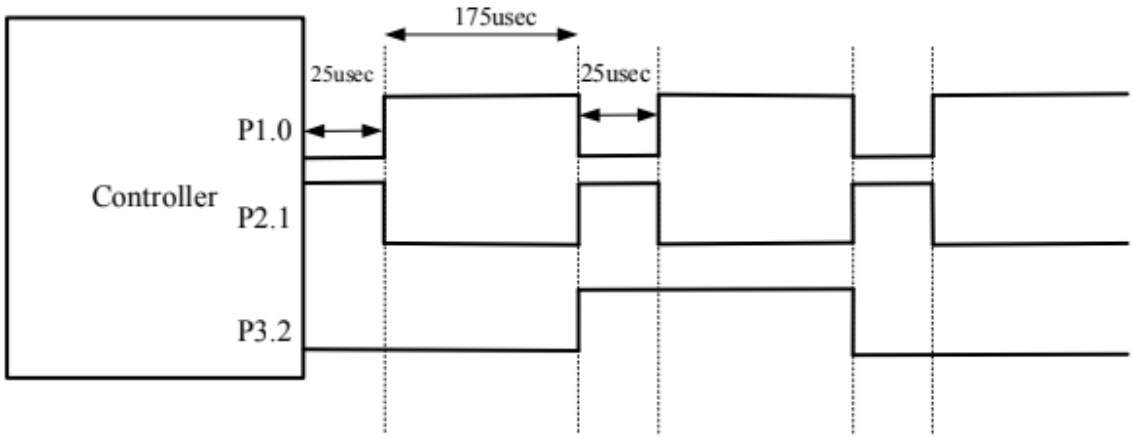
Friday, July 30, 2021

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Q1:

Analyze the timing diagram below and write the code for it to generate these periodic signals. Use timer interrupts for this purpose.

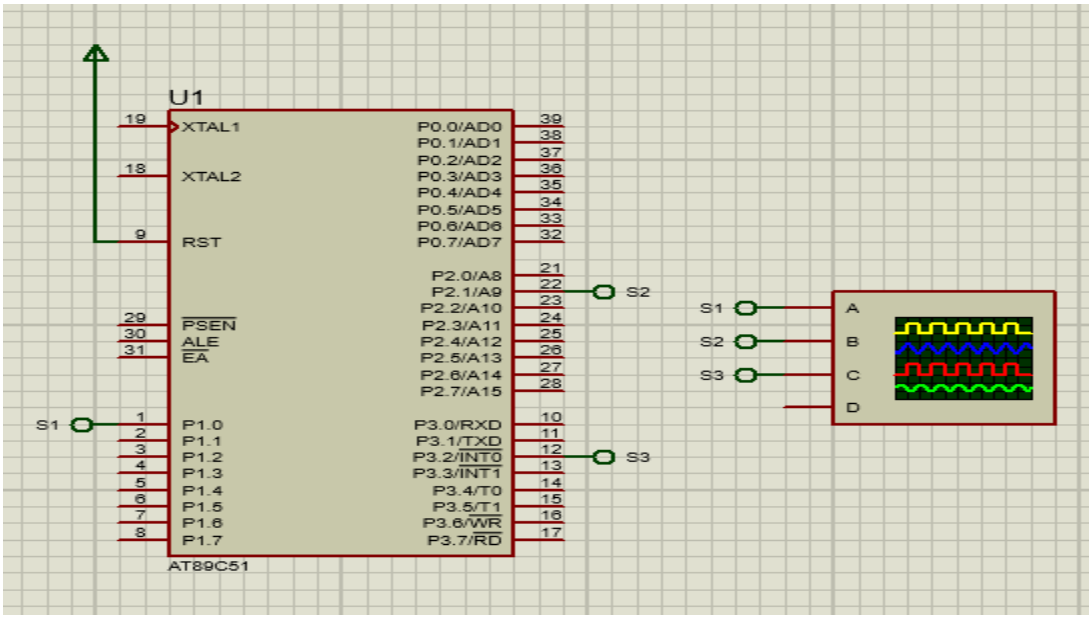


Code:

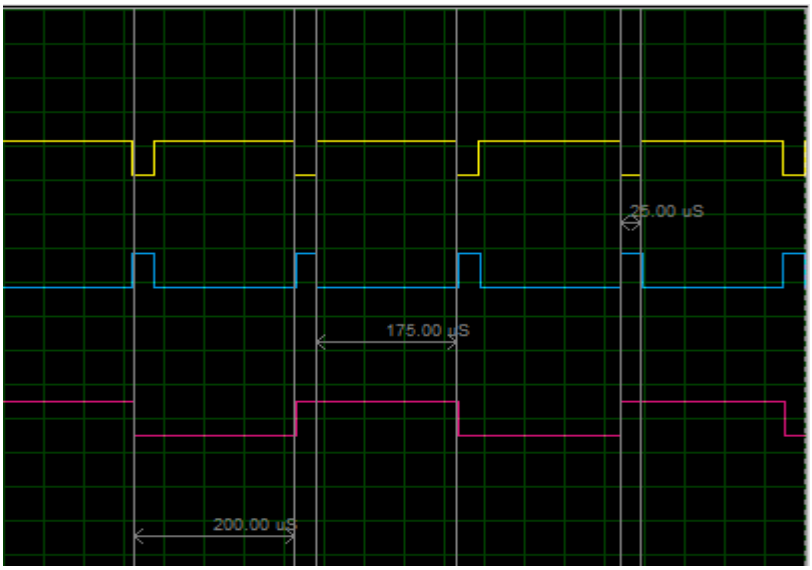
```
#include <reg51.h>
#include <stdio.h>

//Signals wil be generated at these 3 pins
sbit Signal1 = P1^0;
sbit Signal2 = P2^1;
sbit Signal3 = P3^2;

int x = 0;          //This variable is used for generating different duty cycle signals
int i =0;

void timer0() interrupt 1 //called each time the timer0 overflow bit is set
{
  if(Signal1==0)
  {
    //Toggle the first 2 signals
    Signal1 = ~Signal1;
    Signal2 = ~Signal2;
  }
  else
  {
    x++;
    if(x==6)
    {
        i++;     //Just a dummy variable increment to create 2 usec extra delay
        //Toggle all the signals
```

```
            Signal1 = ~Signal1;
            Signal2 = ~Signal2;
            Signal3 = ~Signal3;
            x =0;
     }
   }
   //Timer 0 delay
   TH0 = 0xFF;
   TL0 = 0xF8;

}

void StartTimer()
{
   TR0 =1;          //Start timer 0
}

void Init()
{
   TMOD = 0x1;   //Timer 0 mode 1
   EA = 1;          //Enable global interrupt
   ET0 =1;          //Enable Timer0 interrupt
   /*We want  a delay of 25usec but we have used if coditions which also takes a lot of time
   So to get a fix delay I generated less delay than 25usec*/
   TH0 = 0xFF;
   TL0 = 0xF8;

}

void main(void)
 {
    //Initial values for all the signals
   Signal1 = 0;
    Signal2= 1;
    Signal3 = 0;
    Init(); //Initialize Timer0;
    StartTimer();
   while (1)
      ;
 }
```

Schematic:



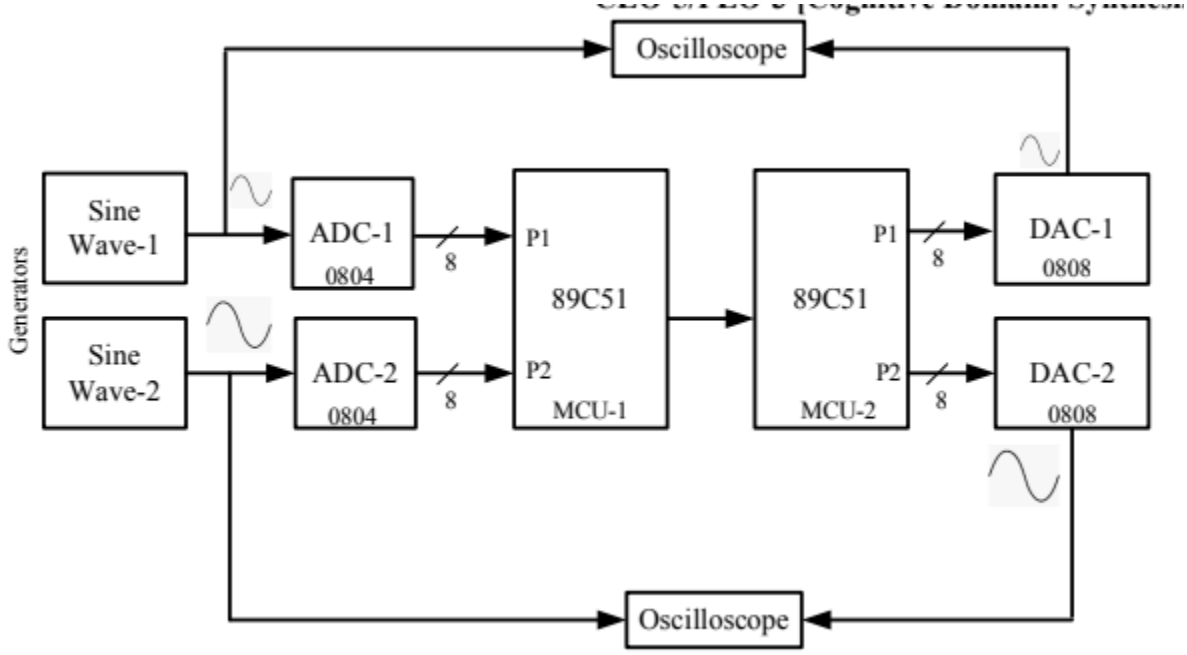
Output:



Can you generate all three signals using a single timer?

Answer: Yes, all three signals can be generated using one timer. I have done this task using only one timer. The base delay is 25usec and 175 and 200 are multiples of 25. So we can easily generate these signals using 1 timer.

Q2:



Implement the project as shown in the Figure above.
The oscillator frequency of both microcontrollers is fixed at 22.118MHz.

Code:

Micro-Controller 1:

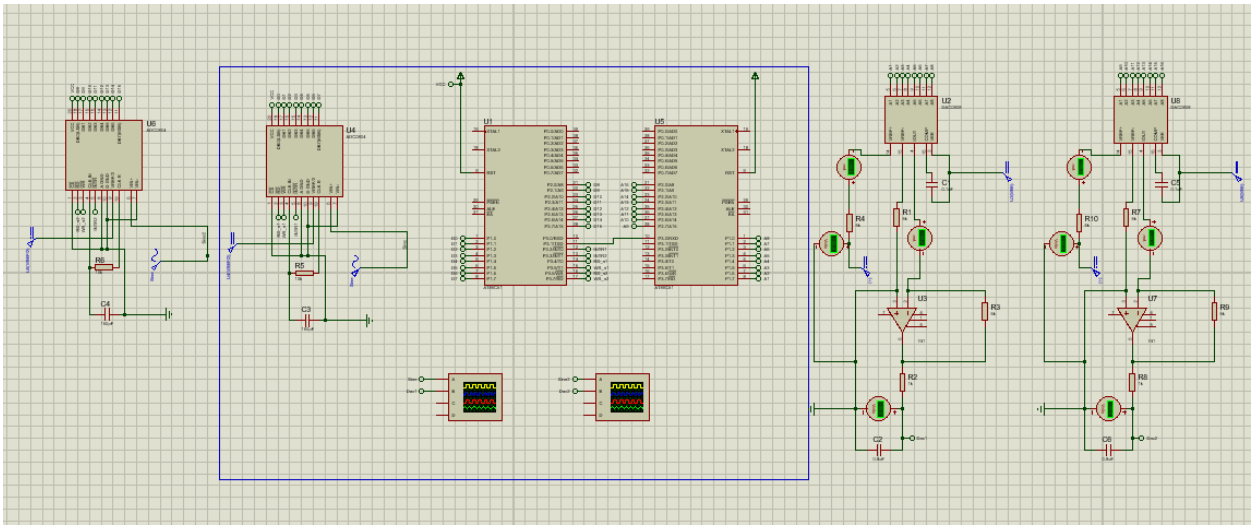
```
#include <reg51.h>
#include <stdio.h>

sbit RD_n1 = P3^4; //P3.4 is connected to the RD pin of ADC1
sbit WR_n1 = P3^5; //P3.5 is connected to the WR pin of ADC1
sbit INTR1 = P3^2; //P3.2 is connected to the INTR pin of ADC1

sbit RD_n2 = P3^6; //P3.6 is connected to the RD pin of ADC2
sbit WR_n2 = P3^7; //P3.7 is connected to the WR pin of ADC2
sbit INTR2 = P3^3; //P3.3 is connected to the INTR pin of ADC2

int x =0;

void main(void)
 {
   P1 = 0xFF; //Set P1 as an input Port
   P2 = 0xFF; //Set P2 as an input Port
   INTR1 = 1; //Set P3.2 as an input pin
   INTR2 = 1; //Set P3.3 as an input pin
   TMOD = 0x20;            //Timer 1 mode 2
```

```
  TH1 = 0xFF;    //28800*2*2  = 115200 bps
  SCON = 0x40; //Mode 1 serial communication
  PCON = 0x80; //SMOD = 1
  TR1= 1;                    //Start timer 1

 while (1)
 {
    RD_n1 = 1; //Set the RD pin to High
   RD_n2 = 1; //Set the RD pin to High
   WR_n1 = 0;//WR = Low
    WR_n2 = 0;//WR = Low
   WR_n1 = 1;//Low-->High
   WR_n2 = 1;//Low-->High

   if(x%2==0)
   {
          while(INTR1==1); //Wait for the ADC to Convert the given voltage
         RD_n1 = 0; //Set the RD pin of ADC from HIGH to LOW
   //The ADC sends the converted value to P1
   SBUF = P1; //Send the value at P1 to SBUF
   while(TI==0);          //While the SBUF is not transmitted, do nothing
   TI = 0;                    //Reset the TI bit to 0
   }
   else
   {
         while(INTR2==1); //Wait for the ADC to Convert the given voltage
   RD_n2 = 0; //Set the RD pin of ADC from HIGH to LOW
   //The ADC sends the converted value to P2
   SBUF = P2; //Send the value at P2 to SBUF
   while(TI==0);          //While the SBUF is not transmitted, do nothing
   TI = 0;                    //Reset the TI bit to 0
 }
  }
}
```

Micro-Controller 2:

```
#include <reg51.h>
#include <stdio.h>
int x=0;

void main(void)
 {
   P1 = 0x00; //Set P1 as an Output Port
   P2 = 0x00; //Set P2 as an Output Port
   TMOD = 0x20;            //Timer 1 mode 2
   TH1 = 0xFF;    //28800*2*2  = 115200 bps
   SCON = 0x50; //Mode 1 serial communication with REN bit set to 1
   PCON = 0x80; //SMOD = 1
   TR1= 1;                 //Start timer 1

  while (1)
  {
    while(RI == 0);         //While the value is not recieved, do nothing
    RI = 0;                 //Reset the RI bit to 0
    if(x%2==0)
          P1 = SBUF;        //Send the value recieved at SBUF to P1
    else
          P2 = SBUF;        //Send the value recieved at SBUF to P2
  }
 }
```

Schematic:



Keeping in view the fastest possible transmission rate of serial comm and ADC conversion rate. How much the frequency of input signals can be increased?

Answer: The maximum frequency can be 11520Hz because then we will need 115200 samples for it which is also the maximum transmission rate of Microcontroller.

Answer the following questions, in a word file.

a. **What will be transmission rate of MCU-1 and MCU-2 in bits per second.**
   Answer: It is 115200 bps.
b. What sampling rate will you choose for ADC-1 and ADC-2.
   Answer: ADC-1 sampling rate 1k and ADC2 sampling rate 2K.

c. If you choose X Hz asthe fastest possible frequency of input sine waves. Which component

will be the problematic at (X+1) Hz. Will it be ADC, DAC, Serial Comm or something

else? Prove mathematically.

Answer: Dac will be problematic as its conversion time is a bit slower.

d. Can we sample both ADCs with the same sampling rate? Under what conditions we can

do that?

Answer: No because the input frequencies are different. It can be same if input frequencies are same.