EE 354 **Fall 2015**
Real Time Clock Example

The program on the following page illustrates how to implement a real time clock to keep track of tenths of a second, seconds, and minutes. It can be easily extended to hours, days, weeks, etc.

There are four global variables named milliseconds, tenthsCounter, seconds, and minutes. These global variables are updated by the interrupt routine running every millisecond. The main program changes the bits on P1 when the tenths, seconds, or minutes change. This program assumes an AT89C51CC03 with a 28.2076MHz crystal running in the x2 mode.

```
//RTCExample.c
// This program uses Timer 0 to produce a 1 msec interrupt
//   It uses this time to alter the bits on P1.0, P1.1, and P1.2
//   to mark tenths of a second, seconds, and minutes.
// Bit 3.2 indicates milliseconds.
#include<REG51AC2.h>

unsigned char seconds;
unsigned char minutes;
unsigned char tenthsCounter;
unsigned char milliSeconds;

void main(void)
   {unsigned char tOld, sOld, mOld;  //previous values of tenths, seconds, and minutes
    CKCON = 0x01;   // x2 mode
    TMOD = 0x01;    //Timer 0 mode = not gated, internal clock, 16-bit mode
    //For fosc = 28.2076MHz in x2 mode timer is clocked at 28.2076Mhz/6 = 4.701MHz
    // so period is 1/4.701Mhz = .2127086 usec.   To get 1 msec we need 1000/.2127086
    // = 4701 counts.   65536 - 4701 = 60835 = 0xEDA3.
    TH0 = 0xED;     //Timer 0 set to EDA3 - 60835
    TL0 = 0xA3;
    TR0 = 1;
    ET0 = 1;        //Timer 0 interrupt enable
    EA = 1;         //Global interrupt enable
    tOld = 0;sOld = 0;mOld = 0;
    while(1)
      {if(tenthsCounter != tOld) //Complement bit when counter changes
          {P1_0 = ~P1_0;
           tOld = tenthsCounter;
          }
       if(seconds != sOld)
         {P1_1 = ~P1_1;
          sOld = seconds;
         }
       if(minutes != mOld)
          {P1_2 = ~P1_2;
           mOld = minutes;
          }
      }
   }
//
void T0Int() interrupt 1 using 1
  {P3_2 = ~P3_2;                //Complement P3.2
   TH0 = 0xED;                  //Timer 0 set to EDA3 - 60835
   TL0 = 0xA3;
   milliSeconds++;
   if(milliSeconds > 99)        //.1 sec = 100 msec
     {milliSeconds = 0;
      tenthsCounter++;
      if(tenthsCounter > 9)     //For each 10 update seconds
         {tenthsCounter = 0;
          seconds++;
          if(seconds > 59)      //For each 60 seconds update minutes
             {seconds = 0;
              minutes++;        //minutes can overflow
             }
         }
      }
   }
```