# Laboratory 06
# INT for Time and Date

CO 2103 Assembly Language

# Objective

AL programming using software interrupt

-INT 21h instructions for system date/time

-Time delay subroutine

# System Date

- Two ways to obtain today's date:
  - mov ah,2Ah
    int 21h
    - CX contains the current Year
    - DH contains the Month number , where January=1
    - DL contains the Day of the month
  - mov ah,04h
    int 1Ah
    - CX contains the current Year
    - DH contains the Month number, where January=1
    - DL contains the Day of the month
      - Numbers in BCD
- There are more ways (INT) …

# System Time

- Many ways (INT) to get current time, the following is probably easiest (and useful):
  - mov ah,2Ch
    int 21h
    - Retrieves DOS maintained clock time
    - CH contains the current hour (0-23)
    - CL contains the current minutes (0-59)
    - DH contains the current seconds (0-59)
    - DL contains the current hundredth seconds (0-99)

# Exercises

- **Task 9:** Write a program to display today's date: save as pdate.asm
- **Task 10:** Why are there two ways? What's the difference between the two?
- **Task 11**: Write a program to display current clock: save as ptime.asm
  - Note that date and time are DOS commands and we should not create programs of same name

# Time Delay Subroutine

- One of the most commonly used subroutine
- A time delay subroutine is basically asking the processor to wait (and do nothing) for a specific amount of time
- Possible implementations:
  - NOP (No Operation, i.e. do nothing) loop
  - Check timer difference: loop or interlaced
  - Set timer interrupt
- This lab will cover first two implementations

# Time Delay: NOP Loop

- Provided we know the number of clock cycles for each instruction and the clock speed of the CPU
  - If clock frequency is f, each clock cycle is T=1/f
  - There 4 instructions in the subroutine, assuming the first instruction takes cc1 clock cycles, while the following instructions take cc2, cc3 and cc4 respectively (to obtain from CPU document)
- Time Delay = [cc1 + n(cc2+cc3) + cc4]T

```
Tdelay proc near
       mov CX,n        ;set delay time
                       ;(can be passed in as parameter)
Kwait:    nop             ;do nothing (can be omitted)
       loop Kwait      ;for CX times
       ret
Tdelay endp
```

# Time Delay: Timer

- Reading the timer (or current time) and wait until the required time delay elapsed
  - Can also reset the timer and wait until the required time delay elapsed

```
Pseudocode:
Delay:
    Stime = current time;
    Etime = current time + n;
    If (current time = Etime) then
       return;
    else
       Goto Delay;
```

# Time Delay: Exercises

- **Task 12**: Write a program to print “Start” on the screen, and print “Stop” on next line after 5 seconds using the following implementations:
  - NOP Loop - save this file as tdelay1.asm
  - Timer – save this file as tdelay2.asm