EE2401 微算機系統 Fall 2018

HW#2 (8051 Timer, Serial port, Interrupt) (10/9/2018)

Due date: 11/1/2018. Severe penalty will be given to late homework.

Note:

1. The homework will be graded based on your **documentation** and **demonstration**.
2. For all **(Software Design)** problems, you are required to us **MCU8051IDE**simulators to simulate and verify your programs.
3. You are required to **type** your homework (first the problem then your solution) by using a **word processor** and submit in .doc(or .docx) format under a filename **EE2401f18-hw2-student\_no-vn.doc(or .docx)**, where **student\_no** is your student number, e.g., **106061xxx**(or **X107xxxx**) and **vn** is your version number, e.g., **v3**. You should **upload your .doc file** in **iLMS** by the specified deadline whenever you have a newer version. Follow the iLMS upload homework process to upload your file.
4. The homework will be graded based on your **latest version**. Old version(s) will be discarded.
5. Each homework assignment will have full score of 100 points. 5 points will be deducted if you do not comply with the naming convention. Severe grade penalty will be given to late homework. 5 points will be taken off per day after deadline.
6. Please treat the above requirements as a kind of training in writing a decent homework report. If you have any problem regarding this homework, please feel free to consult with TA or me. If you think the time is too short to accomplish this homework, please let me know in class.

1. (Software design) (20%)

In some applications, BCD (binary coded decimal) is used instead of binary. Each BCD will use 4 bits to represent. A two digits BCD is usually placed in a single byte. For example, a two digits decimal number 59 would be represented by the byte “0101 1001”, where the upper nibble(半字節) (b7~b4) 0101 stands for “5” and lower nibble (b3~b0) 1001 stands for “9”.

Please write an 8051 assembly program that will add two 6-digit BCD numbers stored in internal data memory location say 30h-32h and 33h-35h and produce a 7 digits BCD result in 36h-39h. You should show at least two results of the BCD addition operation in MCU8051IDE (by print screen).

2. (Software design, simulation and instruction cycle counting) (20%)

8051 is commonly used as the core of a programmable logic controller (PLC) to perform logical manipulations on various incoming signals and issue proper controls accordingly, because it has Boolean instructions, internal bit-addressable bits for storing Boolean variables, and 4 8-bit I/O ports. The basic idea to implement a logic operation with 8051 is to enclose the logic operations in an infinite loop so that it can react to change of input persistently like hardware does. Assume that we want to implement the following logic operations:

P1.0 =(P1.1•P1.2+!P1.3) ⊕ (!P1.4•P1.5+P1.3) if P1.6 = 0 and

(P1.1+!P1.2•P1.3) ⊕ (!P1.4+!P1.5•P1.3) if P1.6 = 1.

(Note: ! means NOT, ⊕ means exclusive or)

1. First construct the truth-table. Apply K-map to simplify the logic if you like.
2. Then draw a flow chart to describe your code for the logic operation.
3. Write an 8051 assembly program to implement the above logic operations. You should **verify your program in MCU8051IDE** by trying several input combinations according to the truth-table.
4. Estimate the best case and worst case propagation delay of your program from an input transition to output transition under 12 MHz operation:

3. (Data sheet study and software design/simulation in MCU8051IDE) (20%)

A 4-bit DIP switch and a **common-anode** 7-segment LED are connected to an 8051 as shown below. The 4-bit DIP switch is connected to P3.4-P3.7. When the DIP is in up position, the corresponding port pin would read in a “1”, and a “0” for down position. To turn on an LED segment, the corresponding control signal input of the 7-segment, i.e., a~g and dp, should be logic zero. Which means the corresponding port pin should be at logic “1” since a 74LS240 is used in the middle.

1. What is common-anode 7-segment LED? Why do we need the 220 resistor and the 74LS240? What’s the function and purpose of 74LS240 in this design? You should search the web to find out relevant data sheets of 74LS240 and common-anode 7-segment LED, read them, and then try to answer the question.
2. Design an 8051 program which can continuously read the DIP position and display the 7-segment LED correspondingly. Test your program **in MCU8051IDE** and use the **simple keypad** and **LED display** virtual HW to demonstrate your program. You should capture at least two examples of the MCU8051IDE simulation results by print screen and put them in your written report.

dp

P1.0

P1.1

P1.2

P1.3

P1.4

P1.5

P1.6

P1.7

P3.4

P3.5

P3.6

P3.7

74LS240

8051

a

a

b

b

c

c

d

d

e

e

f

f

g

g

dp

8x220

7-segment

LED

+5V

DIP

4. (Software Design) (20%)

Square wave/pulse generation

1. Design an 8051 program that can create a square wave on P1.0 with a frequency of 100 kHz, 10 kHz, 1 kHz, and 100 Hz, assuming a 12 MHz 8051 is used. Analyze the accuracy of your design for these 4 square waves.
2. Design an 8051 program to generate a 5 kHz 60/40 duty cycle pulse wave on P1.4. (60/40 means 60% high and 40% low in one cycle)

5. (Software Design) (20%)

Music playing

1. Refer to the C example in the class for playing Do, Re, Mi, Fa, So, La, Ti, Do-H, redesign the program in 8051 assembly language.
2. Modify the program in (a) so that the time duration for each note can be varying. (You might need to use interrupt in your design if you like. Please study the class material and technical references posted in the website)