**CECS 262**

**Lab 2: Viewing Registers and Memory with a Simulator**

**Last, First Name Khan, Umar**

**Last 4 Student ID: 7331.**

**Objectives:**

* To get familiar with the software development tool Keil uVision, learn how to use Keil to edit, compile
* To examine the PSW register bits by continuing to use MOV and ADD instructions
* To exam the stack.
* **Tasks A:**

1. Follow the tutorial, Keil tutorial.pdf posted on Beachboard Labs folder; finish all the steps up to simulation. When you create your source file, use “My Little Chasing Cat” shown below. Run the program on the simulation and *demonstrate* it to the instructor.

Example: My Little Chasing Cat.

==========================================================

; This program is called "My Little Chasing Cat".

; 1. Type in the program, compile and simulate it.

; 2. Observe what happens on port P1

; "My Little Chasing Cat".

ORG 0000H

MOV A, #01H

LOOP: MOV P1, A

RL A

MOV R3, #0FFH

DLY0: MOV R1, #0FFH

DLY1: MOV R2, #0FFH

DLY2: DJNZ R2, DLY2

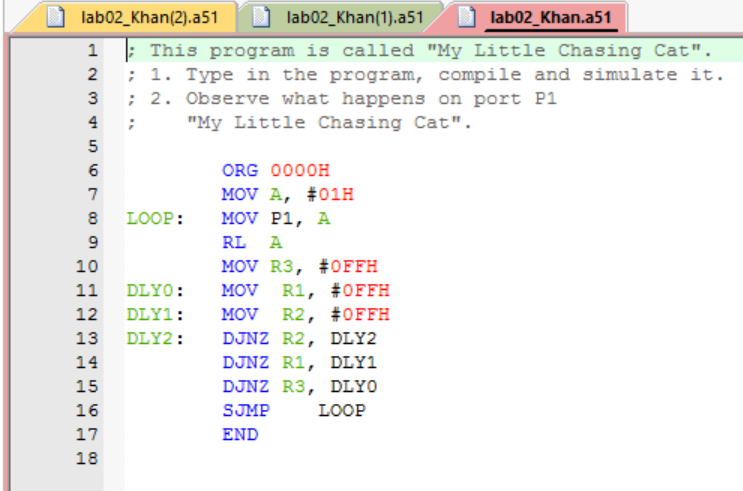
DJNZ R1, DLY1

DJNZ R3, DLY0

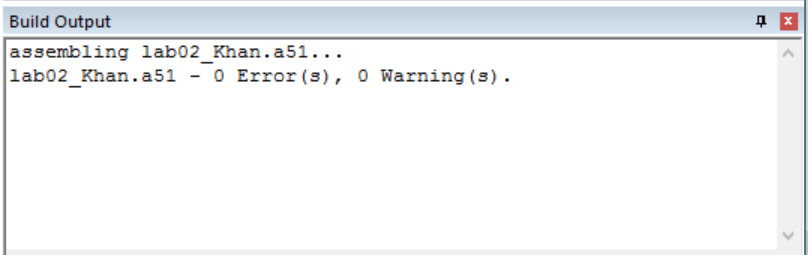
SJMP LOOP

END

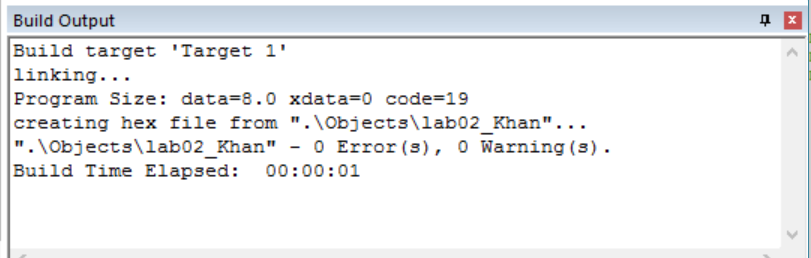
* 1. Source code (here is the sample code)

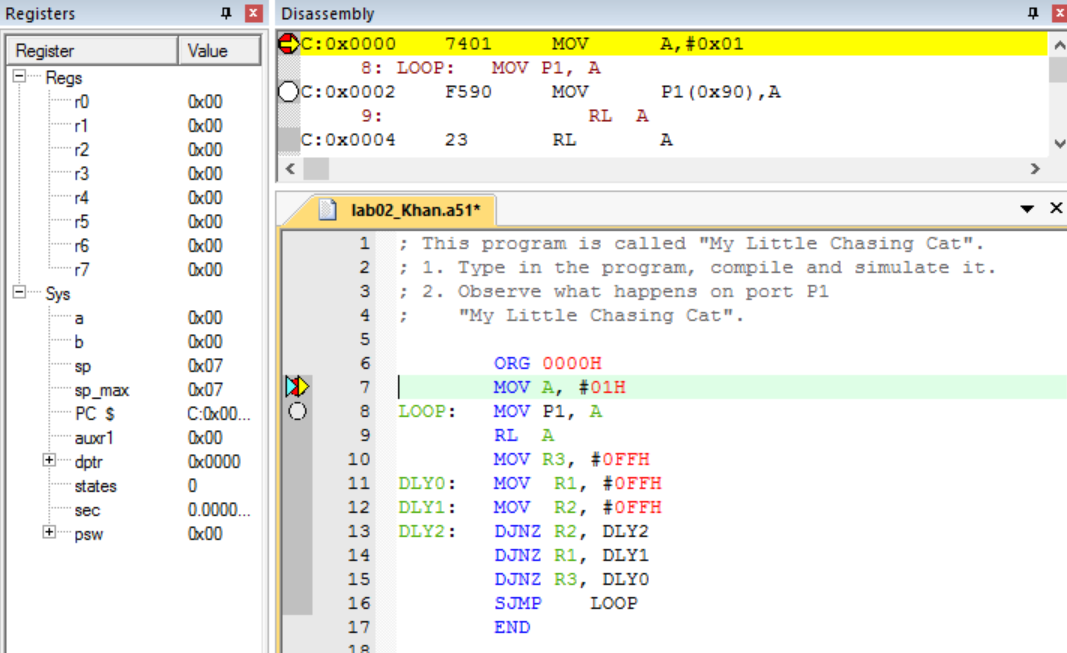
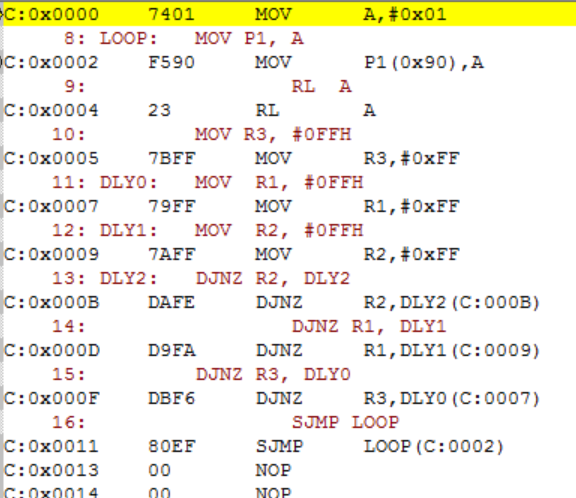


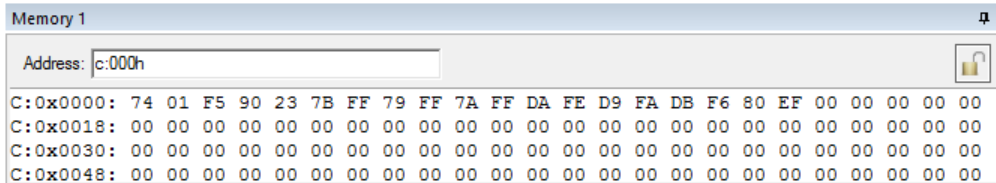
* 1. Build output after the Translate



* 1. Build output after the Build



* 1. Register Window and Disassembly Window after the Debug   
       
     
  2. Memory Window



==========================================================

* **Task B:**

Write and assemble a program to add the following data and save the final result to register R1. Use the simulator to examine the CY/AC/P flag. Screen-shot whenever the addition generates a carry.

92H, E3H, 66H, 87H, F5H

ORG 0000H

Xxxxx

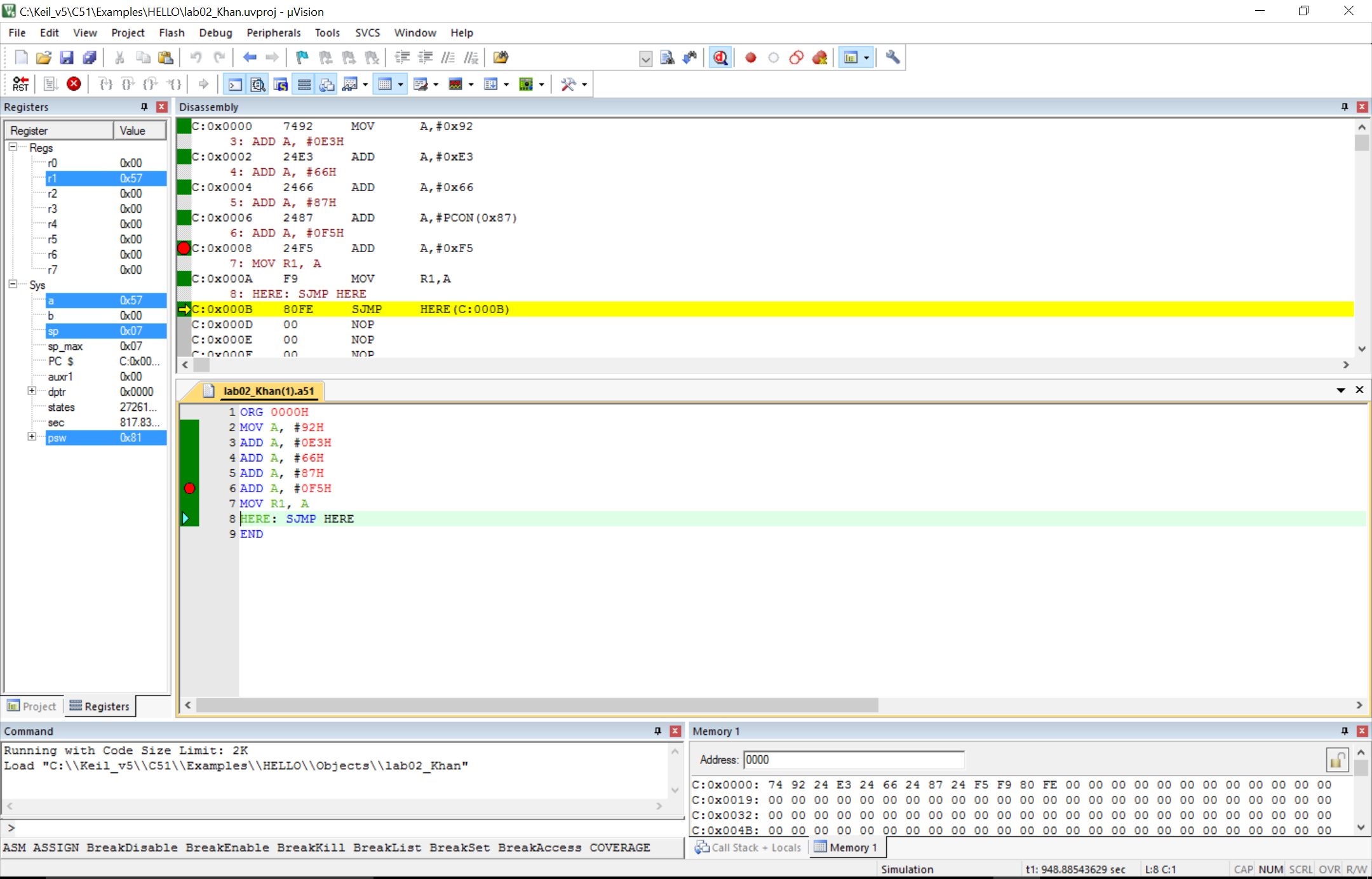
xxxxx

HERE: SJMP HERE

END

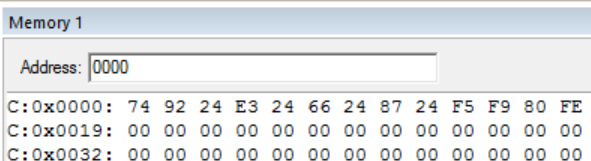
You need to:

1. Show the values for CY/AC/P after detailed manual addition for each step
2. Include related screenshot of the register window for each step and compare with your manual calculation



For example, here presents the register window the executing ADD A, #0F5H.

1. Show the memory window with all involved machine code



* **Task C**

Write and assemble a program to:

* MOV the following 6 different values to internal RAM locations: 31H - 36H,
* then PUSH the contents in those memory locations to the stack, assuming the initial value of SP is 7. When finish the above operation, screenshot the contents in the stack and contents in RAM location 31H – 36H. Data to be moved to 31H – 36H: A3H, 24H, 38H, 7CH, D3H, 79H
* POP the top 6 items from the stack into registers R0 – R5.
* Use the simulator to single-step and examine the registers, the stack, and the stack register. After poping all 6 items, screen-shot the contents in registers R0 – R5, the stack, and SP register.

ORG 0000H

MOV 31H,#0A3H

xxx

PUSH 31H

xxx

POP 0

xxx

HERE: SJMP HERE

END

