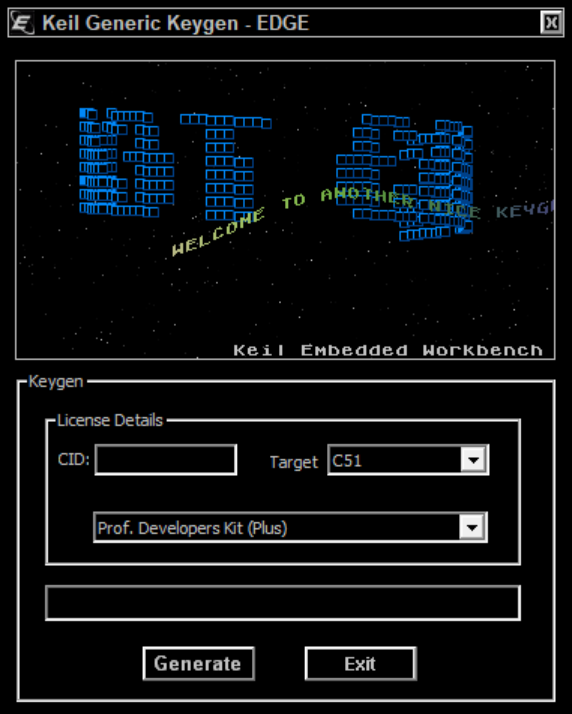
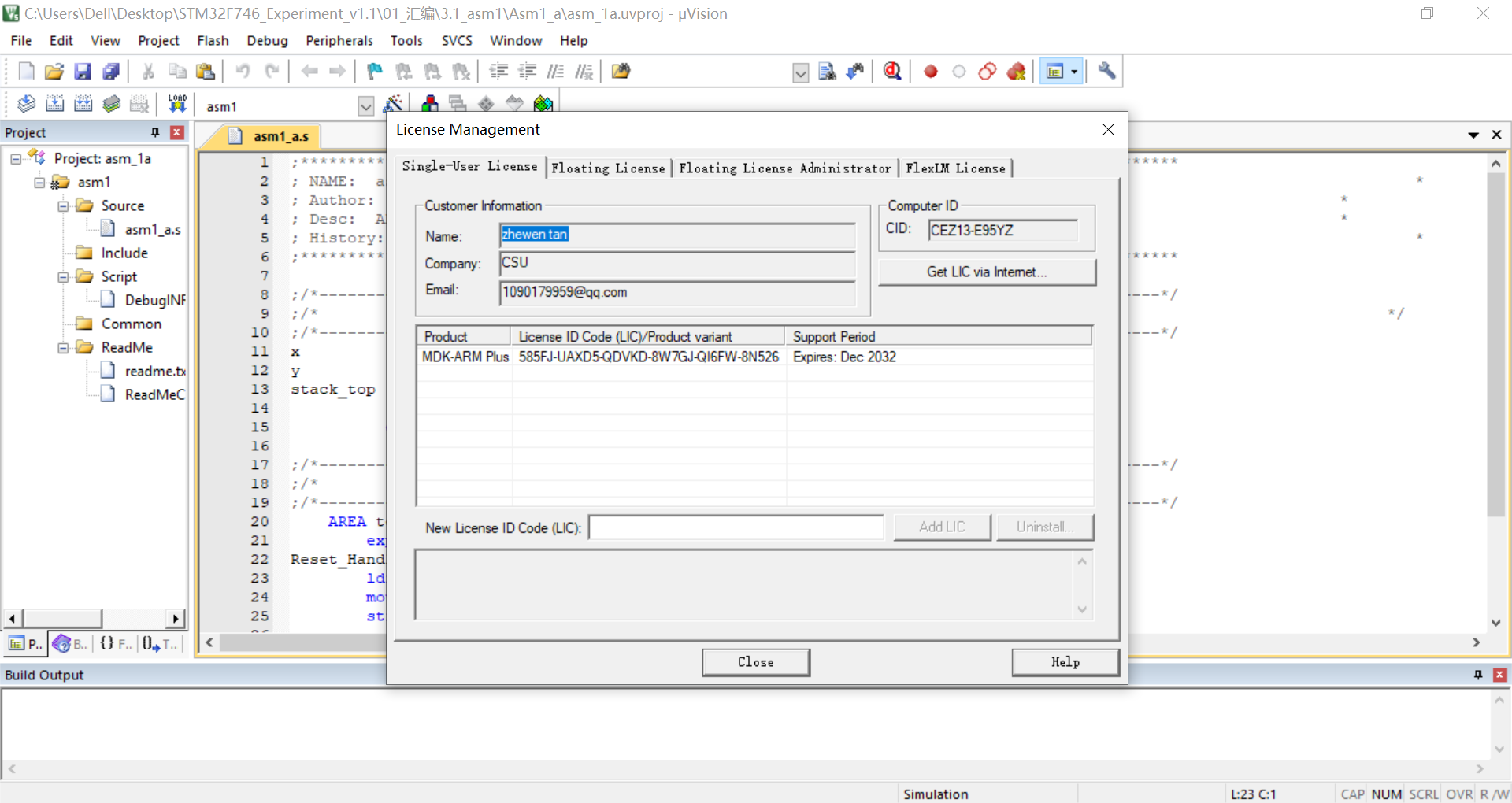
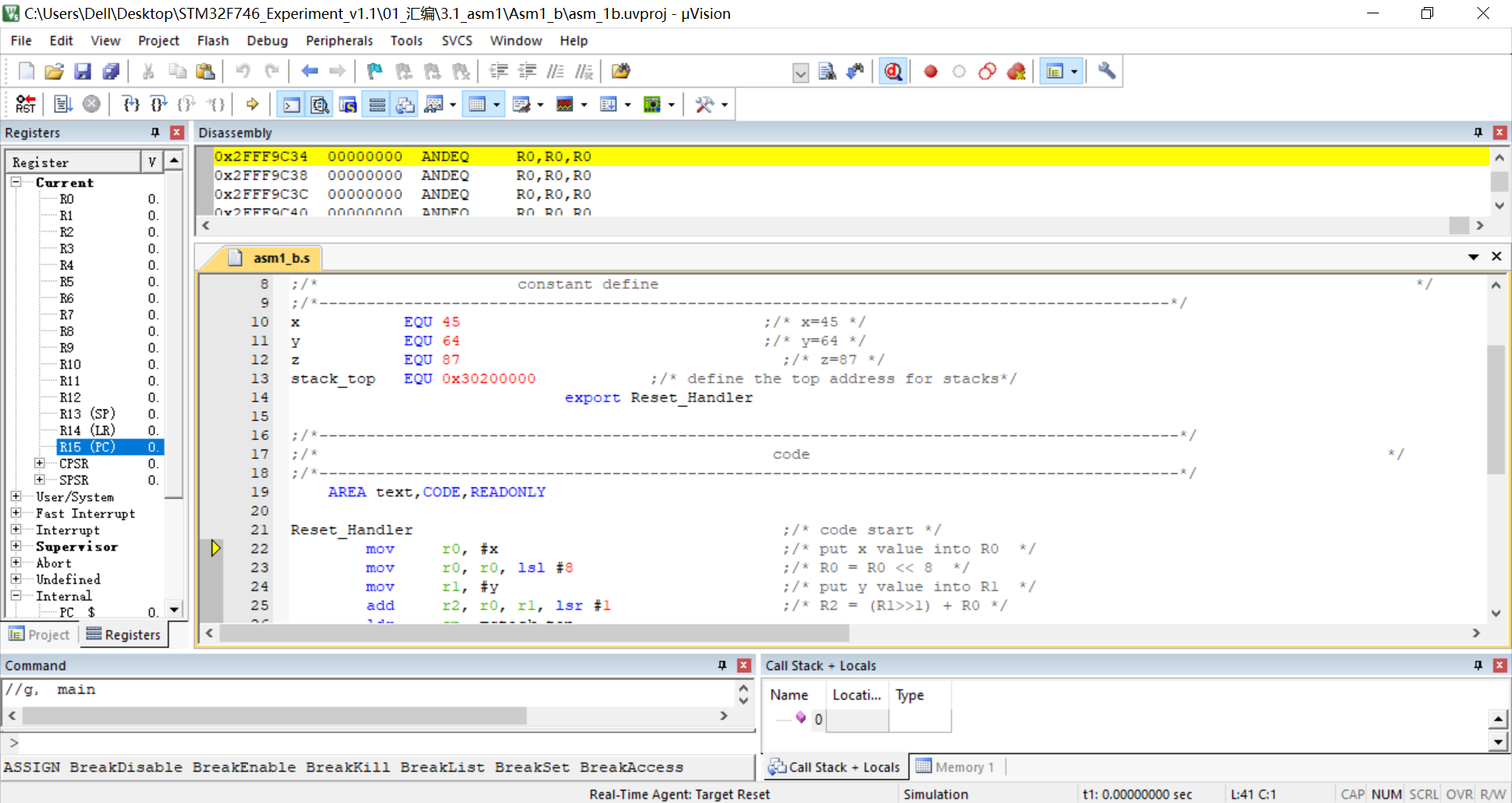
**任务一：安装ARM编译环境**





**任务二：运行STM32F746\_Experiment\_v1.1\01\_汇编\3.1\_asm1\Asm1\_b例程，进入debug模式。**

1. **阅读并调试代码，观察每行代码的地址位置、代码指令编码、代码反编译结果、寄存器更新值。**

**在提交作业中，把所有代码复制一遍，在注释中给出每行代码运行后寄存器（或存储器）的值更新情况。**

1. x   EQU 45             ;*/\* x=45 \*/*
2. y   EQU 64             ;*/\* y=64 \*/*
3. z   EQU 87            ;*/\* z=87 \*/*
4. stack\_top EQU 0x30200000     ;*/\* define the top address for stacks\*/*
5. export Reset\_Handler
6. ;*/\*-------------------------------------------------------------------------------------------\*/*
7. ;*/\*                                  code                                            \*/*
8. ;*/\*-------------------------------------------------------------------------------------------\*/*
9. AREA text,CODE,READONLY
10. Reset\_Handler          ;*/\* code start \*/*
11. mov  r0, #x         ;*/\* put x value into R0£¬=2D  \*/*
12. mov  r0, r0, lsl #8      ;*/\* R0 = R0 << 8 =2D00 \*/*
13. mov  r1, #y        ;*/\* put y value into R1=40  \*/*
14. add  r2, r0, r1, lsr #1     ;*/\* R2 = (R1>>1) + R0=2D20 \*/*
15. ldr  sp, =stack\_top                      ;*/\* SP=30200000 \*/*
16. str  r2, [sp]                            ;*/\* PC=30000018 \*/*
17. mov  r0, #z        ;*/\* put z value into R0 =57 \*/*
18. and  r0, r0, #0xFF      ;*/\* get low 8 bit from R0 £¬PC=30000020\*/*
19. mov  r1, #y        ;*/\* put y value into R1 =40 \*/*
20. add  r2, r0, r1, lsr #1     ;*/\* R2 = (R1>>1) + R0 =77\*/*
22. ldr  r0, [sp]       ;*/\* R0=2D20 \*/*
23. mov  r1, #0x01                           ;*/\* R1=1 \*/*
24. orr  r0, r0, r1                          ;*/\* R0=2D21 \*/*
25. mov  r1, R2        ;*/\* R1=77 \*/*
26. add  r2, r0, r1, lsr #1     ;*/\* R2 = (R1>>1) + R0 =2D5C \*/*
27. stop
28. b  stop            ;*/\* end the code £¬cycling\*/*
29. END

**2）截图uVision5程序运行完毕时，寄存器窗口R0-R15的值，以及memory窗口中0x30200000位置处的值**

