Tutorial 09: ARM Pseudo Instructions

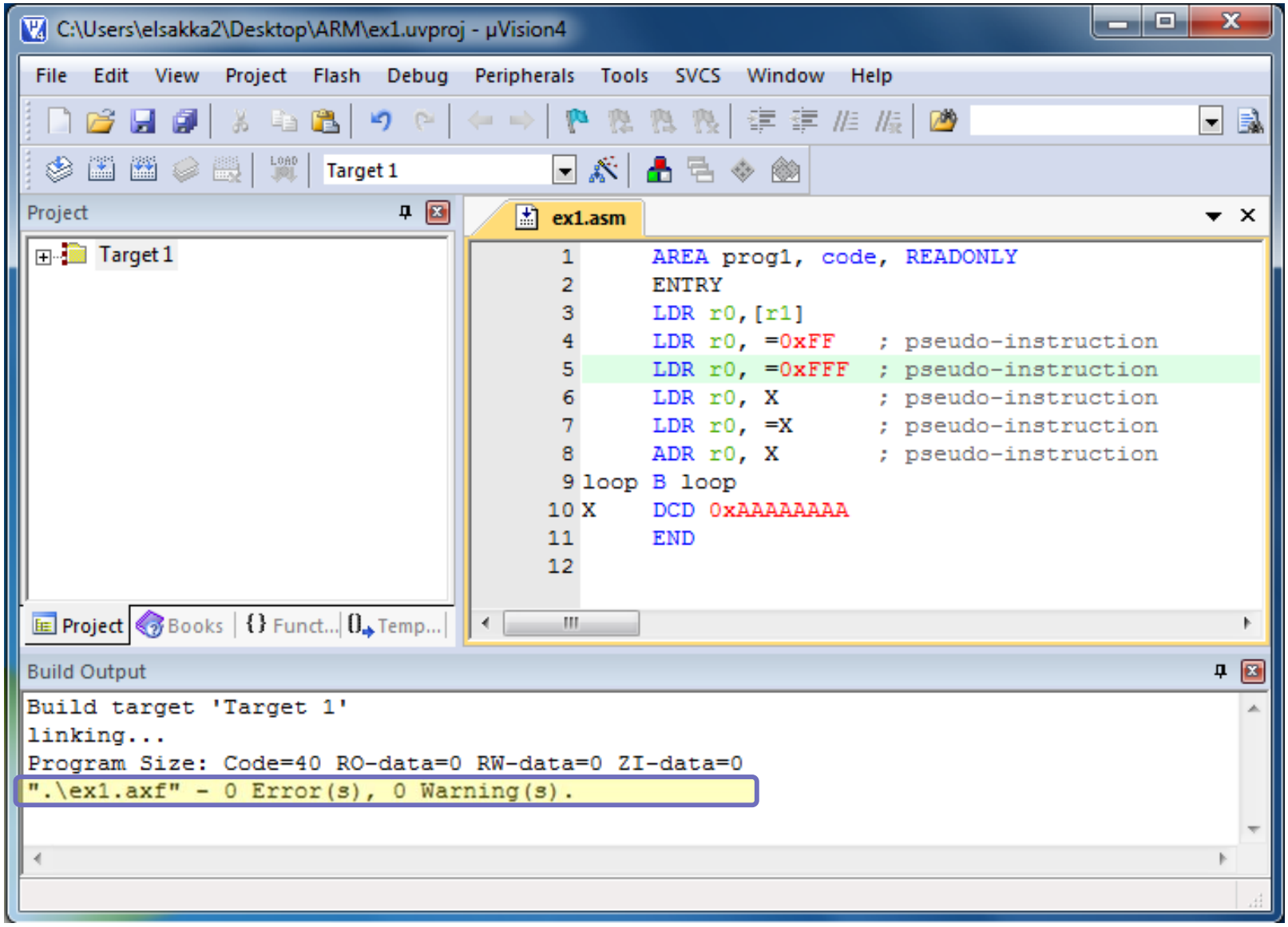
Computer Science Department
CS2208: Introduction to Computer Organization and Architecture
Winter 2020-2021
Instructor: Mahmoud R. El-Sakka
Office: MC-419
Email: elsakka@csd.uwo.ca
Phone: 519-661-2111 x86996

ARM pseudo-instructions

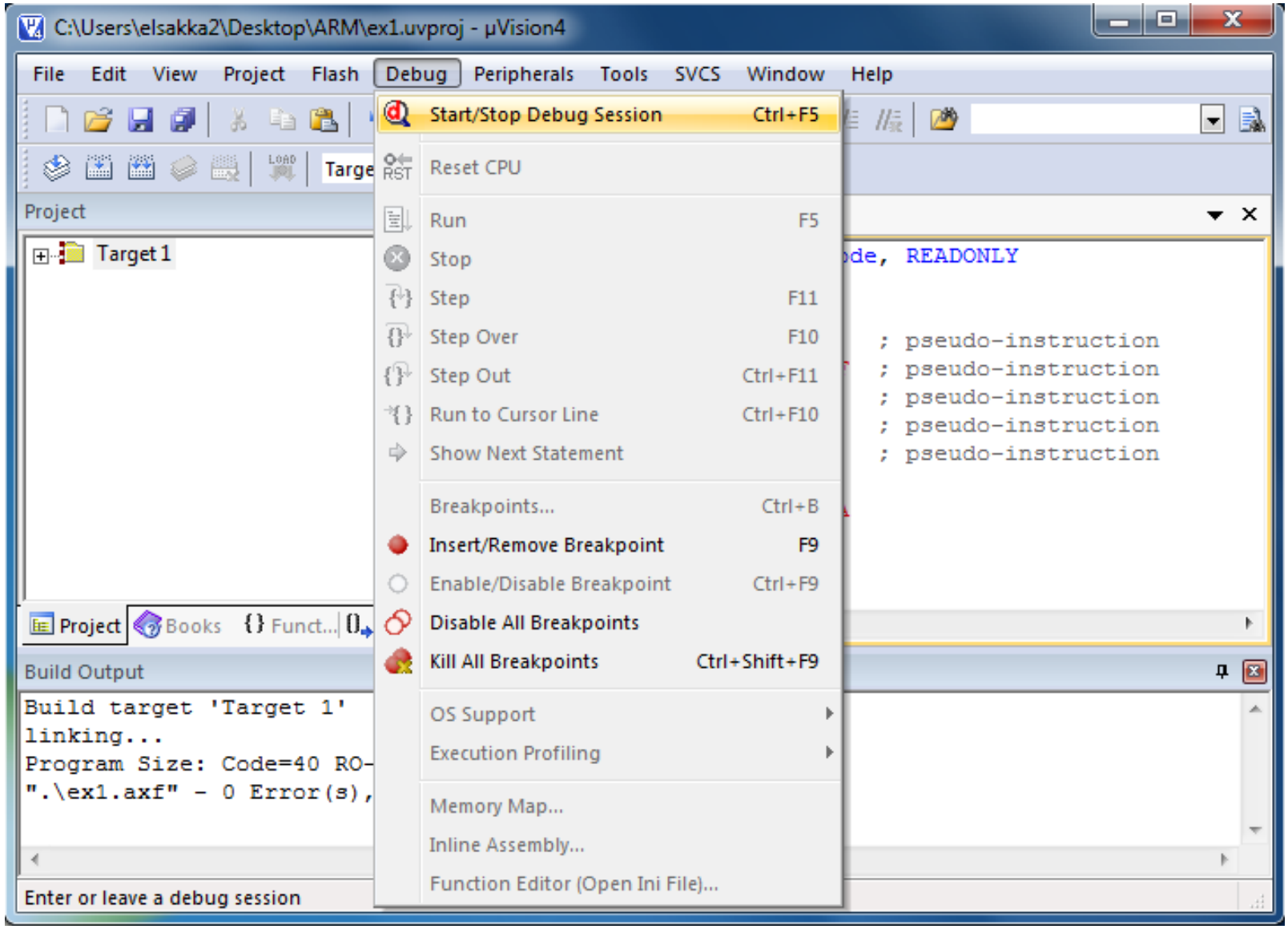
- The ARM assembler supports a number of pseudo-instructions that are translated into the appropriate combination of ARM words at assembly time.
- Consider the following assembly program:

```
      AREA prog1, code, READONLY
      ENTRY
      LDR r0,[r1]
      LDR r0, =0xFF    ; pseudo-instruction
      LDR r0, =0xFFF   ; pseudo-instruction
      LDR r0, X        ; pseudo-instruction
      LDR r0, =X       ; pseudo-instruction
      ADR r0, X        ; pseudo-instruction
loop  B loop
X     DCD 0xAAAAAAAA
      END
```

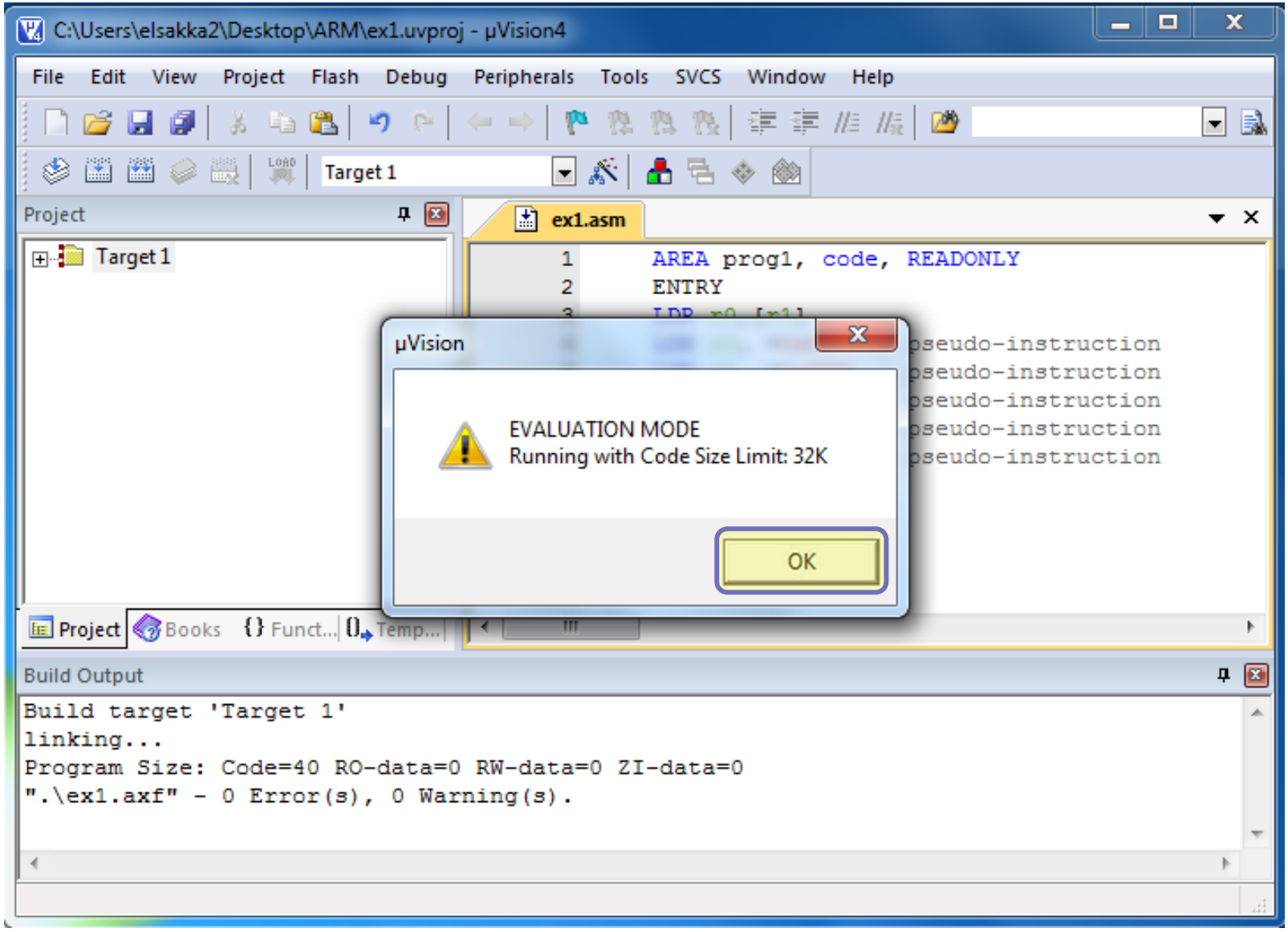
ARM pseudo-instructions



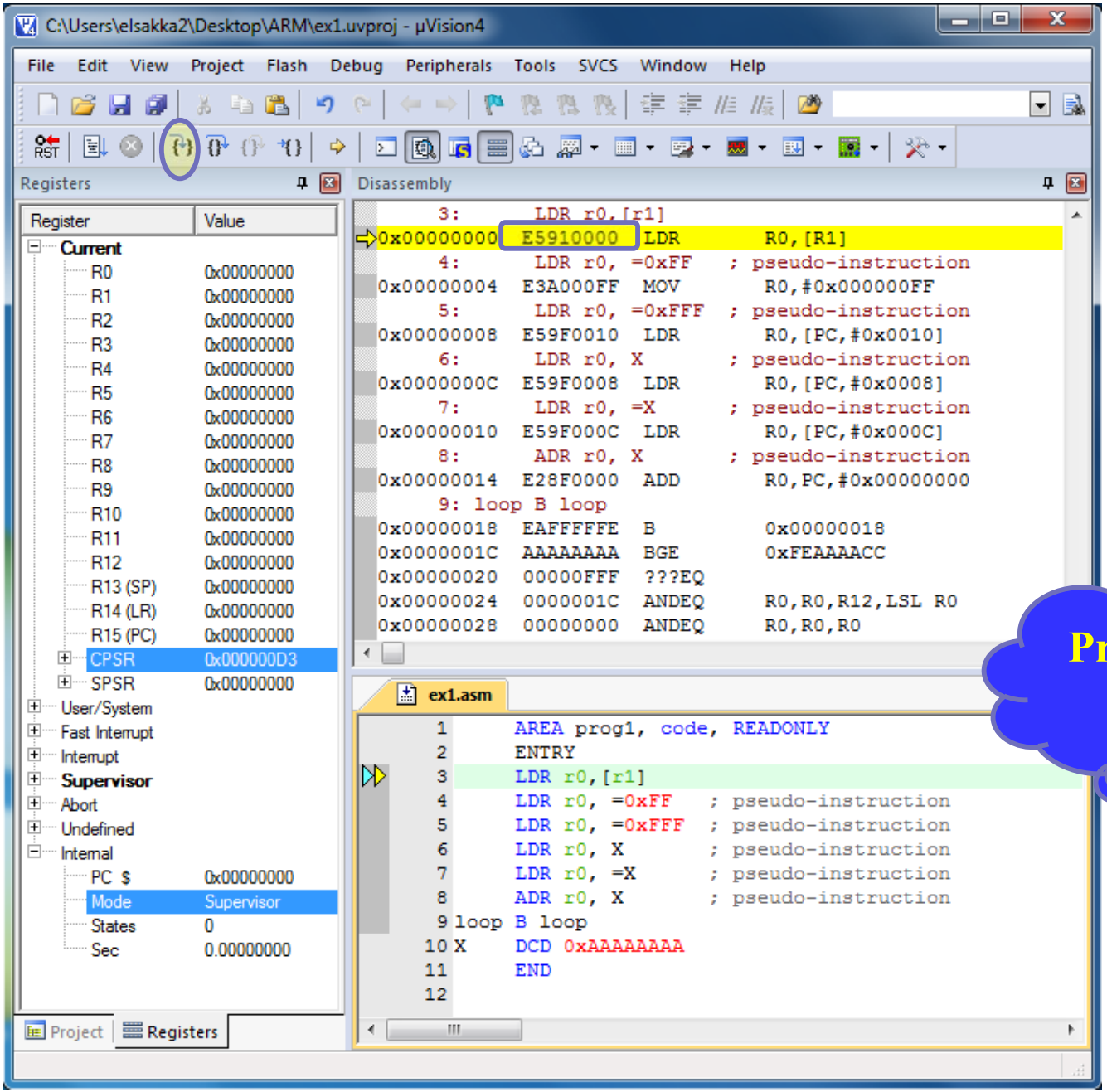
ARM pseudo-instructions



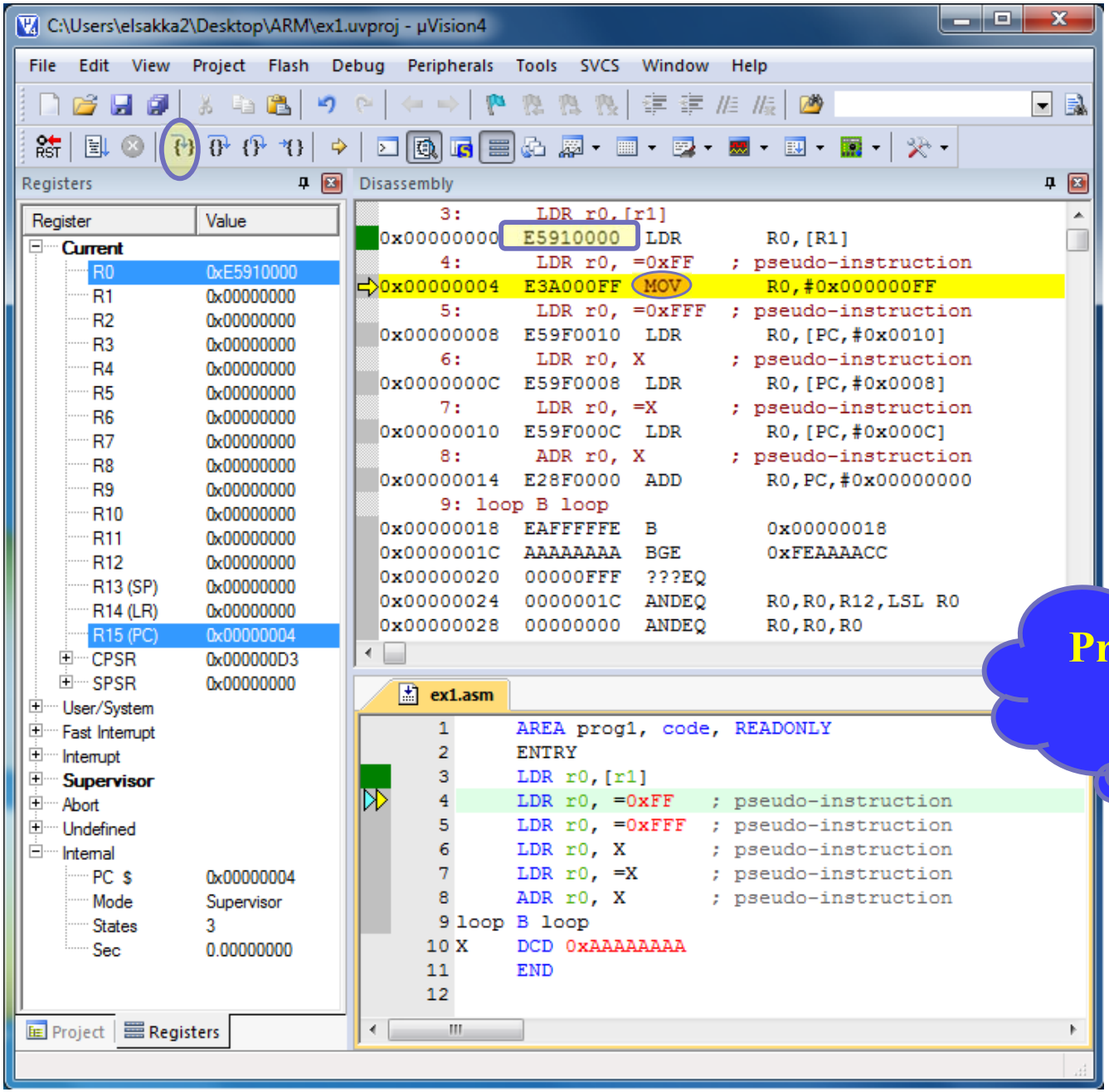
ARM pseudo-instructions



ARM pseudo-instructions



ARM pseudo-instructions

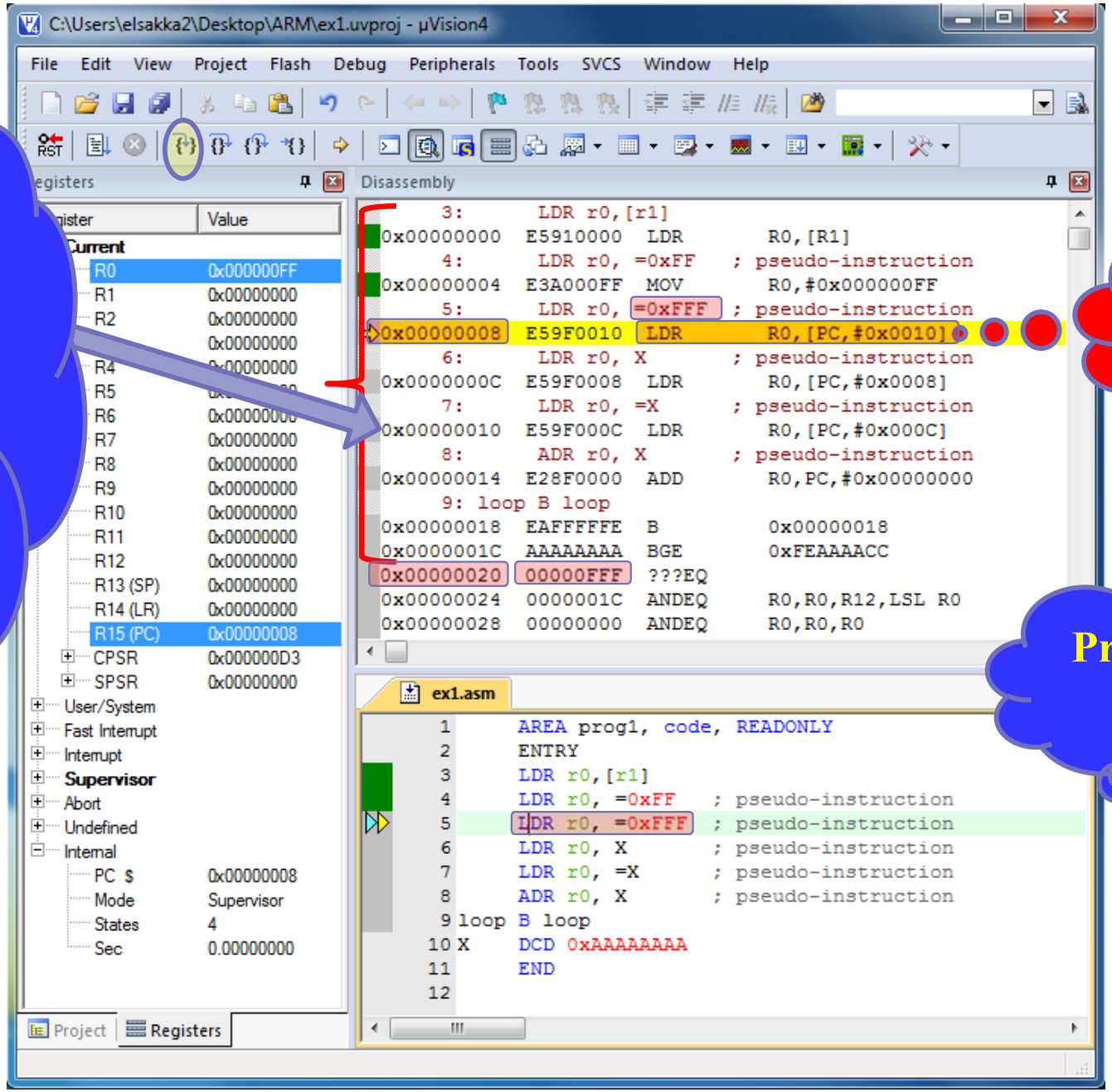


ARM pseudo-instructions

When executing the instruction at location 0x00000008, the PC value will be 0x00000010

How is this offset calculated?

Press Step, or F11

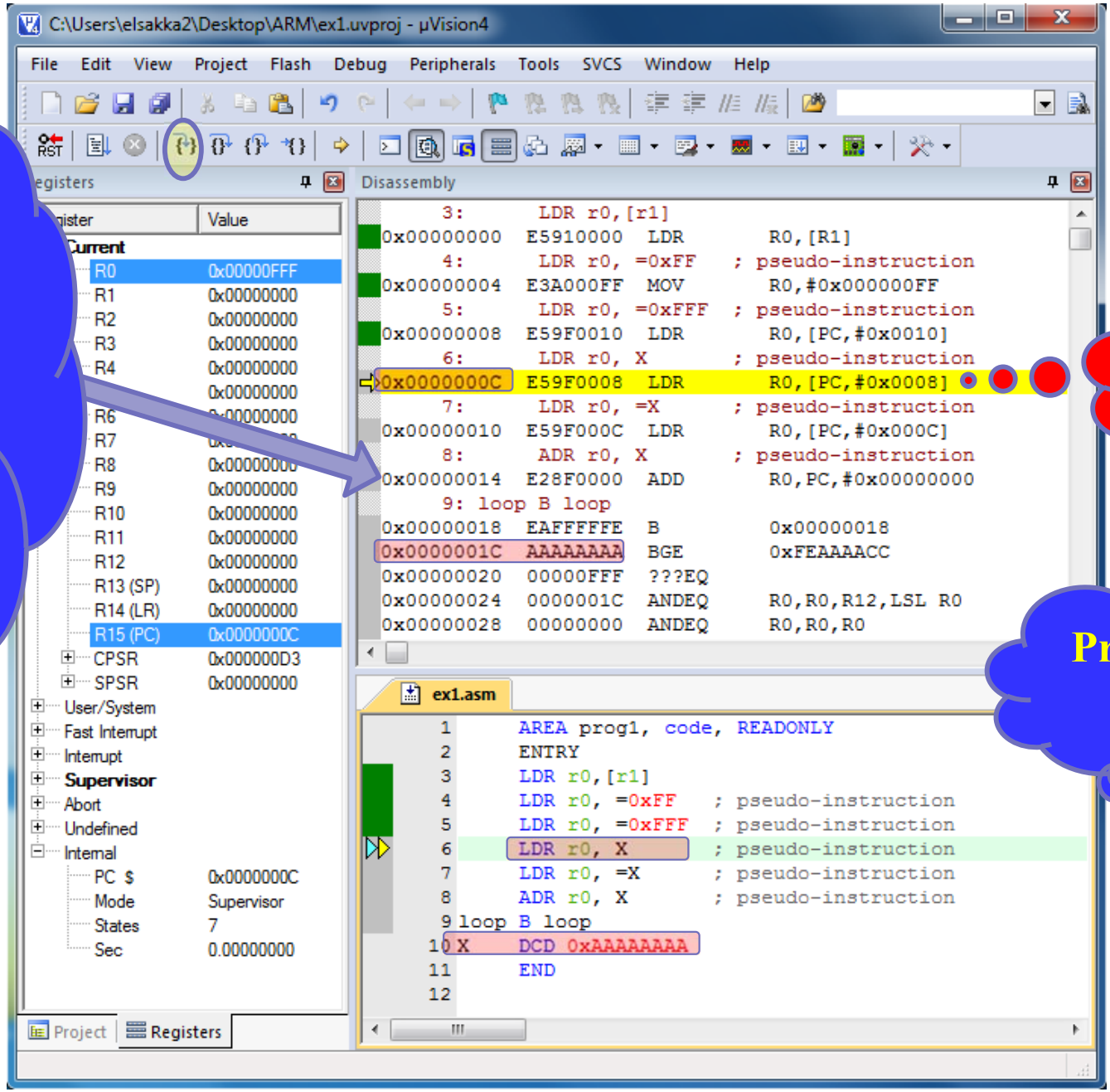


ARM pseudo-instructions

When executing the instruction at location 0x0000000C, the PC value will be 0x00000014

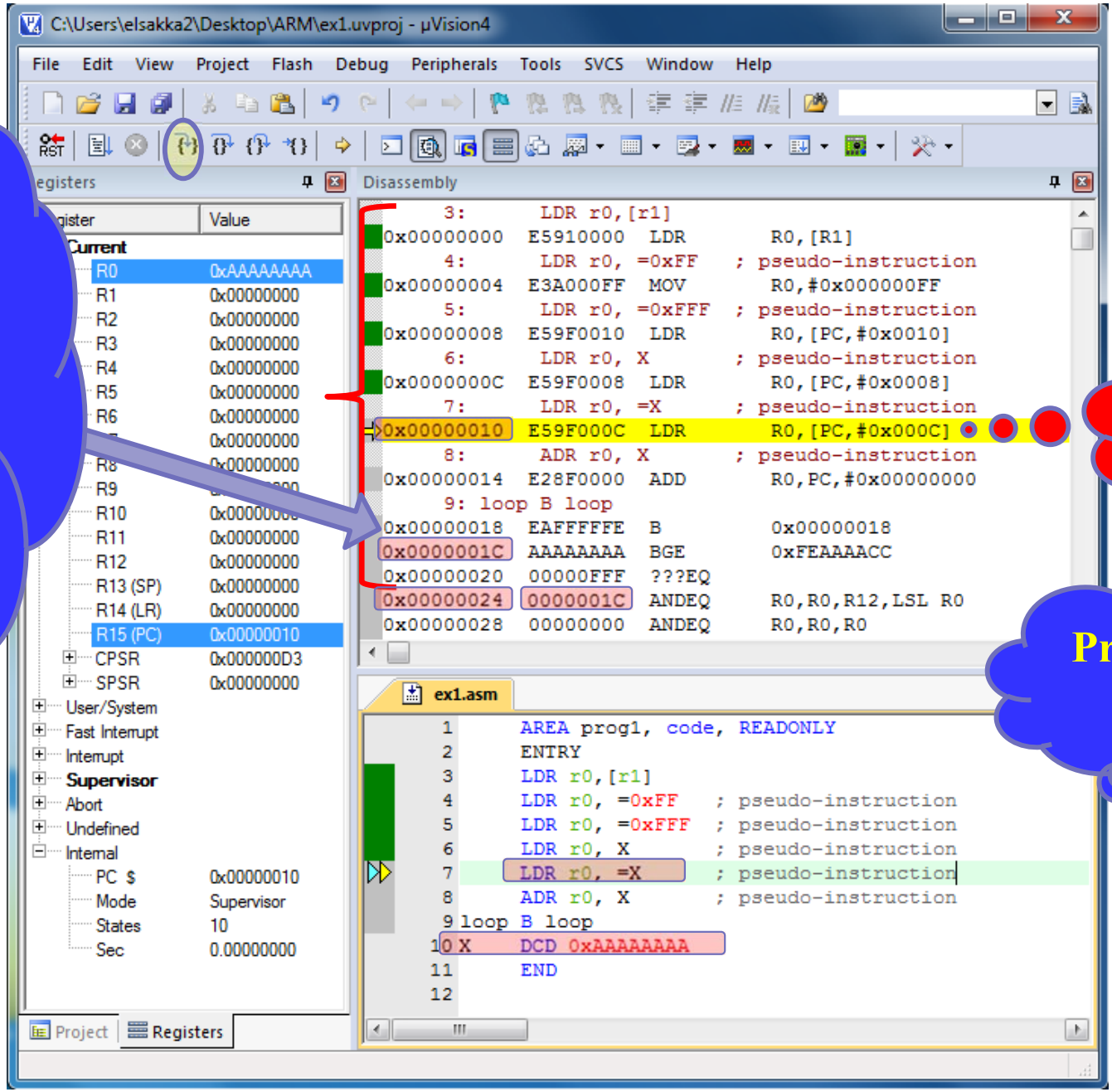
How is this offset calculated?

Press Step, or F11



ARM pseudo-instructions

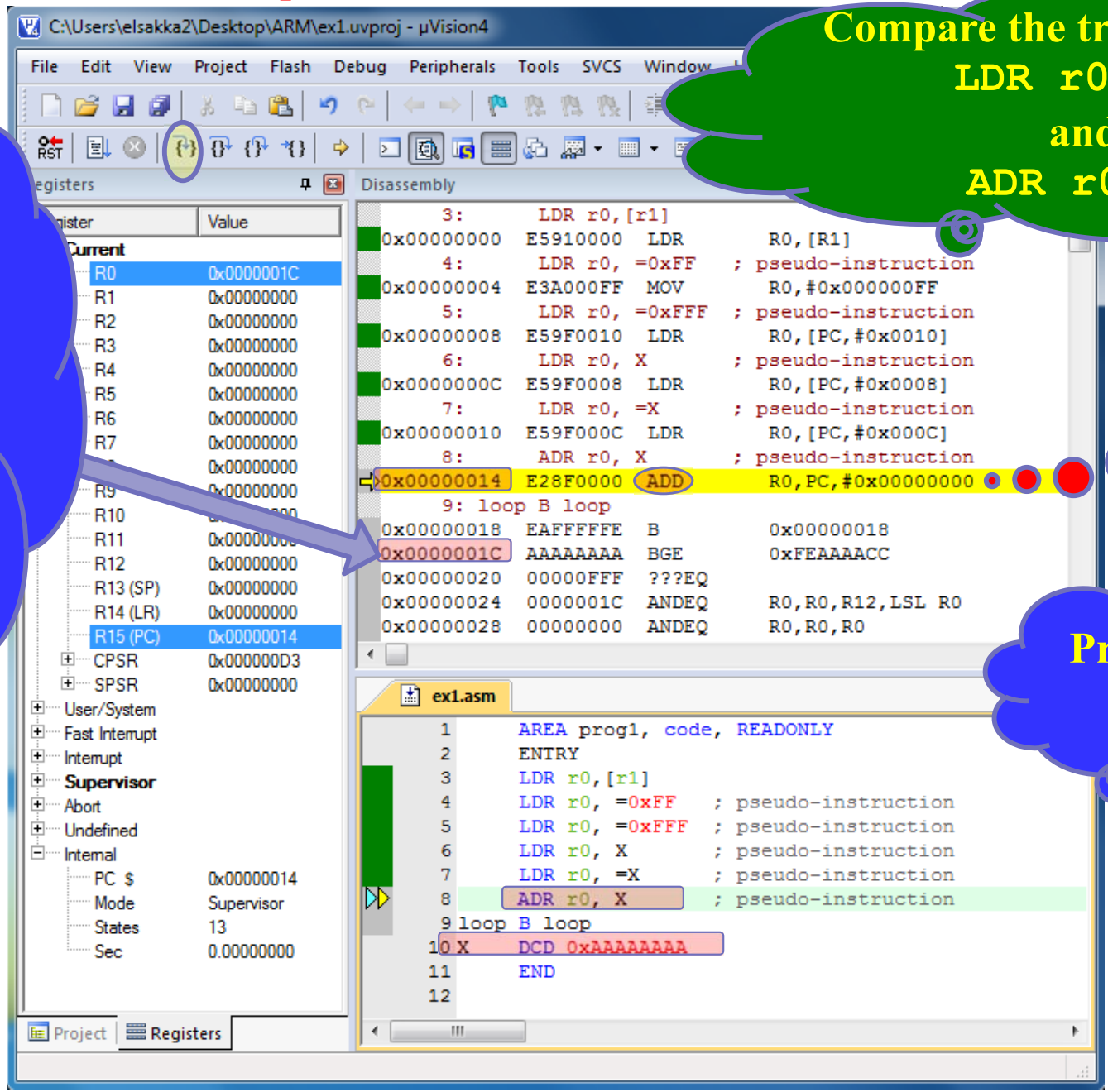
When executing the instruction at location 0x00000010, the PC value will be 0x00000018



How is this offset calculated?

Press Step, or F11

ARM pseudo-instructions



When executing the instruction at location 0x00000014, the PC value will be 0x0000001C

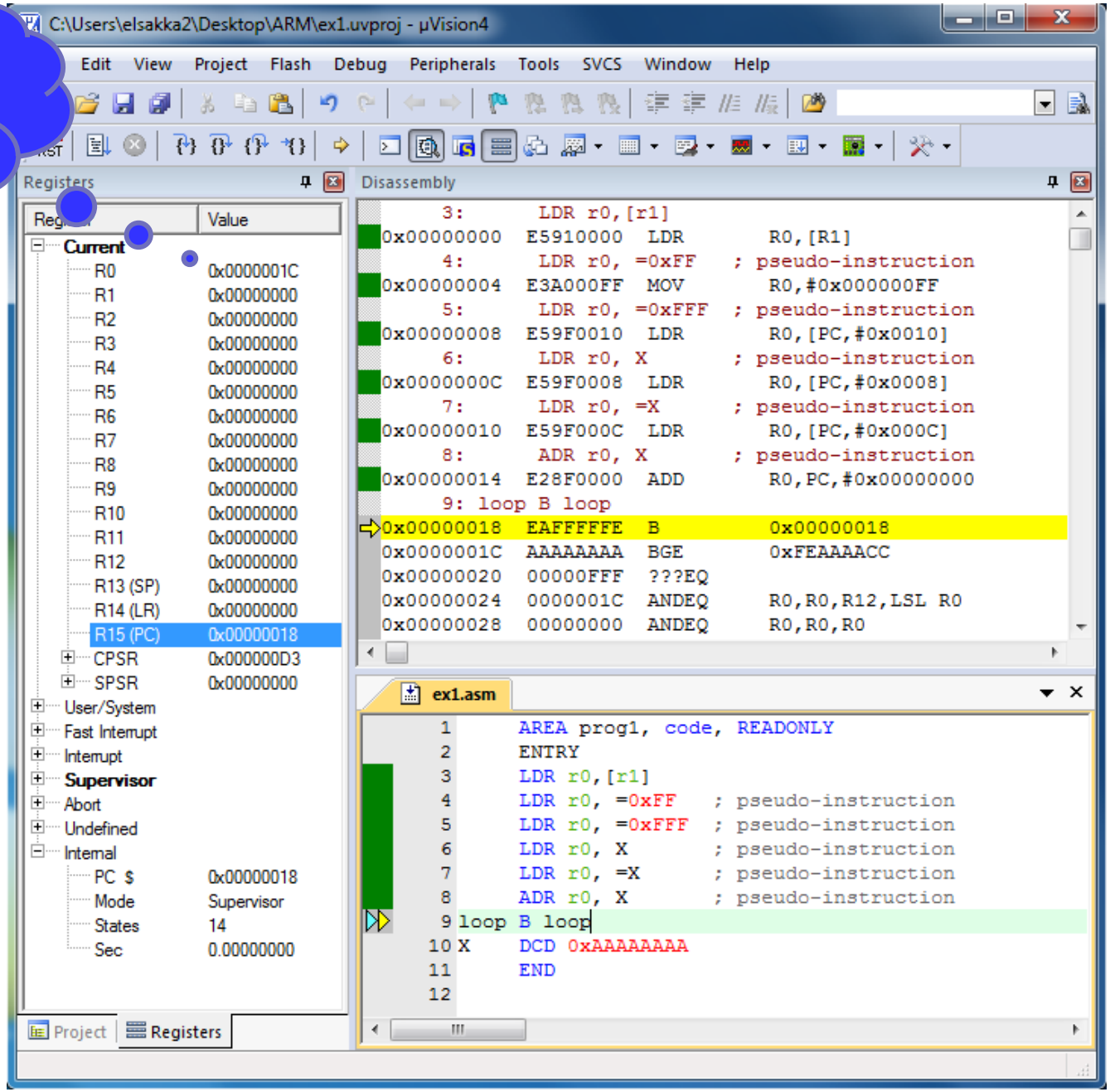
Compare the translations of `LDR r0, =X` and `ADR r0, X`

How is this offset calculated?

Press Step, or F11

ARM pseudo-instructions

Same address (no change)



ARM pseudo-instructions

- Consider we changed the previous program as follow:

```
AREA prog1, code, READONLY
     ENTRY
     LDR r0,[r1]
     LDR r0, =0xFF    ; pseudo-instruction
     LDR r0, =0xFFF   ; pseudo-instruction
     LDR r0, X        ; pseudo-instruction
     LDR r0, =X       ; pseudo-instruction
     ADR r0, X        ; pseudo-instruction
loop B loop
X    DCD 0xAAAAAAAA
     END
```

```
     AREA prog1, code, READONLY
     ENTRY
     LDR r0,[r1]
     LDR r0, =0xFF    ; pseudo-instruction
     LDR r0, =0xFFF   ; pseudo-instruction
     LDR r0, X        ; pseudo-instruction
     LDR r0, =X       ; pseudo-instruction
     ADR r0, X        ; pseudo-instruction
loop B loop

     AREA prog1, data, READONLY
X    DCD 0xAAAAAAAA
     END
```

- What is the effect of this change on the generated code?

ARM pseudo-instructions