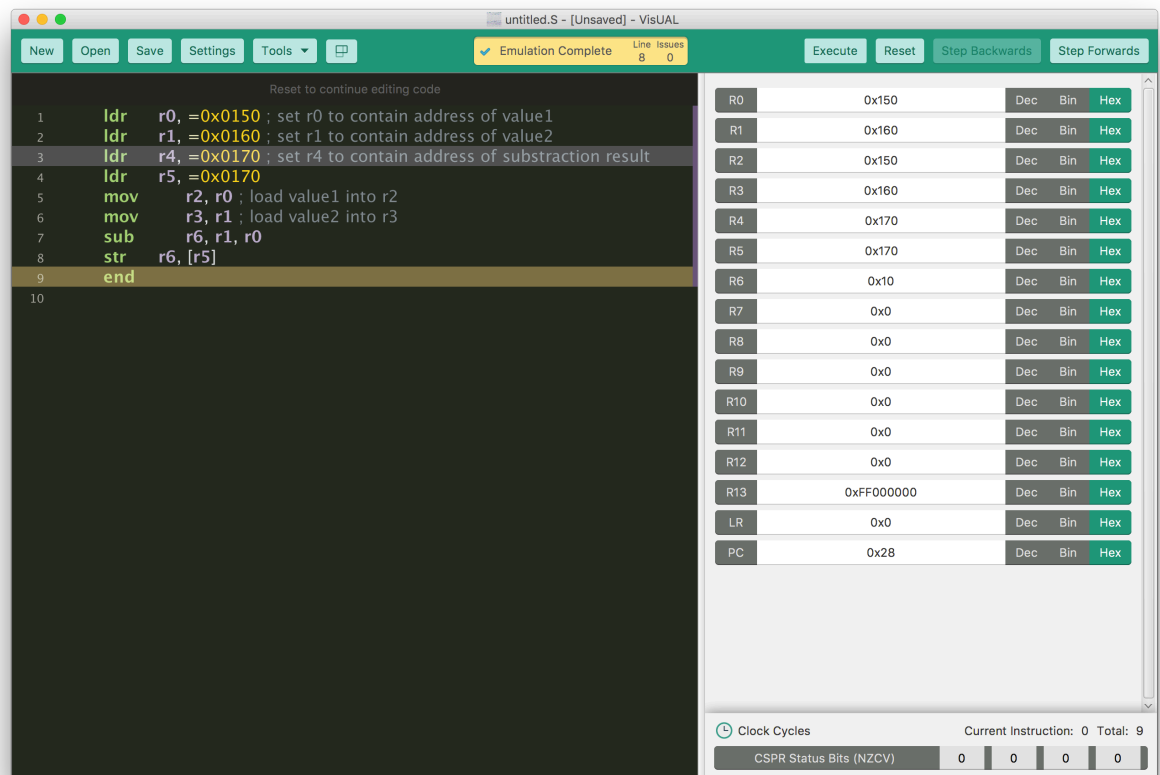


Willis Allstead
4/16/18
CS 219

HW9

1)

- I'm trying to learn how to subtract two 64-bit integers in ARM assembly and save it to a memory location.



View Memory Contents

Start address: 0x150

End address: 0x1100

Memory Map

Word Address	Byte 3	Byte 2	Byte 1	Byte 0	Word Value
0x170	0x0	0x0	0x0	0x10	0x10

Word Value Format

DecHex

Memory Map Key

InstructionsData

View Symbols

Type	Name	Address (value of symbol)	Contents of memory location
No data or code symbols declared			

3)

- I'm trying to learn how to compare two numbers in ARM assembly.

The screenshot shows the VisUAL ARM emulator interface. The main window displays assembly code for comparing two numbers. The code is as follows:

```
1  LDR    r0, =0x0158
2
3  LDR    r1, =0x0150 ; load in val1
4
5  LDR    r2, =0x0154 ; load in val2
6
7  CMP    r1, r2 ; compare them
8
9  BGE    FIRSTLARGER ; if first isn't larger
10
11 STR    r2, [r0] ; store second
12
13 FIRSTLARGER ; else
14
15 STR    r1, [r0] ; store first
16
17 END
18
```

On the right side, there is a register window showing the values of R0 through R13, LR, and PC. The values are:

Register	Value	Dec	Bin	Hex
R0	0x158			
R1	0x150			
R2	0x154			
R3	0x0			
R4	0x0			
R5	0x0			
R6	0x0			
R7	0x0			
R8	0x0			
R9	0x0			
R10	0x0			
R11	0x0			
R12	0x0			
R13	0xFF000000			
LR	0x0			
PC	0x28			

At the bottom, there is a status bar showing "Clock Cycles" and "Current Instruction: 0 Total: 9". The CSCR Status Bits (NZCV) are shown as 1, 0, 0, 0.

The screenshot shows the "View Memory Contents" dialog box. It has input fields for "Start address" (0x150) and "End address" (0x1100). The "Memory Map" tab is selected, showing a table of memory contents:

Word Address	Byte 3	Byte 2	Byte 1	Byte 0	Word Value
0x158	0x0	0x0	0x1	0x50	0x150

At the bottom, there is a "Word Value Format" section with "Dec" and "Hex" buttons. The "Memory Map Key" section has "Instructions" and "Data" buttons.

4)

- I'm trying to learn how to logically left shift a number in ARM assembly and save it to a memory location

The screenshot shows the VisUAL ARM emulator interface. The main window displays assembly code for an ARM processor. The code is as follows:

```
1  ldr r0, =0x0150
2  ldr r1, =0x0154
3  ldr r2, =0x0158 ; set 3 registers to addresses of memory to be used
4
5  mov r4, r0 ; r0 is the number to be shifted
6
7  mov r5, r1 ; r1 is the number to be shifted with
8
9  lsl r6, r4, r5 ; logical shift r4 by r5 and set r6 to result
10
11 str r6, [r2]
12
13 end
14
```

The right side of the interface shows a register table with 14 registers (R0-R13), LR, and PC. The values are as follows:

Register	Value	Dec	Bin	Hex
R0	0x150			
R1	0x154			
R2	0x158			
R3	0x0			
R4	0x150			
R5	0x154			
R6	0x0			
R7	0x0			
R8	0x0			
R9	0x0			
R10	0x0			
R11	0x0			
R12	0x0			
R13	0xFF000000			
LR	0x0			
PC	0x24			

At the bottom, the status bar shows "Clock Cycles: 0", "Current Instruction: 0", "Total: 8", and "CSPR Status Bits (NZCV): 0 0 0 0".

View Memory Contents

Start address: 0x150

End address: 0x1100

Memory Map

Word Address	Byte 3	Byte 2	Byte 1	Byte 0	Word Value	
0x158	0x0	0x0	0x0	0x0	0x0	

Word Value Format

DecHex

Memory Map Key

InstructionsData

View Symbols

Type	Name	Address (value of symbol)	Contents of memory location
No data or code symbols declared			

5)

- I'm trying to learn how to find the largest 32-bit word in a list, and store the largest word in a memory location.

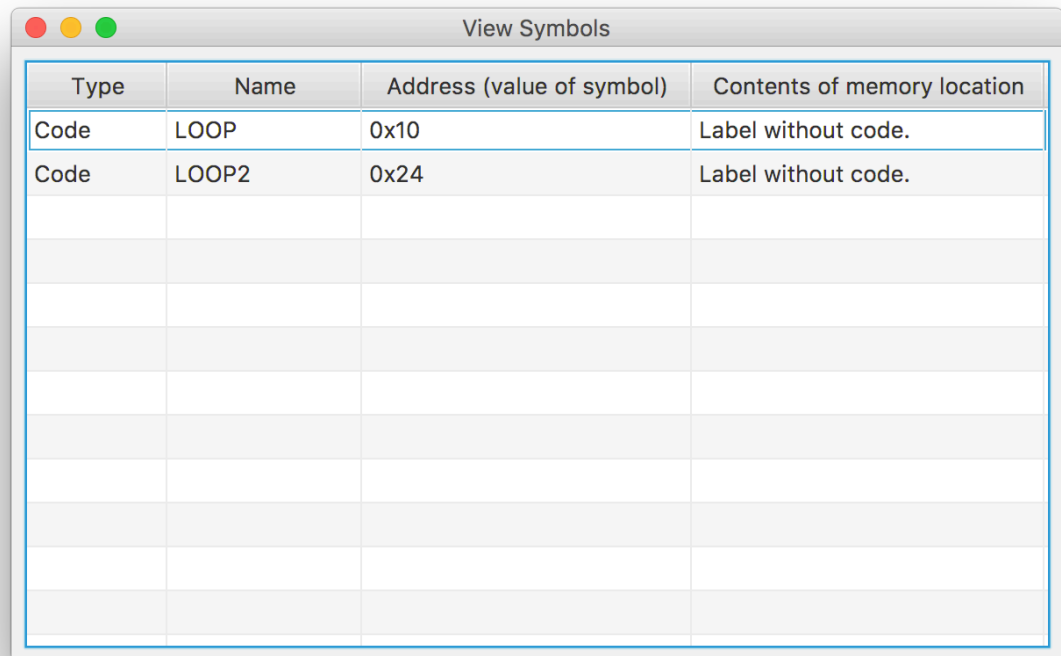
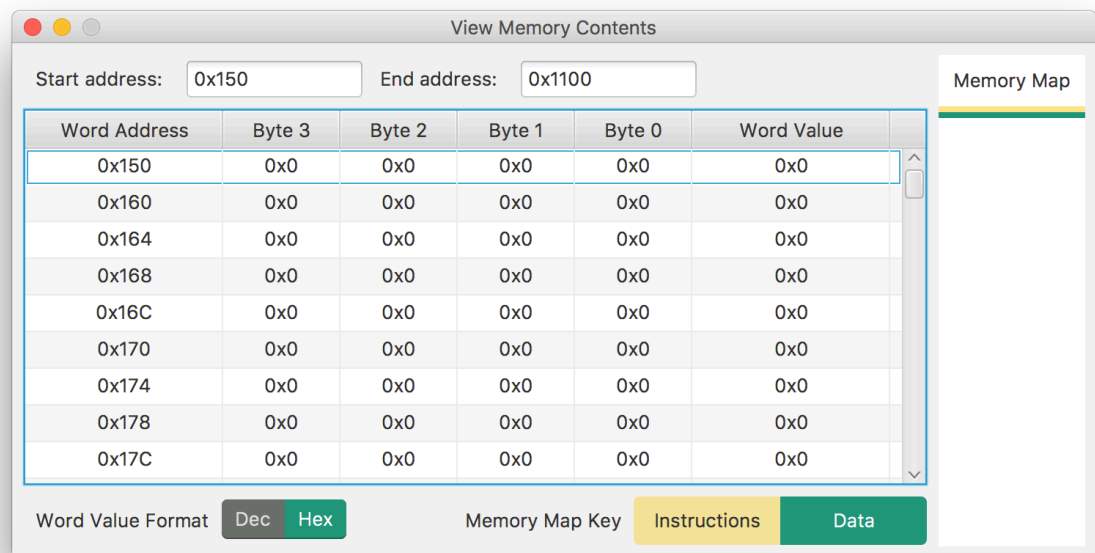
The screenshot shows the VisUAL ARM emulator interface. The main window is titled "untitled.S - [Unsaved] - VisUAL". The interface includes a menu bar (New, Open, Save, Settings, Tools), a toolbar (Emulation Running, Line Issues: 8, 0), and navigation buttons (Execute, Reset, Step Backwards, Step Forwards). The assembly code is displayed in a dark-themed editor with line numbers 1 through 18. The code is as follows:

```
1
2  LDR    R0,=0x0160 ; where list begins address
3  EOR    R1,R1,R1
4  LDR    R4,=0x0150 ; length of list address
5  LDR    R2, [R4] ; length of list
6 LOOP
7  LDR    R3,[R0] ; from beginning of list
8  CMP    R3,R1 ; compare element to r1
9  BCC    LOOP2 ; if lower go to Loop2
10 MOV    R1,R3 ; else copy element into r1
11 LOOP2
12 ADD    R0,R0,#4 ; stride by 4s
13 SUBS   R2,R2,#0x1
14 BNE    LOOP
15 LDR    R0,=0x0154 ; store largest entry at this address
16 STR    R0,[R1] ; store largest entry
17 End
18
```

On the right side, there is a register window showing the values of 16 registers (R0-R15) and the Program Counter (PC). The values are as follows:

Register	Value	Dec	Bin	Hex
R0	0x20A0			
R1	0x0			
R2	0xFFFFF830			
R3	0x0			
R4	0x150			
R5	0x0			
R6	0x0			
R7	0x0			
R8	0x0			
R9	0x0			
R10	0x0			
R11	0x0			
R12	0x0			
R13	0xFF000000			
LR	0x0			
PC	0x38			

At the bottom, there is a status bar showing "Clock Cycles" (Current Instruction: 3, Total: 20005) and "CSPR Status Bits (NZCV)" (1, 0, 1, 0).



6)

- I'm trying to learn how to scan a list of unsigned bytes to find the smallest and largest numbers in the list.

untitled.S - [Unsaved] - VisUAL

NewOpenSaveSettingsTools

Emulation CompleteLine Issues310

ExecuteResetStep BackwardsStep Forwards

Reset to continue editing code

```
1 main
2
3   ldr    r0,=data0
4   ldr    r1,=0x0158 ;length of list
5   ldr    r3,=0x0160 ;starting address
6   ldr    r4,=0x0150 ;smallest integer address
7   ldr    r5,=0x0154 ;largest integer address
8   ldr    r6,[r0],#04 ; load with 4th data item
9   sub    r1,r1,#01
10 back
11   ldr    r7,[r0]
12   cmp    r6,r7
13   bls    less ; if it's less
14   mov    r6,r7
15 less
16   add    r0,r0,#04
17   sub    r1,r1,#01
18   cmp    r1,#00
19   bne    back ; if it's not equal go back
20   str    r6,[r4]
21
22
23 back1
24   ldr    r7,[r0]
25   cmp    r6,r7
26   bhs    high ; if it's not equal or greater go to high
27   mov    r6,r7
28 high
29   add    r0,r0,#04
30   bne    back1
31   str    r6,[r5]
32
33 data0 dcd    &64,&05,&9,&00,&65
34   END
35
```

R0	0x664	Dec	Bin	Hex
R1	0x0	Dec	Bin	Hex
R2	0x0	Dec	Bin	Hex
R3	0x160	Dec	Bin	Hex
R4	0x150	Dec	Bin	Hex
R5	0x154	Dec	Bin	Hex
R6	0x0	Dec	Bin	Hex
R7	0x0	Dec	Bin	Hex
R8	0x0	Dec	Bin	Hex
R9	0x0	Dec	Bin	Hex
R10	0x0	Dec	Bin	Hex
R11	0x0	Dec	Bin	Hex
R12	0x0	Dec	Bin	Hex
R13	0xFF000000	Dec	Bin	Hex
LR	0x0	Dec	Bin	Hex
PC	0x7C	Dec	Bin	Hex

Clock CyclesCurrent Instruction: 0 Total: 4132

CSPR Status Bits (NZCV)0110

View Memory Contents

Start address: 0x150End address: 0x1100

Memory Map

Word Address	Byte 3	Byte 2	Byte 1	Byte 0	Word Value
0x150	0x0	0x0	0x0	0x0	0x0
0x154	0x0	0x0	0x0	0x0	0x0
0x158	0x0	0x0	0x0	0x0	0x0
0x15C	0x0	0x0	0x0	0x0	0x0
0x160	0x0	0x0	0x0	0x0	0x0
0x164	0x0	0x0	0x0	0x0	0x0
0x168	0x0	0x0	0x0	0x0	0x0
0x16C	0x0	0x0	0x0	0x0	0x0
0x170	0x0	0x0	0x0	0x0	0x0

Word Value FormatDecHexMemory Map KeyInstructionsData

