

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date: 27—01—2021

Name: SUHAN B REVANKAR	SRN: PES2UG19CS412	Section G
------------------------	-----------------------	--------------

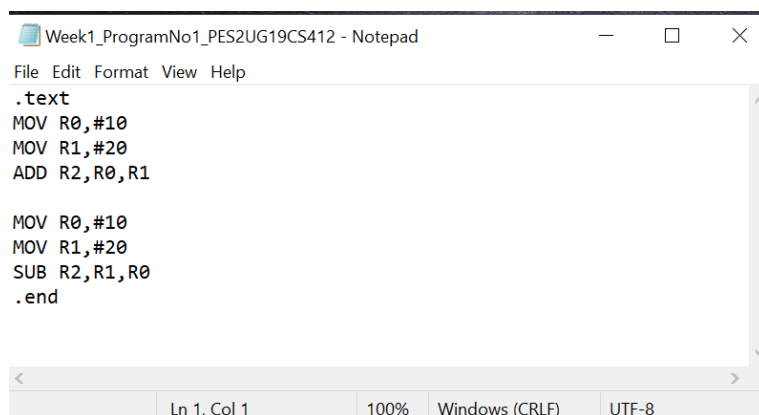
Week# 1 Program Number: 1

Title of the Program

Write an ALP using ARM instruction set to add and subtract two 32 bit numbers .Both numbers are in registers.

I. ARM Assembly Code for each program

Test case used in class

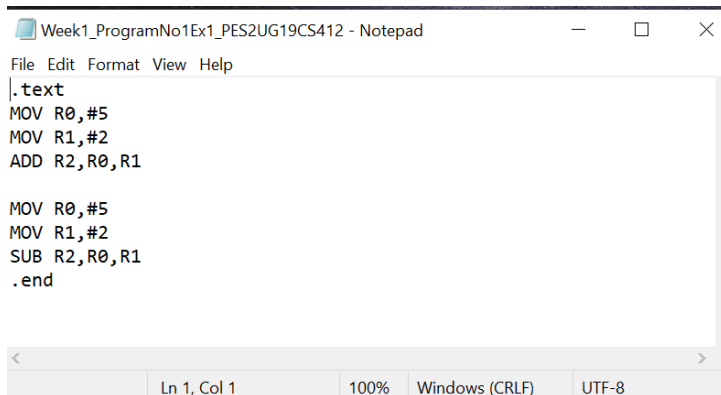


```
Week1_ProgramNo1_PES2UG19CS412 - Notepad
File Edit Format View Help
.text
MOV R0,#10
MOV R1,#20
ADD R2,R0,R1

MOV R0,#10
MOV R1,#20
SUB R2,R1,R0
.end

Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

Own example



```
Week1_ProgramNo1Ex1_PES2UG19CS412 - Notepad
File Edit Format View Help
|.text
MOV R0,#5
MOV R1,#2
ADD R2,R0,R1

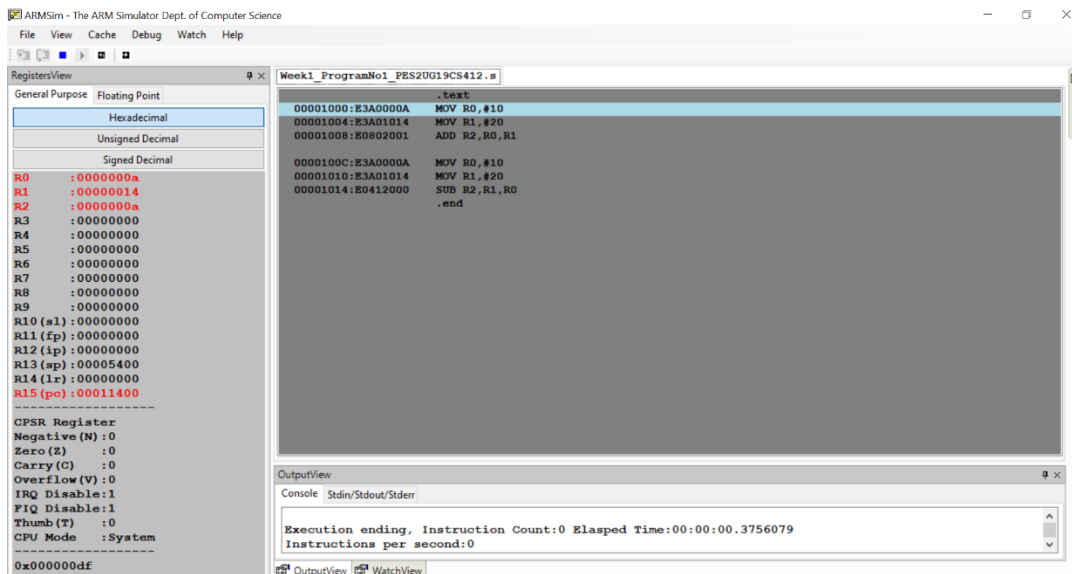
MOV R0,#5
MOV R1,#2
SUB R2,R0,R1
.end

Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

II. Output Screen Shot (Register Window, Output window)

The output should be verified with 2 test cases (one example shown in class, one example of own choice)

Output of test case



ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose Floating Point

Register	Hexadecimal	Unsigned Decimal	Signed Decimal
R0	:0000000a		
R1	:00000014		
R2	:0000000a		
R3	:00000000		
R4	:00000000		
R5	:00000000		
R6	:00000000		
R7	:00000000		
R8	:00000000		
R9	:00000000		
R10 (s1)	:00000000		
R11 (fp)	:00000000		
R12 (ip)	:00000000		
R13 (sp)	:00005400		
R14 (lr)	:00000000		
R15 (pc)	:00011400		

CPSR Register

Negative (N) : 0

Zero (Z) : 0

Carry (C) : 0

Overflow (V) : 0

IRQ Disable: 1

FIQ Disable: 1

Thumb (T) : 0

CPU Mode : System

0x000000df

Week1_ProgramNo1_PES2UG19CS412.s

```
.text
00001000:E3A0000A MOV R0,#10
00001004:E3A01014 MOV R1,#20
00001008:E0802001 ADD R2,R0,R1

0000100C:E3A0000A MOV R0,#10
00001010:E3A01014 MOV R1,#20
00001014:E0412000 SUB R2,R1,R0
.end
```

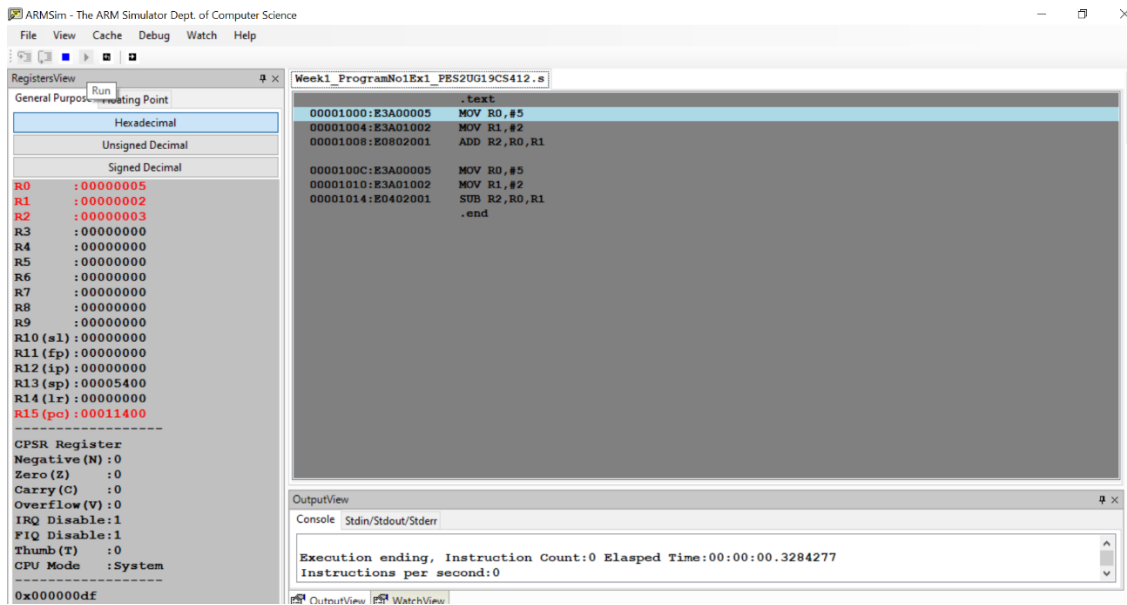
OutputView

Console Stdin/Stdout/Stderr

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.3756079

Instructions per second:0

Output of own example



III. Output table for each program

TESTCASE			
R0=10=Hex 0A R1=20=Hex 14			
After addition R2=30=Hex 1E			
After subtraction R2=10=Hex 0A			
R0	R1	Arithmetic operation	RESULTS
0x0A	0x14	ADD	R2=0x1E
0x0A	0x14	SUB	R2=0x0A

Own example

Own example R0=5=Hex05 R1=2=Hex02 After addition R2=7=Hex07 After subtraction R2=3=Hex03			
R0	R1	Arithmetic operation	Results
0x05	0x02	ADD	R2=0x07
0x05	0x02	SUB	R2=0x03

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:27—01—2021

Name: SUHAN B REVANKAR	SRN: PES2UG19CS412	Section G
------------------------	-----------------------	--------------

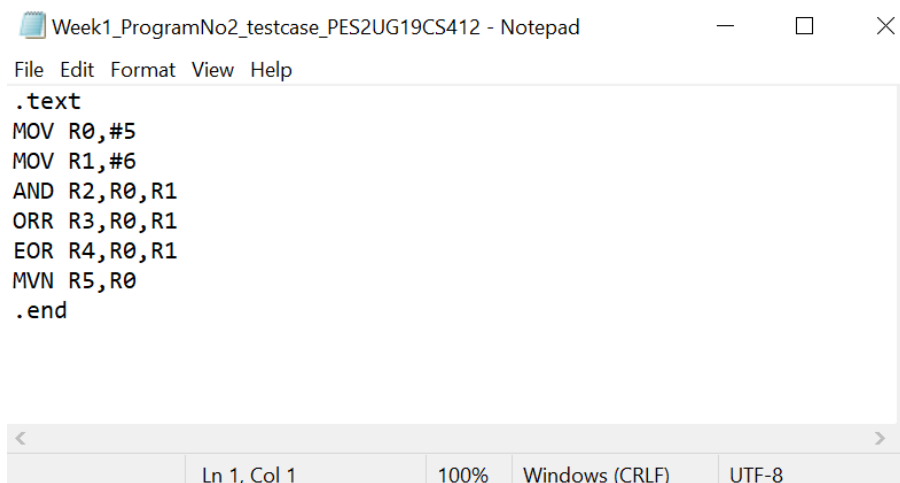
Week# 1 Program Number: 2

Title of the Program

Write an ALP to demonstrate logical operations. All operands are in registers.

I. ARM Assembly Code for each program

Test case used in class

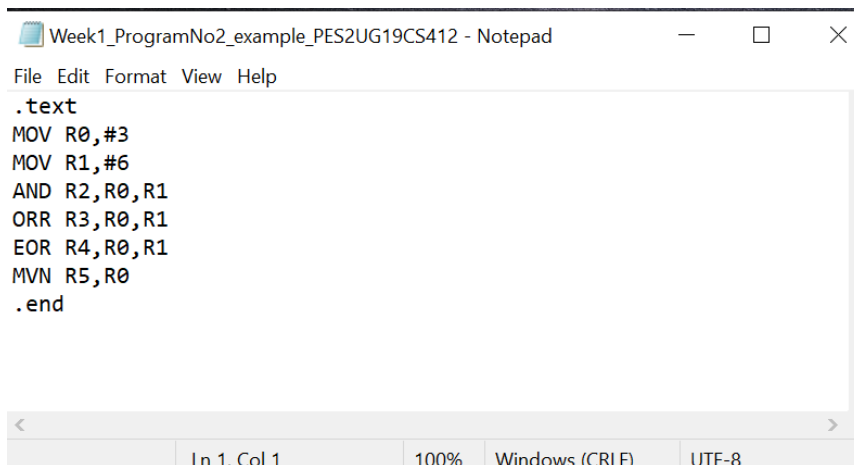


A screenshot of a Notepad window titled "Week1_ProgramNo2_testcase_PES2UG19CS412 - Notepad". The window contains the following ARM assembly code:

```
.text
MOV R0,#5
MOV R1,#6
AND R2,R0,R1
ORR R3,R0,R1
EOR R4,R0,R1
MVN R5,R0
.end
```

The status bar at the bottom indicates "Ln 1, Col 1", "100%", "Windows (CRLF)", and "UTF-8".

Own example



A screenshot of a Notepad window titled "Week1_ProgramNo2_example_PES2UG19CS412 - Notepad". The window contains the following ARM assembly code:

```
.text
MOV R0,#3
MOV R1,#6
AND R2,R0,R1
ORR R3,R0,R1
EOR R4,R0,R1
MVN R5,R0
.end
```

The status bar at the bottom indicates "Ln 1, Col 1", "100%", "Windows (CRLF)", and "UTF-8".

II. Output Screen Shot (Register Window, Output window)

The output should be verified with 2 test cases
(one example shown in class, one example of own choice)

Output of testcase used in the class

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose Floating Point

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 : 00000005
R1 : 00000006
R2 : 00000004
R3 : 00000007
R4 : 00000003
R5 : 0000000a
R6 : 00000000
R7 : 00000000
R8 : 00000000
R9 : 00000000
R10 (s1) : 00000000
R11 (fp) : 00000000
R12 (ip) : 00000000
R13 (sp) : 00005400
R14 (lr) : 00000000
R15 (pc) : 00011400

CPSR Register
Negative (N) : 0
Zero (Z) : 0
Carry (C) : 0
Overflow (V) : 0
IRQ Disable : 1
FIQ Disable : 1
Thumb (T) : 0
CPU Mode : System

0x000000df

Week1_ProgramNo2_testcase_PES20G19CS412.s

```
.text
00001000:E3A00005 MOV R0,#5
00001004:E3A01006 MOV R1,#6
00001008:E0002001 AND R2,R0,R1
0000100C:E1803001 ORR R3,R0,R1
00001010:E0204001 EOR R4,R0,R1
00001014:E1E05000 MVN R5,R0
.end
```

OutputView

Console Stdin/Stdout/Stderr

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.5268387
Instructions per second:0

Output of own example

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose Floating Point

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 : 00000003
R1 : 00000006
R2 : 00000002
R3 : 00000007
R4 : 00000005
R5 : 0000000c
R6 : 00000000
R7 : 00000000
R8 : 00000000
R9 : 00000000
R10 (s1) : 00000000
R11 (fp) : 00000000
R12 (ip) : 00000000
R13 (sp) : 00005400
R14 (lr) : 00000000
R15 (pc) : 00011400

CPSR Register
Negative (N) : 0
Zero (Z) : 0
Carry (C) : 0
Overflow (V) : 0
IRQ Disable : 1
FIQ Disable : 1
Thumb (T) : 0
CPU Mode : System

0x000000df

Week1_ProgramNo2_example_PES20G19CS412.s

```
.text
00001000:E3A00003 MOV R0,#3
00001004:E3A01006 MOV R1,#6
00001008:E0002001 AND R2,R0,R1
0000100C:E1803001 ORR R3,R0,R1
00001010:E0204001 EOR R4,R0,R1
00001014:E1E05000 MVN R5,R0
.end
```

OutputView

Console Stdin/Stdout/Stderr

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.4140562
Instructions per second:0

III. Output table for each program

TESTCASE

R0	R1	Logic operation		Results
0x05	0x06	ADD	ADD	R2=0x04
0x05	0x06	OR	OR	R3=0x07
0x05	0x06	EX-OR	EX-OR	R4=0x03
0x05		NOT	MVN	R5=0xffffffffa

Own example

R0	R1	Logic operations		Results
0x03	0x06	AND	AND	R2=0x02
0x03	0x06	OR	OR	R3=0x07
0x03	0x06	EX-OR	EX-OR	R4=0x05
0x03		NOT	MNV	R5=fffffffc

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:27—01--2021

Name: SUHAN B REVANKAR	SRN: PES2UG19CS412	Section G
------------------------	-----------------------	--------------

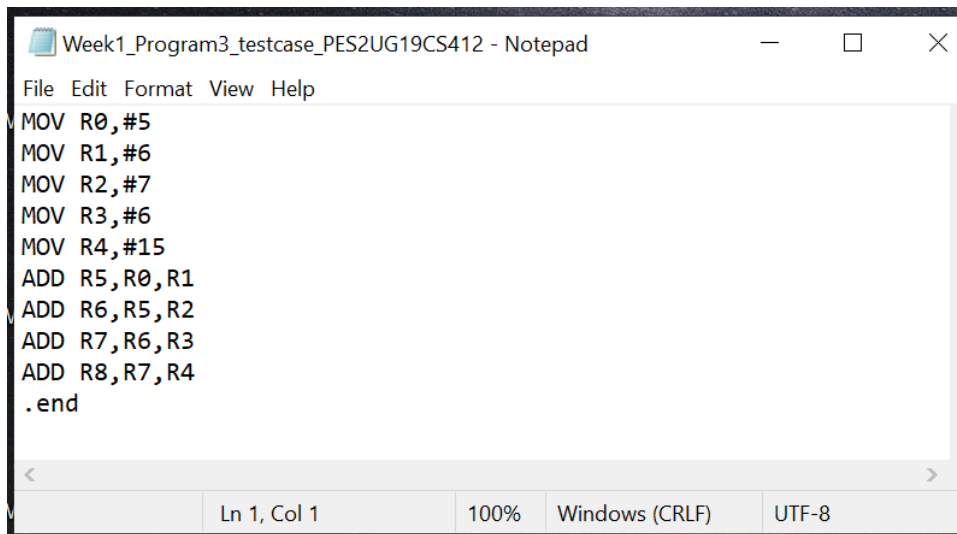
Week# ____1____ Program Number: ____3____

Title of the Program

**Write an ALP to add 5 numbers where values are present
in registers.**

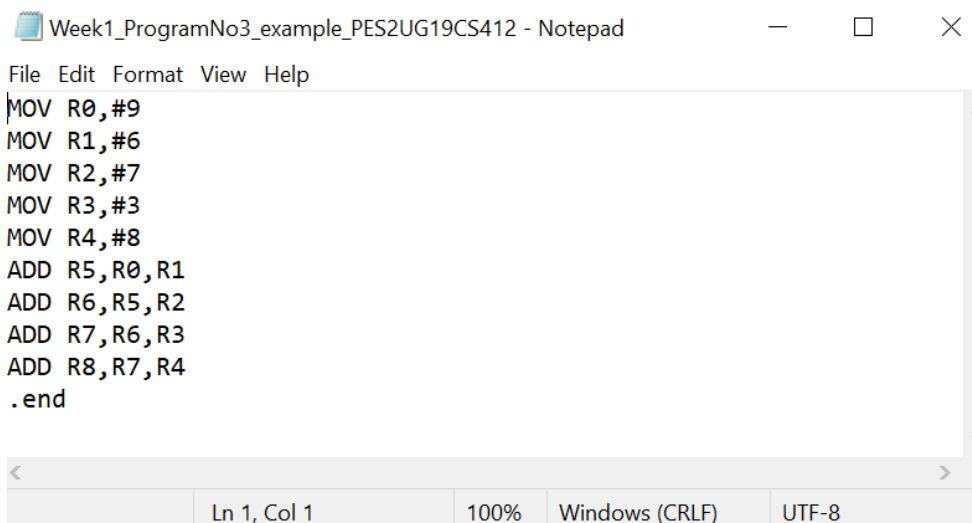
I. ARM Assembly Code for each program

Testcase used in class



```
File Edit Format View Help
MOV R0,#5
MOV R1,#6
MOV R2,#7
MOV R3,#6
MOV R4,#15
ADD R5,R0,R1
ADD R6,R5,R2
ADD R7,R6,R3
ADD R8,R7,R4
.end
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

Own example

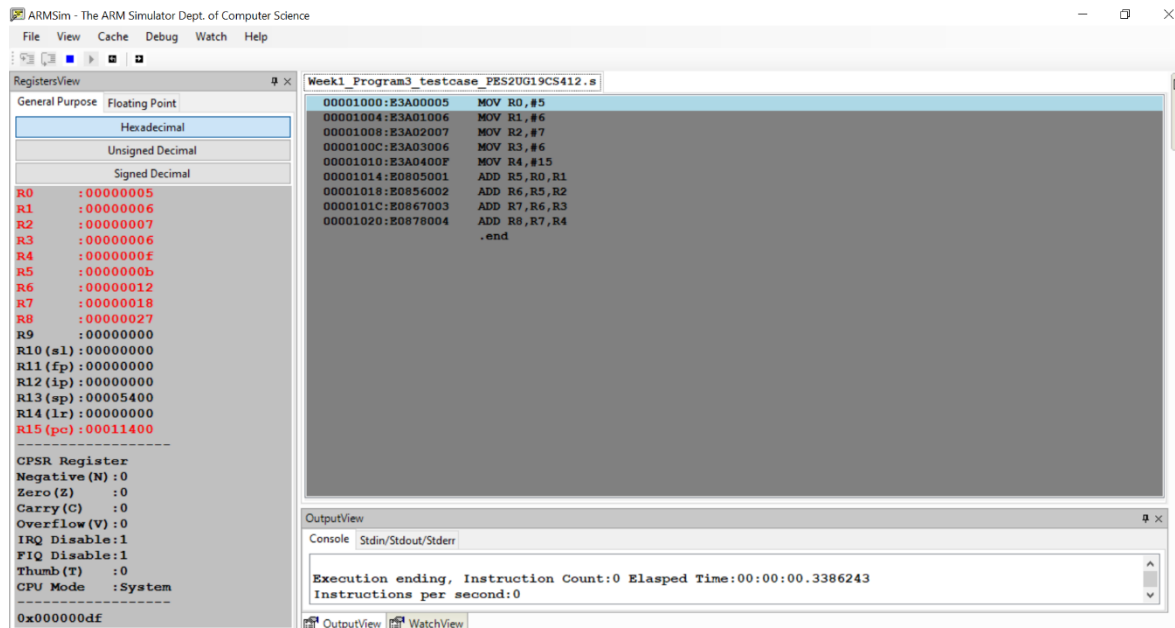


```
File Edit Format View Help
MOV R0,#9
MOV R1,#6
MOV R2,#7
MOV R3,#3
MOV R4,#8
ADD R5,R0,R1
ADD R6,R5,R2
ADD R7,R6,R3
ADD R8,R7,R4
.end
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

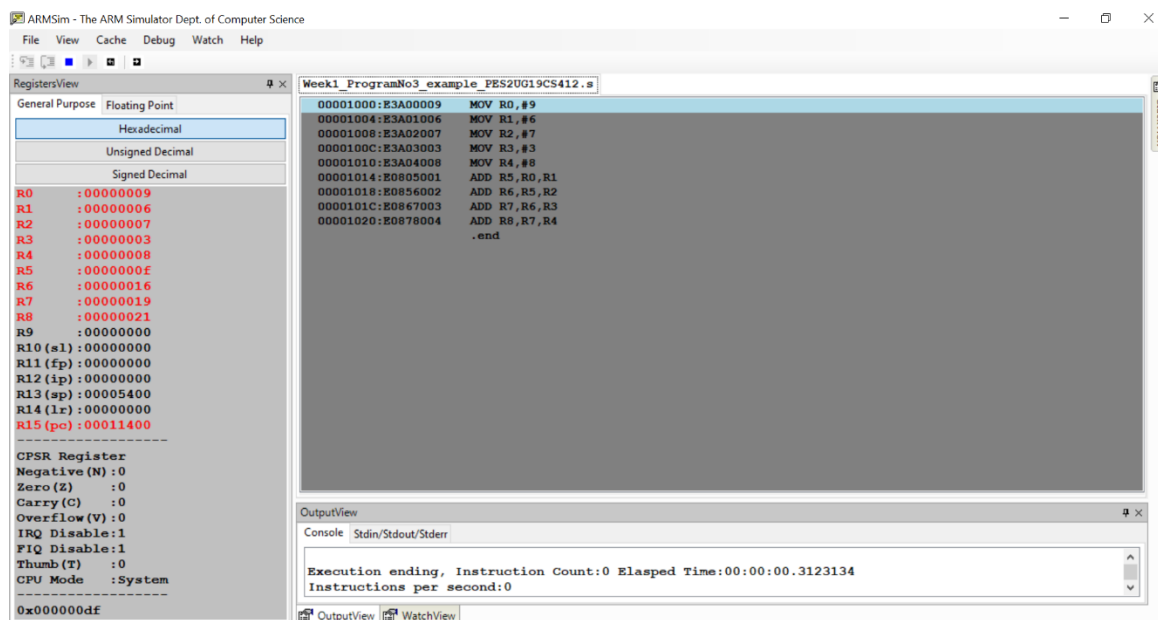
II. Output Screen Shot (Register Window, Output window)

The output should be verified with 2 test cases (one example shown in class, one example of own choice)

Output of testcase



Output of own example



III. Output table for each program

TESTCASE

R0		0x05
R1		0x06
R2		0x07
R3		0x06
R4		0x15
R5	R0+R1	0x0b
R6	R5+R2	0x12
R7	R6+R3	0x18
R8	R7+R4	0x27

Own example

R0		0x09
R1		0x06
R2		0x07
R3		0x03
R4		0x08
R5	R0+R1	0x0f
R6	R5+R2	0x16
R7	R6+R3	0x19
R8	R7+R4	0x21

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:27—01—2021

Name: SUHAN B REVANKAR	SRN: PES2UG19CS412	Section G
------------------------	-----------------------	--------------

Week# 1 Program Number: 4

Title of the Program

Write an ALP using ARM instruction set to check if a number stored in a register is even or odd. If even, store 00 in R0, else store FF in R0

I. ARM Assembly Code for each program

```

Week1_ProgramNo4_PES2UG19CS412 - Notepad
File Edit Format View Help
MOV r1, #6
ANDS r2, r1, #1
BEQ loc1
B loc2
loc1: MOV r0, #0
B p2
loc2: MOV r0, #0xff

p2:
MOV r1, #5
ANDS r2, r1, #1
BEQ loc3
B loc4
loc3: MOV r0, #0
B last
loc4: MOV r0, #0xff
last:
SWI 0x11
.end

```

II. Output Screen Shot (Register Window, Output window)

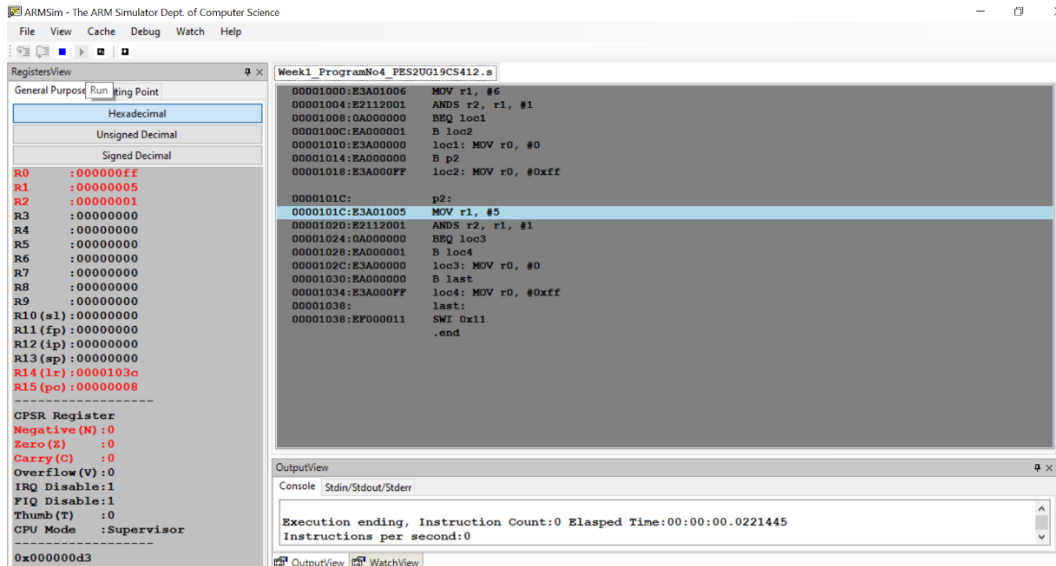
The output should be verified with 2 test cases (one example shown in class, one example of own choice)

The screenshot displays the ARMSim ARM Simulator interface. The main window is titled "Week1_ProgramNo4_PES2UG19CS412.s". The left pane shows the "RegistersView" with a table of registers and their values. The right pane shows the "OutputView" with a console output.

Register	Value
R0	:00000000
R1	:00000006
R2	:00000000
R3	:00000000
R4	:00000000
R5	:00000000
R6	:00000000
R7	:00000000
R8	:00000000
R9	:00000000
R10 (s1)	:00000000
R11 (fp)	:00000000
R12 (ip)	:00000000
R13 (sp)	:00005400
R14 (lr)	:00000000
R15 (pc)	:0000101c

CPUSR Register
 Negative (N) : 0
 Zero (Z) : 1
 Carry (C) : 0
 Overflow (V) : 0
 IRQ Disable : 1
 FIQ Disable : 1
 Thumb (T) : 0
 CPU Mode : System
 0x400000df

OutputView
 Console Stdin/Stdout/Stderr
 Loading assembly language file C:\Users\shash\Desktop\Week1_ProgramNo4_PES2UG19CS412.s



III. Output table for each program

CASE1	R1		0x06
	R2	After AND operation	0x00
	R0	(EVEN)	0x00
CASE2	R1		0x05
	R2	After AND operation	0x01
	R0	(ODD)	0xff

Disclaimer:

- The programs and output submitted is duly written, verified and executed by me.

- I have not copied from any of my peers nor from the external resource such as internet.
- If found plagiarized, I will abide with the disciplinary action of the University.

Signature: suhanb

Name: SUHAN B REVANKAR

SRN: PES2UG19CS412

Section: G

Date: 27—01—2021