

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:26/01/2021

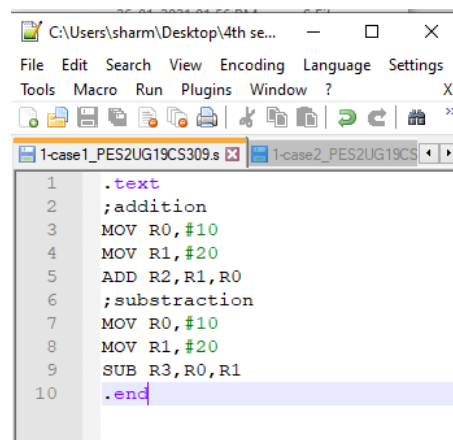
Name: R SHARMILA	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 1 Program Number: 1

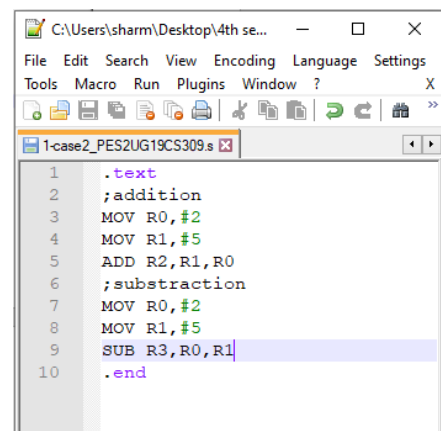
Title of the Program

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



```
1 .text
2 ;addition
3 MOV R0,#10
4 MOV R1,#20
5 ADD R2,R1,R0
6 ;substruction
7 MOV R0,#10
8 MOV R1,#20
9 SUB R3,R0,R1
10 .end
```



```
1 .text
2 ;addition
3 MOV R0,#2
4 MOV R1,#5
5 ADD R2,R1,R0
6 ;substruction
7 MOV R0,#2
8 MOV R1,#5
9 SUB R3,R0,R1
10 .end
```

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

The image displays two screenshots of the ARMSim - The ARM Simulator interface, showing the Register Window and Output Window.

Top Screenshot:

- RegistersView:** Shows the state of registers R0 through R15. R15 (PC) is highlighted in red and shows the value 00001018. The CPSR Register shows Negative (N): 0, Zero (Z): 0, Carry (C): 0, Overflow (V): 0, IRQ Disable: 1, FIQ Disable: 1, Thumb (T): 0, and CPU Mode: System.
- Code Window:** Displays assembly code for 1-case1_PES2UG19CS309.s. The code includes instructions like MOV R0, #10, MOV R1, #20, ADD R2, R1, R0, MOV R0, #10, MOV R1, #20, and SUB R3, R0, R1.
- OutputView:** Shows the console output, which includes the message "Loading assembly language file C:\Users\sharm\Desktop\4th sem\Microprocessor\Laab work\MPCALAB_WEEK1_PES2UG19CS309\1-c".

Bottom Screenshot:

- RegistersView:** Shows the state of registers R0 through R15. R15 (PC) is highlighted in red and shows the value 0000101c. The CPSR Register shows Negative (N): 0, Zero (Z): 0, Carry (C): 0, Overflow (V): 0, IRQ Disable: 1, FIQ Disable: 1, Thumb (T): 0, and CPU Mode: System.
- Code Window:** Displays assembly code for 1-case2_PES2UG19CS309.s. The code includes instructions like MOV R0, #2, MOV R1, #5, ADD R2, R1, R0, MOV R0, #2, MOV R1, #5, and SUB R3, R0, R1.
- OutputView:** Shows the console output, which includes the message "Loading assembly language file C:\Users\sharm\Desktop\4th sem\Microprocessor\Laab work\MPCALAB_WEEK1_PES2UG19CS309\1-c".

III. Output table for each program

Example R0=10=Hex 0A R1=20=Hex 14 After Addition R2=30=Hex 1E After Subtraction R3 = 10 = Hex 0A			
R0	R1	Arithmetic Operation	Result
0x0A	0x14	ADD	R2=0x1E
0x0A	0x14	SUBTRACT	R3=0x0A
Example R0=2=Hex 02 R1=5=Hex 05 After Addition R2=7=Hex 07 After Subtraction R3= 3 = Hex 03			
R0	R1	Arithmetic Operation	Result
0x02	0x05	ADD	R2=0x07
0x02	0x05	SUBTRACT	R3=0x03

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:26/01/2021

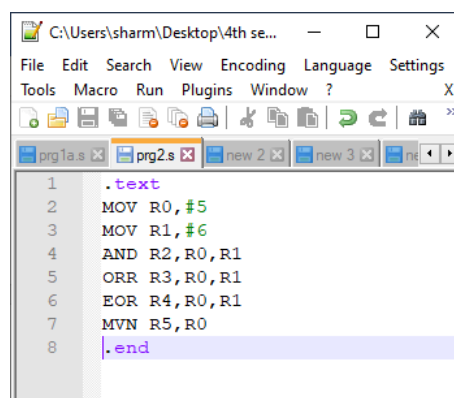
Name: R SHARMILA	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 1 Program Number: 2

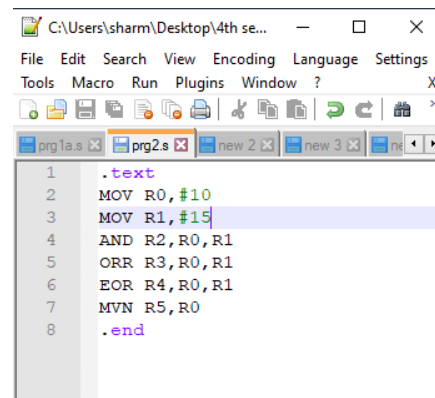
Title of the Program

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

I. ARM Assembly Code for each program

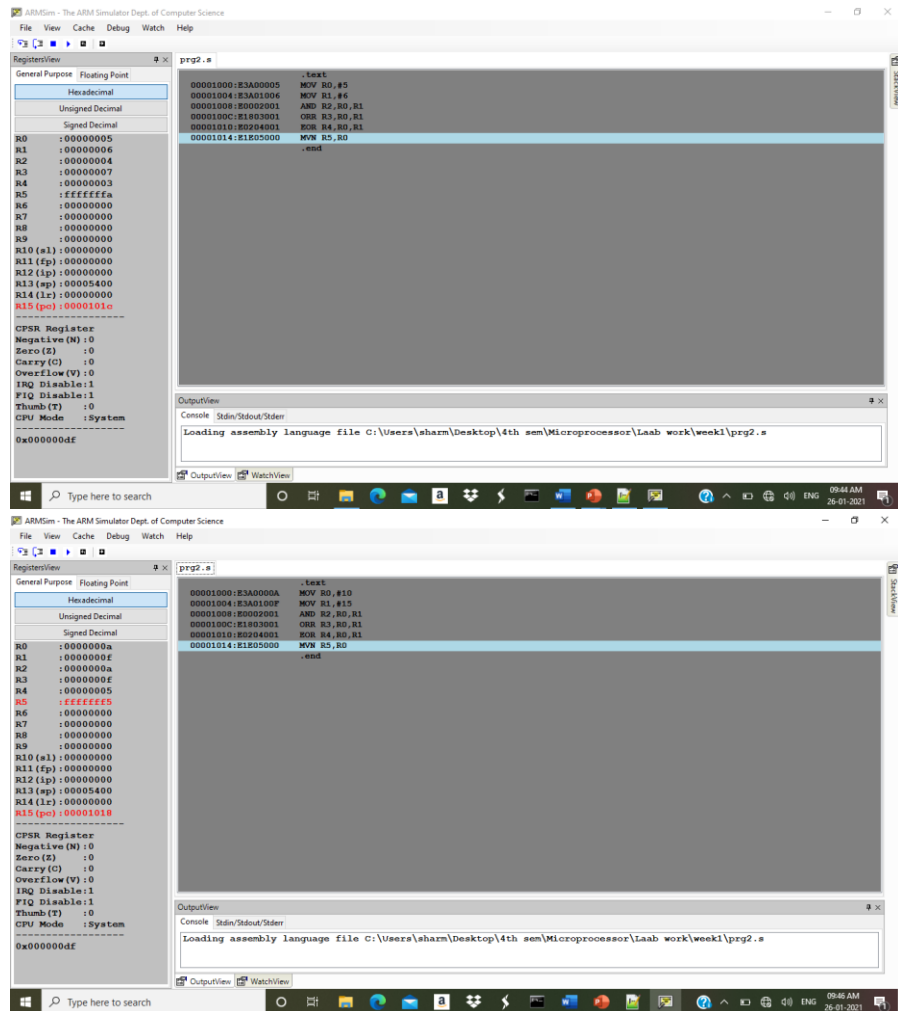


```
1 .text
2 MOV R0, #5
3 MOV R1, #6
4 AND R2, R0, R1
5 ORR R3, R0, R1
6 EOR R4, R0, R1
7 MVN R5, R0
8 .end
```



```
1 .text
2 MOV R0, #10
3 MOV R1, #15
4 AND R2, R0, R1
5 ORR R3, R0, R1
6 EOR R4, R0, R1
7 MVN R5, R0
8 .end
```

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



III. Output table for each program

R0	R1	Logical Operation	Instruction	Result
0x05	0x06	AND	AND	R2 =0x04
0x05	0x06	OR	ORR	R3 =0x07
0x05	0x06	EX-OR	EOR	R4 =0x03
0x05		NOT	MVN	R5 =0xffffffffa

R0	R1	Logical Operation	Instruction	Result
0x0a	0x0f	AND	AND	R2 =0x0a
0x0a	0x0f	OR	ORR	R3 =0x0f
0x0a	0x0f	EX-OR	EOR	R4 =0x05
0x0a		NOT	MVN	R5 =0xffffffff5

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:26/01/2021

Name: R SHARMILA	SRN:PES2UG19CS309	Section E
------------------	-------------------	--------------

Week# 1 Program Number: 3

Title of the Program

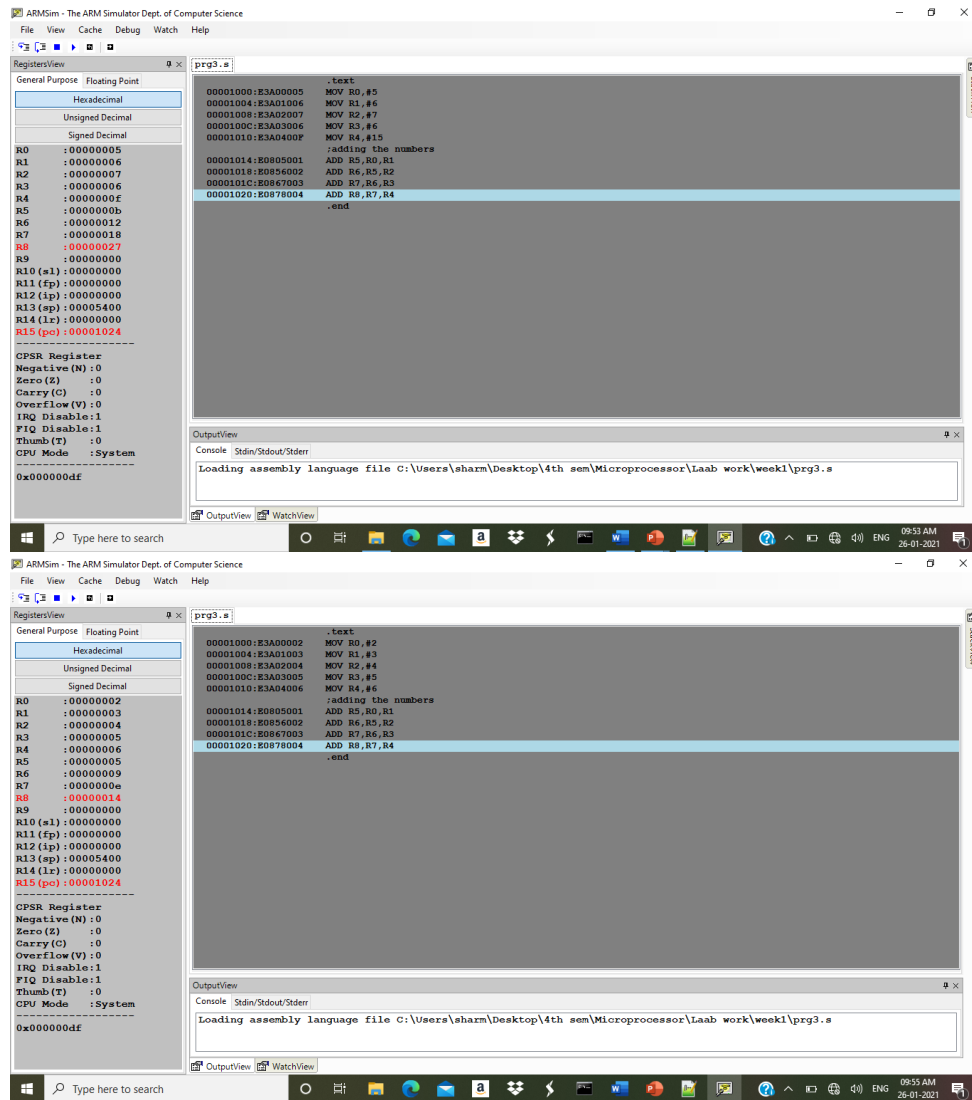
Write an ALP to add 5 numbers where values are present in registers.

I. ARM Assembly Code for each program

```
C:\Users\sharm\Desktop\4th se...
File Edit Search View Encoding Language Settings
Tools Macro Run Plugins Window ? X
prg1a.s prg2.s prg3.s new 3 ne
1 .text
2 MOV R0,#5
3 MOV R1,#6
4 MOV R2,#7
5 MOV R3,#6
6 MOV R4,#15
7 ;adding the numbers
8 ADD R5,R0,R1
9 ADD R6,R5,R2
10 ADD R7,R6,R3
11 ADD R8,R7,R4
12 .end
13
```

```
C:\Users\sharm\Desktop\4th se...
File Edit Search View Encoding Language Settings
Tools Macro Run Plugins Window ? X
prg1a.s prg2.s prg3.s new 3 ne
1 .text
2 MOV R0,#2
3 MOV R1,#3
4 MOV R2,#4
5 MOV R3,#5
6 MOV R4,#6
7 ;adding the numbers
8 ADD R5,R0,R1
9 ADD R6,R5,R2
10 ADD R7,R6,R3
11 ADD R8,R7,R4
12 .end
13
```

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)

III. Output table for each program

Case 1:

R0		0x05
R1		0x06
R2		0x07
R3		0x06
R4		0x0f
R5	R0+R1	0x0b
R6	R5+R2	0x12
R7	R6+R3	0x18
R8	R7+R4	0x27

Case2:

R0		0x02
R1		0x03
R2		0x04
R3		0x05
R4		0x06
R5	R0+R1	0x05
R6	R5+R2	0x09
R7	R6+R3	0x0e
R8	R7+R4	0x14

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date:26/01/2021

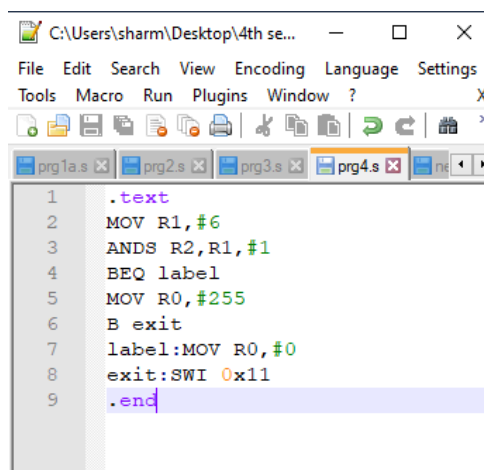
Name: R SHARMILA	SRN: PES2UG19CS309	Section E
------------------	-----------------------	--------------

Week# 1 Program Number: 4

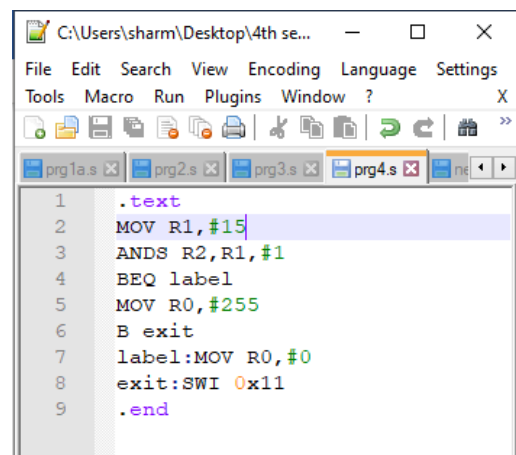
Title of the Program

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

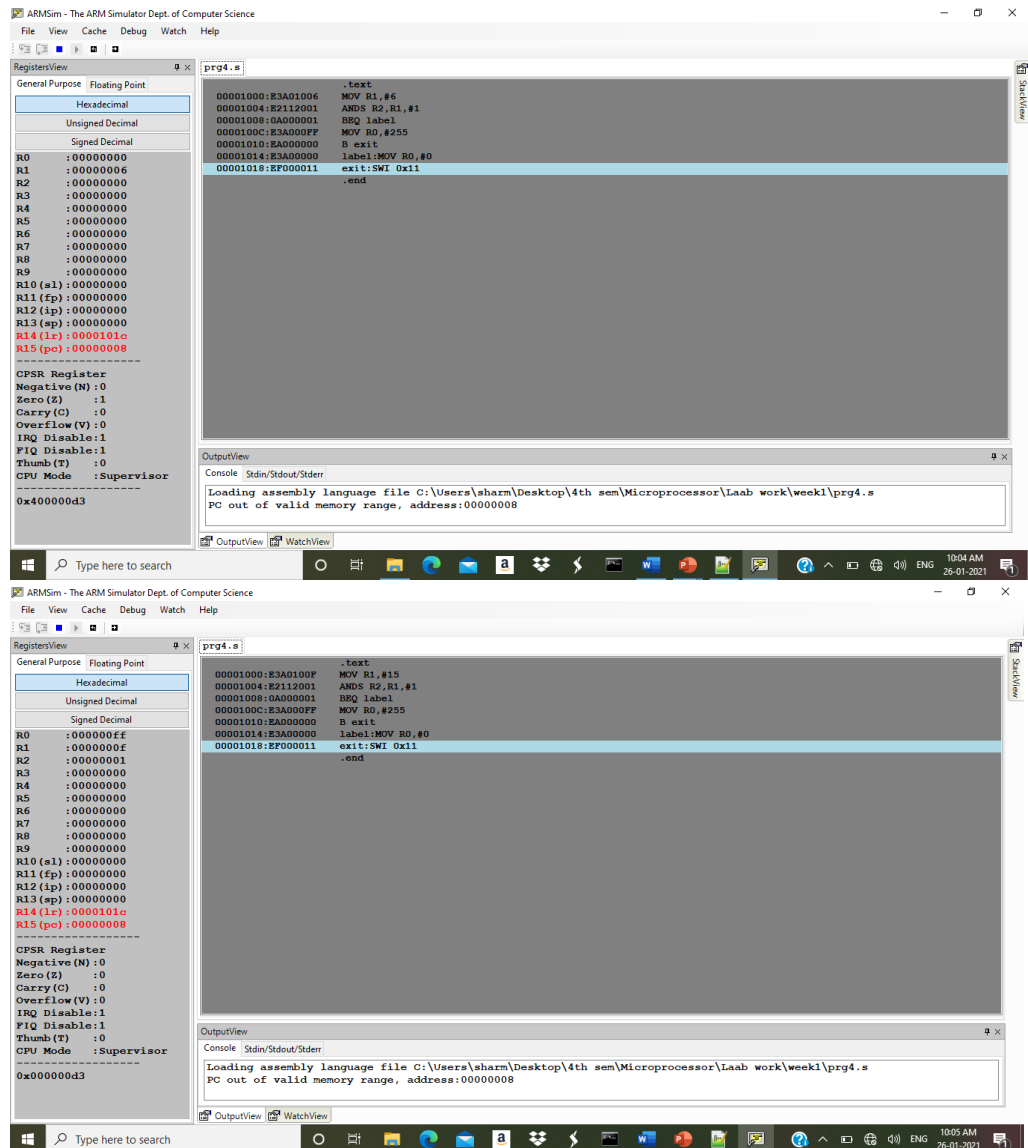


```
1 .text
2 MOV R1, #6
3 ANDS R2, R1, #1
4 BEQ label
5 MOV R0, #255
6 B exit
7 label: MOV R0, #0
8 exit: SWI 0x11
9 .end
```



```
1 .text
2 MOV R1, #15
3 ANDS R2, R1, #1
4 BEQ label
5 MOV R0, #255
6 B exit
7 label: MOV R0, #0
8 exit: SWI 0x11
9 .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)

III. Output table for each program

CASE 1	R1		0x06
	R2	After AND operation	0x00
	R0	(EVEN)	0x00
CASE 2	R1		0x0f
	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: SHARMILA

Name: R SHARMILA

SRN: PES2UG19CS309

Section: E

Date:26/01/2021

