

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

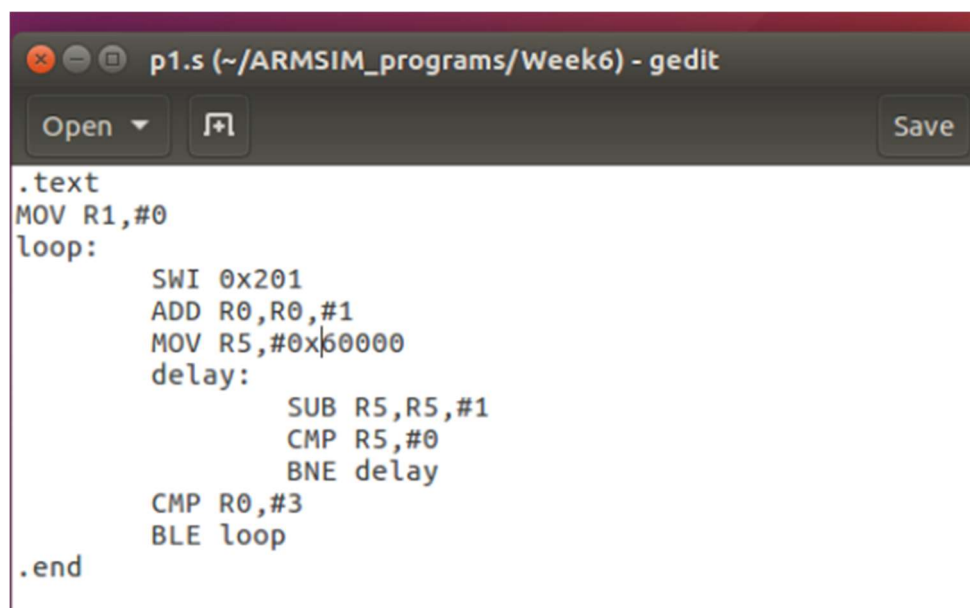
Date: 4/3/2021

Name: B.Pravena	SRN: PES2UG19CS076	Section: B
-----------------	--------------------	------------

Week# 6 Program Number: 1

1. Write an ALP to blink LEDs. First, the right LED is switched on and the left LED is switched off. After 1 second, the right LED is switched off and the left LED is switched on and the program continue to blink both the LEDs.

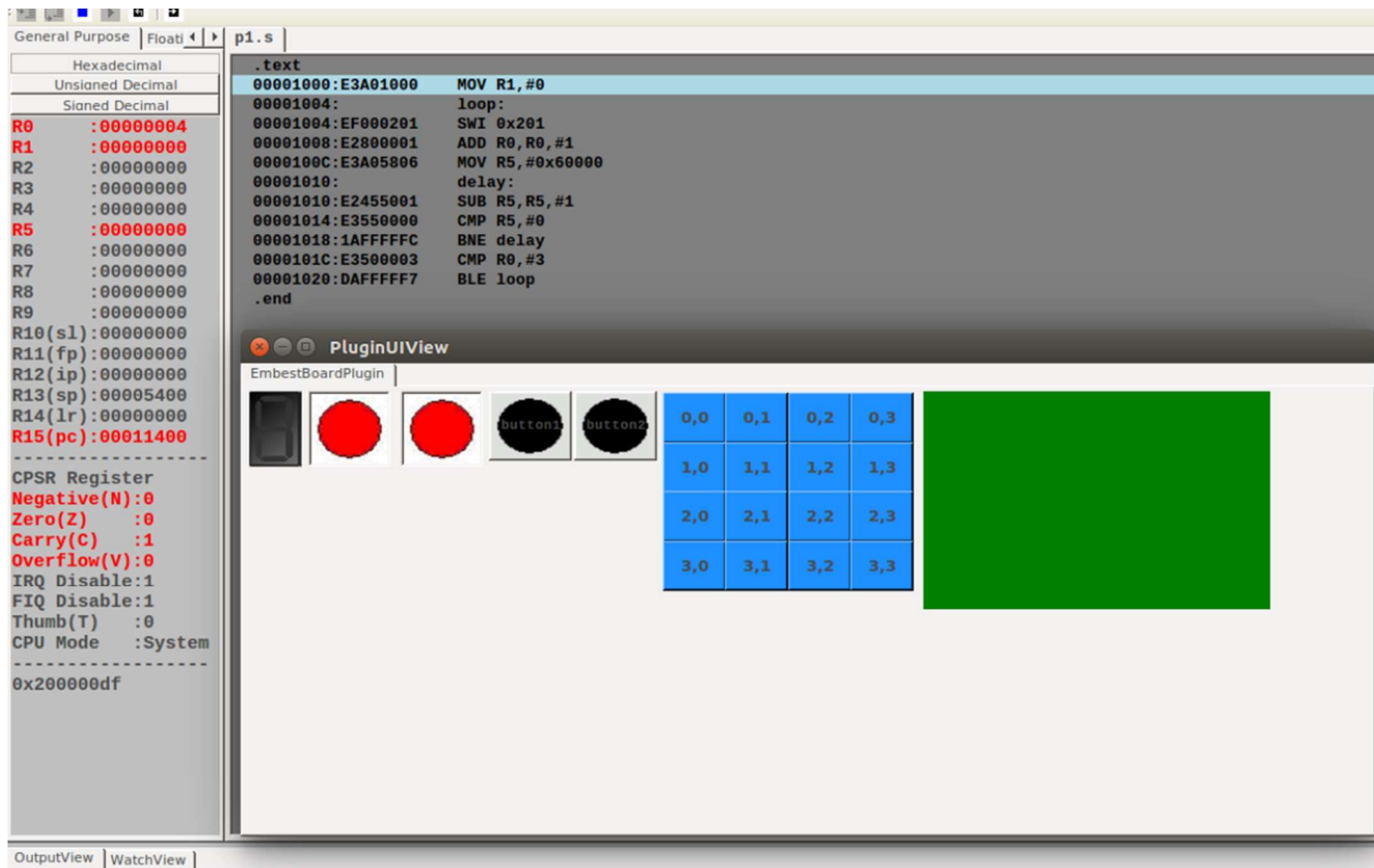
I. ARM Assembly Code



```
p1.s (~/ARMSIM_programs/Week6) - gedit
Open Save

.text
MOV R1,#0
loop:
    SWI 0x201
    ADD R0,R0,#1
    MOV R5,#0x60000
    delay:
        SUB R5,R5,#1
        CMP R5,#0
        BNE delay
    CMP R0,#3
    BLE loop
.end
```

II. Output Screen Shot



Microprocessor and Computer Architecture Laboratory UE19CS256

4th Semester, Academic Year 2020-21

Date: 4/3/2021

Name: B.Pravena	SRN: PES2UG19CS076	Section: B
-----------------	--------------------	------------

Week# 6

Program Number: 2

Write an ALP to display 0-9, A-F (up and down count) on an 8 segment display

I. ARM Assembly Code

```
Week6_Program2_PES2UG19CS076 - Notepad
File Edit Format View Help
|.text
MOV R0,#0
MOV R2,#0
again:
    SWI 0x202
    CMP R0,#1
    BEQ loop1
    CMP R0,#2
    BEQ loop2
    B again

loop1:
    MOV R5,#16
    LDR R1,=zero
back1:
    LDRB R0,[R1]
    SWI 0x200
    BL delay
    ADD R1,R1,#1
    SUB R5,R5,#1
    CMP R5,#0
    BNE again

loop2:
    MOV R5,#6
    LDR R1,=F
    LDRB R0,[R1]
    SWI 0x200
    BL delay
    SUB R1,R1,#1
    SUB R5,R5,#1
    CMP R5,#0
    BNE loop2
B again

delay:
    MOV R4,#0x64000
delay_loop:
    SUB R4,R4,#1
    CMP R4,#0
    BGE delay_loop
    MOV PC,LR

.data
zero: .byte 0b11101101
one: .byte 0b01100000
two: .byte 0b11001110
three: .byte 0b11101010
four: .byte 0b01100011
five: .byte 0b10101011
six: .byte 0b10101111
seven: .byte 0b11100000
eight: .byte 0b11101111
nine: .byte 0b11101011
A: .byte 0b11100111
B: .byte 0b11101111
C: .byte 0b10001101
D: .byte 0b11101101
E: .byte 0b10001111
F: .byte 0b10000111

.end
```

II. Output Screen Shot

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 : 00000000
R1 : 00000000
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) : 00001000

CPSR Register

Negative (N) : 0
Zero (Z) : 0
Carry (C) : 0
Overflow (V) : 0
IRQ Disable : 1

PluginUIView

EmbestBoardPlugin

0.0 0.1 0.2 0.3
1.0 1.1 1.2 1.3

Week6_Program2_PES2UG19CS076.s

```
.text
00001000:E3A00000 MOV R0,#0
00001004:E3A02000 MOV R2,#0
00001008:          again:
00001008:EF000202      SWI 0x202
0000100C:E3500001      CMP R0,#1
00001010:0A000002      BEQ loop1
00001014:E3500002      CMP R0,#2
00001018:0A000009      BEQ loop2
0000101C:EAF0FFF9      B again

00001020:          loop1:
00001020:E3A05010      MOV R5,#16
00001024:E59F1054      LDR R1,=zero
00001028:          back1:
00001028:E5D10000      LDRB R0,[R1]
0000102C:EF000200      SWI 0x200
00001030:EB00000D      BL delay
00001034:E2811001      ADD R1,R1,#1
```

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 : 00000000
R1 : 00000000
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) : 00001000

CPSR Register

Negative (N) : 0
Zero (Z) : 0
Carry (C) : 0
Overflow (V) : 0
IRQ Disable : 1

PluginUIView

EmbestBoardPlugin

0.0 0.1 0.2 0.3
1.0 1.1 1.2 1.3

Week6_Program2_PES2UG19CS076.s

```
.text
00001000:E3A00000 MOV R0,#0
00001004:E3A02000 MOV R2,#0
00001008:          again:
00001008:EF000202      SWI 0x202
0000100C:E3500001      CMP R0,#1
00001010:0A000002      BEQ loop1
00001014:E3500002      CMP R0,#2
00001018:0A000009      BEQ loop2
0000101C:EAF0FFF9      B again

00001020:          loop1:
00001020:E3A05010      MOV R5,#16
00001024:E59F1054      LDR R1,=zero
00001028:          back1:
00001028:E5D10000      LDRB R0,[R1]
0000102C:EF000200      SWI 0x200
00001030:EB00000D      BL delay
00001034:E2811001      ADD R1,R1,#1
```

Microprocessor and Computer Architecture Laboratory

UE19CS256

4th Semester, Academic Year 2020-21

Date: 4/3/2021

Name: B.Pravena	SRN: PES2UG19CS076	Section: B
-----------------	--------------------	------------

Week# ____6____

Program Number: ____3____

Write an ALP to move a string from Right to Left on LCD (40 columns by 15 rows).

I. ARM Assembly Code

```
p3.s (~/ARMSIM_programs/Week6) - gedit
.data
str:.ASCIZ "HELLO WORLD"
num:.WORD 15000
.text
MOV R0,#15
MOV R1,#5
LDR R2,=str
LDR R8,=num
LDR R6,[R8]
MOV R7,#0
loop:
SWI 0x204
BL delay
CMP R0,#0
SUBNE R0,R0,#1
SWIEQ 0x011
B loop
delay:
ADD R7,R7,#1
CMP R7,R6
BNE delay
SWI 0x206
MOV R7,#0
MOV PC,LR
.end
```

II. Output Screen Shot

The screenshot displays a debugger interface with four main panels:

- RegistersView:** Shows the state of 16 registers (R0-R15). All registers are at 0x00000000. Below the registers, the CPSR register is shown with various flags: 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. The PC register is at 0x000000df.
- PluginUI View:** Contains an 'EmbestBoardPlugin' section with a 4x4 grid of buttons labeled 0.0 through 1.3. A green rectangular area on the right displays the text 'HELLO WORL'.
- Assembly View:** Displays the assembly code for 'Week6_Program3_PES2UG19CS076.s'. The code includes data definitions for 'str' and 'num', followed by a series of instructions including MOV, LDR, SWI, BL, CMP, SUBNE, SWIEQ, B, and ADD.
- MemoryView:** Shows a memory dump starting at address 00001050. The first line of memory contains the ASCII string 'HELLO WORLD'.