



**Department of Computer Science &
Engineering
Microprocessor & Computer Architecture -
UE20CS252**

NAME : VISHWAS M

SEC : F

SRN : PES2UG20CS390

DATE: 26/03/2022

WEEK 6

Sl. No

Programs

Week
No.6

- Write a program in ARM7TDMI-ISA to generate a diagonal matrix.
Note: do not read the matrix elements.

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 :4180

R1 :2

R2 :3

R3 :0

R4 :32

R5 :3

R6 :4

R7 :0

R8 :0

R9 :0

R10 (s1) :0

R11 (fp) :0

R12 (ip) :0

R13 (sp) :21504

R14 (lr) :0

R15 (pc) :4156

CPSR Register

Negative (N) :0

Zero (Z) :1

Carry (C) :1

Overflow (V) :0

IRQ Disable:1

FIQ Disable:1

Thumb (T) :0

CPU Mode :Sy

0x600000df

DIAGONAL.S

```

;Write a program in ARM7TDMI-ISA to generate a diagonal matrix.
; Note: do not read the matrix elements.
.DATA
00001054: A: .WORD 0,0,0,0,0,0,0,0,0

.TEXT
00001000:E59F0048 LDR R0,=A ; ARRAY A
; LET THE DIAGONAL VALUE BE 2
00001004:E3A01002 MOV R1,#2
00001008:E3A02000 MOV R2,#0 ; ROW NUMBER
0000100C:E3A03000 MOV R3,#0 ; COL NUMBER
00001010:E3A05003 MOV R5,#3 ; NUMBER OF ELEMENTS IN THE ROW
00001014:E3A06004 MOV R6,#4 ; MULTIPLICATION FACTOR

00001018:E1520003 LOOP: CMP R2,R3
0000101C:0A000007 BEQ ASSIGN
00001020:E2833001 L1: ADD R3,R3,#1
00001024:E3530003 CMP R3,#3
00001028:1AFFFFFA BNE LOOP
0000102C:E3A03000 MOV R3,#0
00001030:E2822001 ADD R2,R2,#1
00001034:E3520003 CMP R2,#3
00001038:1AFFFFF6 BNE LOOP
0000103C:EF000011 SWI 0X011

00001040:E0243592 ASSIGN: MLA R4,R2,R5,R3
00001044:E0040496 MUL R4,R6,R4
00001048:E7801004 STR R1,[R0,R4] ; STORE 2 IN THE DIAGONAL POSITION
0000104C:EAF0FFF3 B L1
00001050:00001054 .END

```

OutputView

Console Stdin/Stdout/Stderr

Loading assembly language file D:\vishwas\sem4\MPCA lab\week6\DIAGONAL.S

Execution starting ...

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.0464812

Instructions per second:0

OutputView WatchView

2. Write a program in ARM7TDMI-ISA to find the sum of all the positive numbers in the array. Use subroutine SUMPOSITIVE for the same.

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView ADDPOSNUM.S

General Purpose Hexadecimal Unsigned Decimal Signed Decimal

R0 : 4208
R1 : 10
R2 : 28
R3 : -10
R4 : 4208
R5 : 0
R6 : 0
R7 : 0
R8 : 0
R9 : 0
R10 (s1) : 0
R11 (fp) : 0
R12 (ip) : 0
R13 (sp) : 21504
R14 (lr) : 4116
R15 (pc) : 4120

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

0x600000df

OutputView

Console Stdin/Stdout/Stderr

Loading assembly language file D:\vishwas\sem4\MPCA lab\week6\ADDPOSNUM.S
Execution starting ...

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.0080830
Instructions per second:0

```
;Write a program in ARM7TDMI-ISA to find the sum of all the positive
;numbers in the array. Use subroutine SUMPOSITIVE for the same.

.DATA
00001048:      A: .WORD 1,-2,3,4,5,-6,7,8,-9,-10
00001070:      SUM: .WORD 0

.TEXT
00001000:E59F0038      LDR R0,=A ; ARRAY
00001004:E3A01000      MOV R1,#0 ; INDEX VARIABLE
00001008:E3A02000      MOV R2,#0 ; SUM VARIABLE
0000100C:E59F4030      LDR R4,=SUM
00001010:EB000001      BL LOOP
00001014:E5842000      STR R2,[R4]
00001018:EF000011      SWI 0X011

0000101C:E4903004      LOOP: LDR R3,[R0],#4
00001020:E1B03003      MOVS R3,R3
00001024:5A000003      BPL POSITIVE
00001028:E2811001      L1: ADD R1,R1,#1
0000102C:E351000A      CMP R1,#10
00001030:1AFFFFF9      BNE LOOP
00001034:E1A0F00E      MOV PC,LR

00001038:E0822003      POSITIVE: ADD R2,R2,R3
0000103C:EAF0FFF9      B L1
```

3. Write a program in ARM7TDMI-ISA to check the parity of given 32 bit number using function subprogram PARITYCHECK. Display appropriate messages as ODD PARITY or EVEN PARITY number.

a)for odd parity:

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 : 4188

R1 : 0

R2 : 4228

R3 : 0

R4 : 3

R5 : 0

R6 : 32

R7 : 0

R8 : 0

R9 : 0

R10 (s1) : 0

R11 (fp) : 0

R12 (ip) : 0

R13 (sp) : 21504

R14 (lr) : 0

R15 (pc) : 4164

CPSR Register

Negative (N) : 0

Zero (Z) : 0

Carry (C) : 1

Overflow (V) : 0

IRQ Disable : 1

FIQ Disable : 1

Thumb (T) : 0

CPU Mode : Sy

0x200000df

PARITYCHECK.S

```
.DATA
0000105C: ODD: .ASCIZ "ODD PARITY NUMBER"
0000106E: EVEN: .ASCIZ "EVEN PARITY NUMBER"
00001084: A: .WORD 0X00000007

.TEXT
00001000:E59F2048 LDR R2,=A
00001004:E5923000 LDR R3,[R2]
00001008:E3A06000 MOV R6,#0 ; COUNT REGISTER
0000100C:E3A04000 MOV R4,#0 ; COUNT NUMBER OF 1s.

00001010:E3130001 L2: TST R3,#1
00001014:1A00000B BNE L1
00001018:E1A030A3 LOOP:MOV R3,R3,LSR #1
0000101C:E2866001 ADD R6,R6,#1
00001020:E3560020 CMP R6,#32
00001024:1AFFFFF9 BNE L2
00001028:E3140001 TST R4,#1
0000102C:0A000002 BEQ L3
00001030:E59F001C LDR R0,=ODD
00001034:EF000002 SWI 0X02
00001038:EAD00001 B EXIT
0000103C:E59F0014 L3: LDR R0,=EVEN
00001040:EF000002 SWI 0X02

00001044:EF000011 EXIT:SWI 0X011

00001048:E2844001 L1:ADD R4,R4,#1
0000104C:EAF0FFF1 B LOOP

.END
```

OutputView

Console Stdin/Stdout/Stderr

ODD PARITY NUMBER

b)for even parity:

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 : 4206

R1 : 0

R2 : 4228

R3 : 0

R4 : 0

R5 : 0

R6 : 32

R7 : 0

R8 : 0

R9 : 0

R10 (s1) : 0

R11 (fp) : 0

R12 (ip) : 0

R13 (sp) : 21504

R14 (lr) : 0

R15 (pc) : 4164

CPSR Register

Negative (N) : 0

Zero (Z) : 1

Carry (C) : 1

Overflow (V) : 0

IRQ Disable : 1

FIQ Disable : 1

Thumb (T) : 0

CPU Mode : Sy

0x600000df

PARITYCHECK.S

```
;Write a program in ARM7TDMI-ISA to check the parity of given 32 bit
;number using function subprogram PARITYCHECK. Display appropriate
;messages as ODD PARITY or EVEN PARITY number.

.DATA
0000105C: ODD: .ASCIZ "ODD PARITY NUMBER"
0000106E: EVEN: .ASCIZ "EVEN PARITY NUMBER"
00001084: A: .WORD 0X00000000

.TEXT
00001000:E59F2048 LDR R2,=A
00001004:E5923000 LDR R3,[R2]
00001008:E3A06000 MOV R6,#0 ; COUNT REGISTER
0000100C:E3A04000 MOV R4,#0 ; COUNT NUMBER OF 1s.

00001010:E3130001 L2: TST R3,#1
00001014:1A00000B BNE L1
00001018:E1A030A3 LOOP:MOV R3,R3,LSR #1
0000101C:E2866001 ADD R6,R6,#1
00001020:E3560020 CMP R6,#32
00001024:1AFFFFF9 BNE L2
00001028:E3140001 TST R4,#1
0000102C:0A000002 BEQ L3
00001030:E59F001C LDR R0,=ODD
00001034:EF000002 SWI 0X02
00001038:EAD00001 B EXIT
0000103C:E59F0014 L3: LDR R0,=EVEN
00001040:EF000002 SWI 0X02

00001044:EF000011 EXIT:SWI 0X011

00001048:E2844001 L1:ADD R4,R4,#1
```

OutputView

Console Stdin/Stdout/Stderr

EVEN PARITY NUMBER

Student exercises:

1. Write a program in ARM7TDMI-ISA to find the sum of all the digits in an 32bit number.

.data

A: .WORD 2312

B: .WORD 0xCCCCCCCD

.text

mov r0,#0

LDR r1,=A

LDR r2,[r1]

ldr r4,=B

LDR r5,[r4]

l1:

CMP r2,#10

BLT L4

MOV r3,r2

B L2

L2:

CMP r3,#10

BLT L3

SUB r3,r3,#10

B L2

L3:

ADD r0,r0,r3

UMULL r6,r2,r2,r5

MOV r2,r2,LSR #3

B l1

L4:

ADD r0,r0,r2

.end

//SUM IS STORED IN R0

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose Floating

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 : 8
R1 : 4176
R2 : 2
R3 : 3
R4 : 4180
R5 : -85899345
R6 : 171798692
R7 : 0
R8 : 0
R9 : 0
R10 (s1) : 0
R11 (fp) : 0
R12 (ip) : 0
R13 (sp) : 21504
R14 (lr) : 0
R15 (pc) : 70656

CPSR Register

Negative (N) : 1

Zero (Z) : 0

Carry (C) : 0

Overflow (V) : 0

IRQ Disable : 1

FIQ Disable : 1

Thumb (T) : 0

CPU Mode : System

0x800000df

add_32_bit_number.s

```
.data
00001050:      A: .WORD 2312
00001054:      B: .WORD 0xCCCCCD

.text
00001000:E3A00000      mov r0,#0
00001004:E59F103C      LDR r1,=A
00001008:E5912000      LDR r2,[r1]
0000100C:E59F4038      ldr r4,=B
00001010:E5945000      LDR r5,[r4]

00001014:      L1:
00001014:E352000A      CMP r2,#10
00001018:BA000009      BLT L4
0000101C:E1A03002      MOV r3,r2
00001020:EAF0FFFF      B L2
00001024:      L2:
00001024:E353000A      CMP r3,#10
00001028:BA000001      BLT L3
0000102C:E243300A      SUB r3,r3,#10
00001030:EAF0FFFF      B L2

00001034:      L3:
00001034:E0800003      ADD r0,r0,r3
00001038:E0826592      UMULL r6,r2,r2,r5
0000103C:E1A021A2      MOV r2,r2,LSR #3
00001040:EAF0FFFF      B L1
00001044:      L4:
00001044:E0800002      ADD r0,r0,r2
00001048:00001050      .end
0000104C:00001054
```

OutputView

Console Stdin/Stdout/Stderr

OutputView

WatchView

2. Write a program in ARM7TDMI-ISA to find the number of occurrences of a given character in a string.

Example: Given string : My name is Bond.

Character : 'n'.

Expected Output : Display 2 in a register.

;R7 HOLDS THE NUMBER OF OCCURENCES

.DATA

A: .asciz "MY NAME IS BOND"

B: .asciz "N"

.TEXT

LDR R1, =A

LDR R2, =B

MOV R7, #0

L1: LDRB R3, [R1]

LDRB R4, [R2]

CMP R3, R4

BEQ L2

ADD R1, R1, #1

CMP R3, #0

BNE L1

SWI 0X11

L2: ADD R7, R7, #1

ADD R1, R1, #1

B L1

ARMSim - The ARM Simulator Dept. of Computer Science

File View Cache Debug Watch Help

RegistersView

General Purpose Floati

Hexadecimal

Unsigned Decimal

Signed Decimal

R0 : 0

R1 : 4176

R2 : 4176

R3 : 0

R4 : 78

R5 : 0

R6 : 0

R7 : 2

R8 : 0

R9 : 0

R10(s1): 0

R11(fp): 0

R12(ip): 0

R13(sp): 21504

R14(lr): 0

R15(po): 4136

CPSR Register

Negative(N): 0

Zero(Z): 1

Carry(C): 1

Overflow(V): 0

IRQ Disable: 1

FIQ Disable: 1

Thumb(T): 0

CPU Mode : System

0x600000df

OutputView

Console Stdin/Stdout/Stderr

Loading assembly language file D:\vishwas\sem4\MPCA lab\week6\occurence.s

Execution starting ...

Execution ending, Instruction Count:0 Elapsed Time:00:00:00.0050002

Instructions per second:0

OutputView WatchView

- Step 1: Get number by user
- Step 2: Get the modulus/remainder of the number
- Step 3: sum the remainder of the number
- Step 4: Divide the number by 10
- Step 5: Repeat the step 2 while number is greater than 0.

udiv r0, r6, r7 @ no, div to get quotient

operand1 MOD operand2

--	--

MPCA-Laboratory/Assignment/Hands-on/Project

--