**Experiment 1.b: Modes of Addressing**

**Aim:**

Addition of two values with immediate, direct and indirect mode of addressing.

**Tool Used:**

Keil uVision4

**Theory:**

**Immediate Addressing:**Used in cases where we want to load a constant value into an address. Example: MOV R0, #10

**Register Direct Addressing:**Used in cases where you want to move data between two registers. Example: MOV R0, R1

**Register Indirect Addressing:**Used in cases where you want to load data from an address stored in a register. Example: LDR R0, [R1]

**Code:**

AREA PROGRAM, CODE, READONLY

ENTRY

MAIN

 LDR R0, =0x00120011

 LDR R1, =0x00000003

 LDR R2, VALUE1 ;00120011

 LDR R3, VALUE2 ;00000003

 LDR R4, MEM1 ;mem location of R2

 LDR R5, MEM2 ;mem location of R3

 LDR R6, [R4] ;R6 is loaded with data on memory location pointer by R4:=(R2)

 LDR R7, [R5] ;R7 is loaded with data on memory location pointer by R5:=(R3)

 ADD R8, R0, R1 ;immediate addressing

 ADD R9, R2, R3 ;direct addressing

 ADD R10, R6, R7 ;indirect addressing

 LDR R11, =0x00000008 ;mem location to store the result of R11

 STR R10, [R11] ;store value of R10 to mem location pointed by R11

 AREA PROGRAM, DATA, READONLY

VALUE1 DCD &00120011 ; DCD = Define Constant Double word

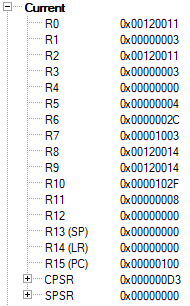
VALUE2 DCD &00000003

MEM1 DCD &00000000 ;memory address of 1st location

MEM2 DCD &00000004 ;memory address of 2nd location

 END

**Register Output:**

****

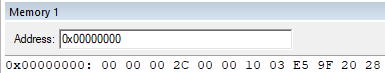
For Inputs at R0 and R1 the input is fed using immediate addressing and the output value is stored at R8.

For Inputs at R2 and R3 the input is loaded using direct addressing and the output is stored at R9.

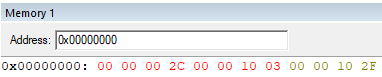
The value at R6 and R7 is changed at the memory locations and loaded using indirect addressing. The output is stored at R10 and also stored back at memory location using indirect addressing.

**Indirect Addressing Result:**

Before Running

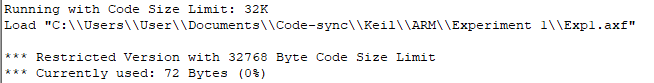
****

After Running

****

The result 0000102F is visible at the 3rd memory location.

**Outputs:**

****

**Result:**

The values are verified after addition and found to be correct on all the three modes of addressing.