



Department of Computer Science & Engineering
Microprocessor & Computer Architecture
MPCA-Laboratory/Assignment/Hands-on/Project
UE21CS251B
Course Information

Sl. No	Lab Programs
Week No.1	<p>1. Sample programs</p> <p>a. Write an ALP using ARM instruction set to add and subtract two 32 bit numbers .Both numbers are in registers.</p> <p>b. Write an ALP to demonstrate logical operations. All operands are in registers.</p> <p>c. Write an ALP to add 5 numbers where values are present in registers.</p> <p>2. Assignment:</p> <p>d. Write an ALP using ARM instruction set to check if a number stored in a register is even or odd.</p> <p>e. Write an ALP to compare the value of R0 and R1, add if R0 = R1, else subtract</p> <p>f. Based on the value of the number in R0, Write an ALP to store 1 in R1 if R0 is zero, Store 2 in R1 if R0 is positive, Store 3 in R1 if R0 is negative.</p>
Week No. 2	<p>g. Write an ALP to add two 32 bit numbers loaded from memory and store the result in memory</p> <p>h. Write an ALP to add two 16 bit numbers loaded from memory and store the result in memory.</p> <p>i. Write an ALP to copy n numbers from Location A to Location B.</p> <p>j. Write an ALP to find the factorial of a number stored in R0. Store the value in R1 (without using LDR and STR instructions)</p> <p>k. Write an ALP to find GCD of two numbers (without using LDR and STR instructions).Both numbers are in registers</p> <p>l. Write an ALP to find the GCD of given numbers (both numbers in memory).Store result in memory.</p> <p>m. Write an ALP to add an array of ten 32 bit numbers from memory.</p> <p>n. Add array of ten 16 bit numbers taking data from memory location stored as byte data (use .byte to store the data instead of .word)</p> <p>o. Check whether a given number is present in array using Linear Search(Without SWI 0x02), if found move its position to R7 else move -1 to R6 (if number not found)</p>
Week No 3.	<p>p. Generate Fibonacci Series and store them in an array.</p> <p>q. Write an ALP to find smallest number in an array of n 32 bit numbers</p> <p>r. A) To perform Convolution using MUL instruction (Addition of multiplication of respective numbers of loc A and loc B)</p> <p>s. B) To perform Convolution using MLA instruction (Addition of multiplication of respective numbers of loc A and loc B).</p>

	<p>t. Write an ALP to find mul (add(a,b),c)</p> <p>u. Write an ALP to find factorial using subroutine</p> <p>v. Write an ALP to perform multiplication using shift method (without using MUL)</p> <p>w.</p> <p>x.</p>
Week No 4.	<p>y. Write an ALP to add two 64 bit numbers loaded from memory and store the result in memory.</p> <p>z. Write an ALP to find 1's and 2's complement of a 32 bit number</p> <p>aa. Write an ALP to scan a 32 bit number if it is negative or positive</p> <p>bb. Write an ALP to find the number of zeroes, positive and negative numbers in a given array</p> <p>cc. Write an ALP to count the number of 1's and 0's in a given 32 bit number.</p> <p>dd. Write an ALP to check the given number has odd or even number of 1's and display the result. (Even Parity and Odd Parity)</p>
Week No 5.	<p>ee. Write an ALP to implement $B=a[i][j]$</p> <p>ff. Write an ALP to implement $C[k]=a[i]+b[j]$</p> <p>gg. Write an ALP to implement $Sum[i]+=a[i][j]$</p> <p>hh. Write an ALP to implement $C[i][j]=a[i][j]+b[i][j]$</p>
Week No 6.	<p>ii. Write an ALP to find the length of a given string</p> <p>jj. Write an ALP to copy string from one location to another</p> <p>kk. Write an ALP to find whether a given character is present in a string.</p> <p>ll. Write an ALP to find how many times a given character is present in a string</p>
mm.	Hardware Programming
nn.	Hardware Programming
oo.	Hardware Programming
pp.	Pipelining
qq.	Pipelining
rr.	Hazards and their Solutions