

## LAB 5: Addressing Modes

### Objectives:

- a) Understand the seven major modes of addressing.
- b) Understand why there are various addressing modes.
- c) Investigate various addressing modes using Frances-A

1) Background: In this lab you will investigate the various ways that operands are accessed. The address field(s) given as operands in the assembly instructions represents various techniques for addressing operands. There are seven common techniques for doing this. We will list each here and use Frances-A to examine how some of these addressing modes are used in a real assembly language.

- i) Immediate addressing mode has the actual operand in the instruction. There is no memory access required. For example, the instruction `add $0x10, %esp` adds the value 16 (hexadecimal10) to the value in the `esp` register. The `$0x10` is the immediate value in the instruction.
- ii) Direct addressing mode has the address of the operand in the instruction. This address is known as the *effective address*.
- iii) Indirect addressing mode has the address of the memory location which contains the address of the operand in the instruction. This is done to allow for a larger address and thus larger range of memory locations.
- iv) Register addressing mode has a register in the instruction which contains the operand.
- v) Register Indirect addressing mode provides a register which contains the address of the operand.
- vi) Displacement addressing mode provides an address and a displacement value which, when added together gives the address of the operand. There are a variety of ways to implement this technique. It could be a register and a value, or an address and register that contains the displacement value, or two registers.
- vii) Stack addressing mode uses the top of the stack implicitly for the location of the operand.

2) Exercises:

- a) Go to the Frances-A website and compile the following program.

```
int main(){
    int A[5] = { 1, 2, 3, 4,5}, i, k=1;
    for (i=0;i<5;++i)
        { A[i] = 7 + k;
          k = k*2;
        }
}
```

- i) Each instruction has one or two operands. Write each assembly instruction and next to it write the addressing mode of each of its operands.

- b) Go to the Frances-A website and compile the default program.

```
int main(){
    int A[5] = {1, 2, 3, 4,5}, i, k=1;
    int *ip;
    ip = A;
    for (i=0;i<5;++i)
        { *(ip+i) = 7 + k;
          k = k*2;
        }
}
```

- i) Each instruction has one or two operands. For the lines beginning with the assembly line implementing  $ip = A$  and ending with the end of the loop body, write the assembly instruction and next to it write the addressing mode of each of its operands.