COMP 228
Practice Probems 3

Question 1:

Consider the following basic single bus CPU internal organization.

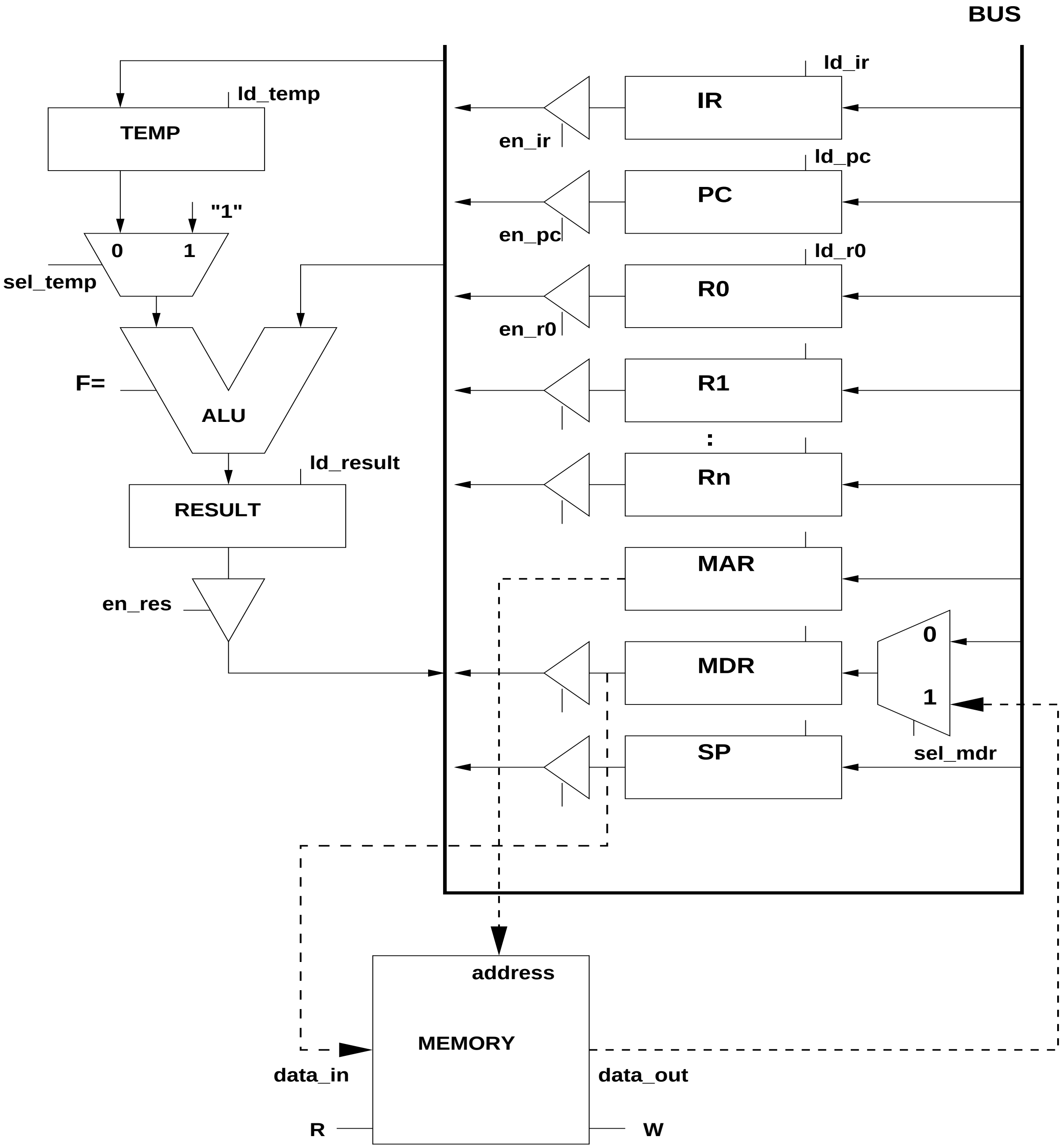

Figure 1: Basic single bus CPU internal organization.

Consider the following three-address instruction:

MAC source1, source2, source3/destination

This instruction (called “multiply and accumulate”) will perform the operation:

source3/destination <--- (source1 x source 2) + source3/destination

In other words, the MAC instruction multiplies the two specified source operands and adds the result of the mul;tiplication to the third source which also acts as the final destination.

(a) What advantage does the multiplexer input of the input of MDR offer in terms of program exectution speed?

(b) Write the sequence of control steps needed to peform an instruction FETCH given that the MDR multiplexer is NOT present (i.e. the MDR is loaded directly from the bus). What modifications would be required to the existing bus organization to accommodate this change?

(c) Give the sequence of control steps to EXECUTE the:

MAC #NUM, (R0)+, R1 instruction.

You may assume that this instruction occupies 1 location in main memory and you may also assume that the MDR multiplexer is present (i.e use the bus organization in Figure 1). You may also assume that the ALU is capable of performing a multiply of it’s two input operands.

Question 2:

Consider the 2-bus internal CPU organization shown in Figure 2. All data transfers between the two buses go through the arithmetic and logic unit (ALU). Among the usual arithmetic and logical operations, the ALU has the added capability of performing the following operations:

F = A	F = B
F = A + 1	F = B + 1
F = A - 1	F = B - 1

where A and B are the ALU inputs and F is the ALU output. Write the sequence of control steps required to execute the instruction JUMP TO SUBROUTINE (JSR). In this particular machine, which is word-addressable, the JSR instruction occupies two words. The first word is the opcode, and the second word contains the starting address of the subroutine. The JSR instruction saves the return address onto the stack, pointed to by the stack pointer SP. It then loads the program counter (PC) with the address specified in the second word of the JSR instruction. The stack GROWS FROM HIGH MEMORY LOCATIONS TOWARDS LOW MEMORY LOCATIONS. Assume that all registers are of the edge-triggered variety, hence data may be transferred from any register R through the ALU and back into the register R in one control step.

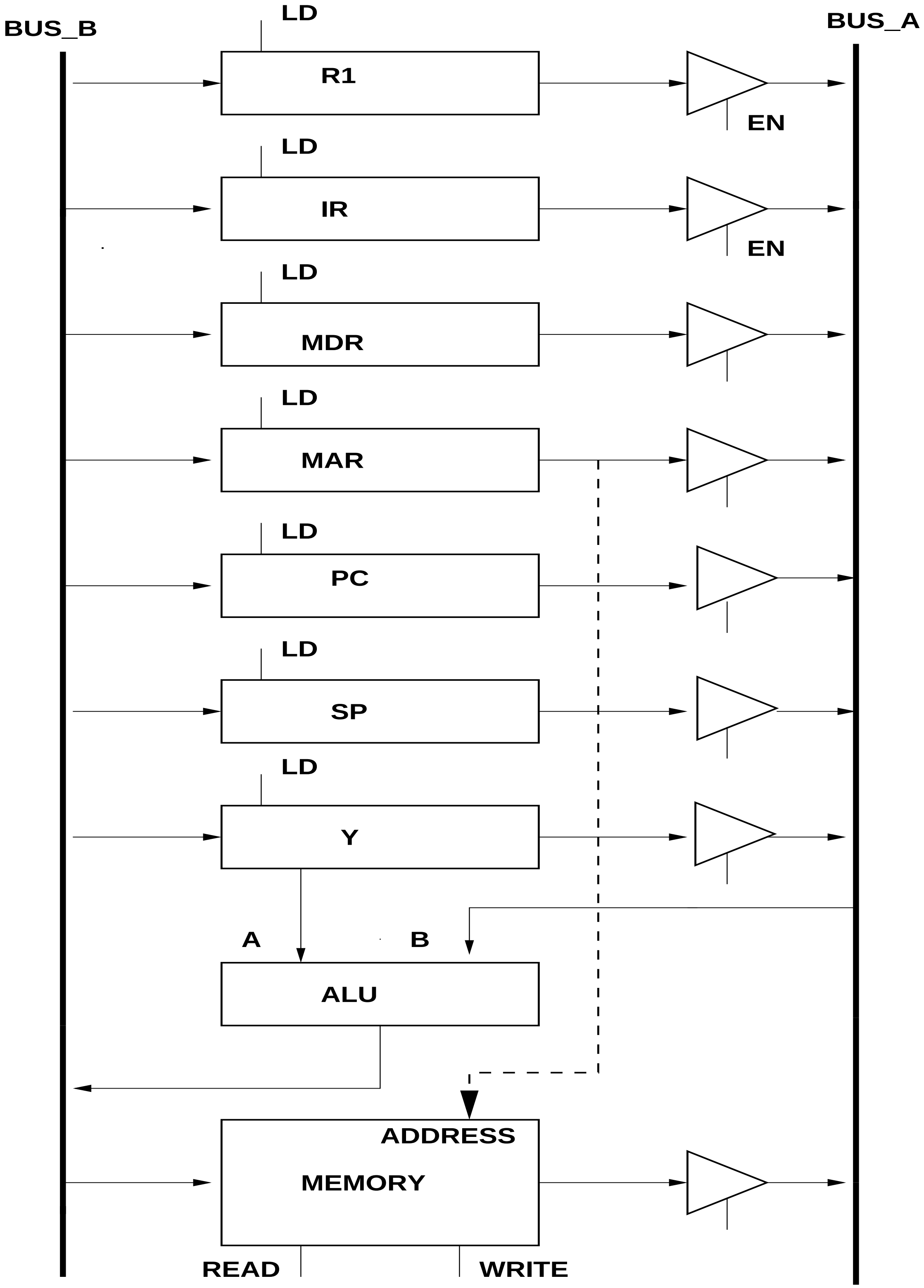

Figure 2: 2-bus CPU organization.

Question 3:

For the 3-bus CPU organization shown in Figure 3, given the sequence of control steps in register transfer notation neeeded to execute the following instruction : POP R0

Assume that the stack grows from high addresses to low-addresses (for a PUSH, therefore for a POP to SP register should first be INCREMENTED). The steps are:

1. increment SP
2. copy SP into MAR
3. perform a memory READ
4. copy MDR into R0.

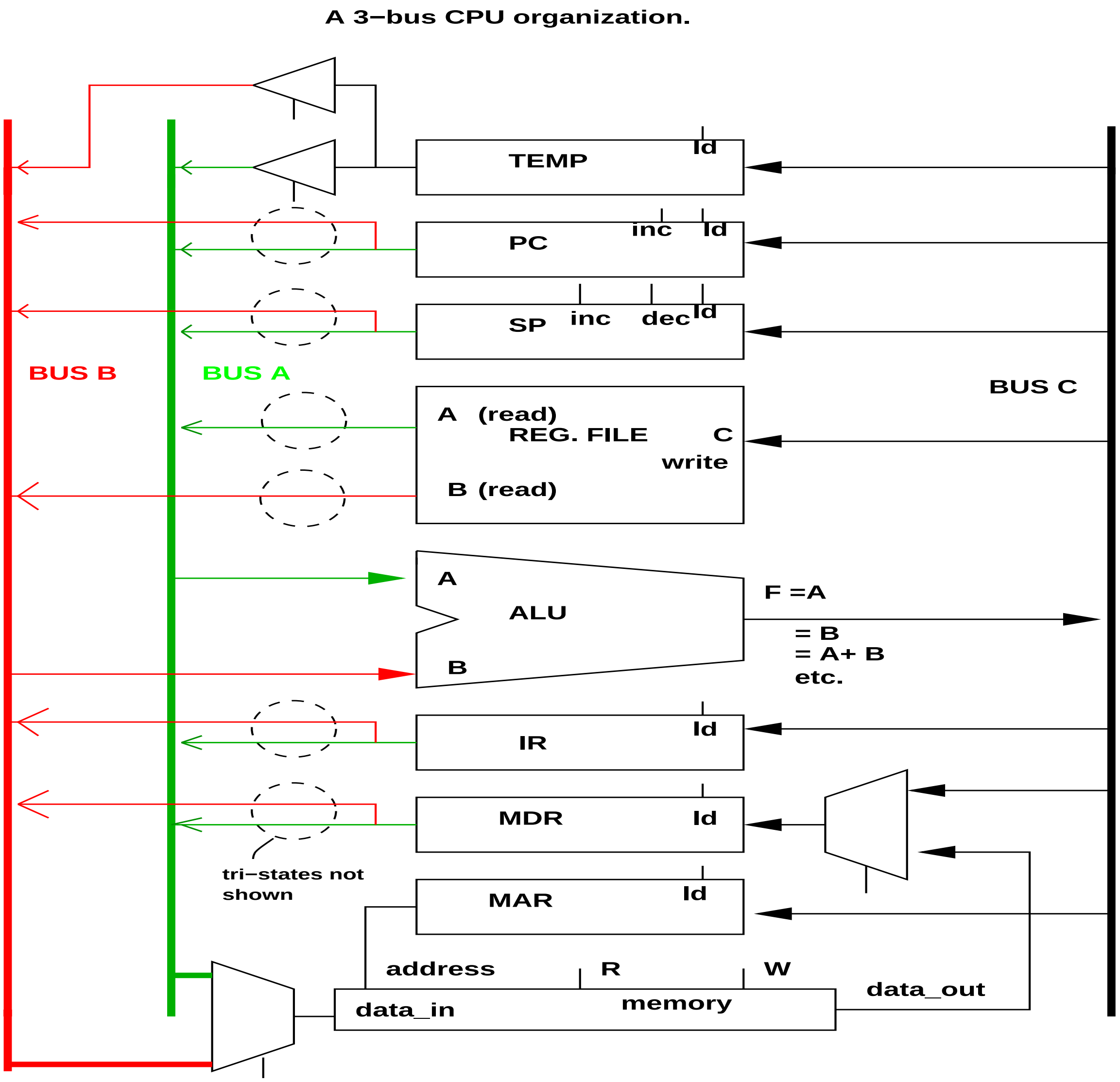

Figure 3: 3-bus CPU organization.