İTÜ Computer Engineering Department **Number:**
28.03.2013 **Name Surname:**
Time: 1 hour 45 minutes **Signature:**

COMPUTER ARCHITECTURE 1ST MIDTERM

QUESTION 1: (35 Points)

A CPU has an instruction pipeline with the following 4 segments:

1. FI (*Fetch Instruction)*
2. DA (*Decode, Address*)
3. FO (*Fetch Operand*)
4. EX (*Execution*)

The clock cycle of the pipeline is 10 ns.

Without the pipeline, one machine-language instruction would have been completed in 30 ns.

a) How long does it take to run a program of 100 instructions by this CPU with the pipeline without considering pipeline hazards?
Calculate the speedup provided by the pipeline for 100 instructions.

b) How much (clock cycles) is the branch penalty in this system if a misprediction occurs? Why?

c) Assume that there is a loop with 10 instructions that contains only one conditional branch instruction (1 branch + 9 other instructions). This loop runs 100 times.
The system uses the static branch prediction strategy “Always predict NOT taken”.
Assume that the ratio of taken branches is 70% and for not taken branches 30%.
How long does it take to run this program (100 x loops)?
Calculate the speedup obtained for this program.

Solution:

İTÜ Computer Engineering Department **Number:**
28.03.2013 **Name Surname:**
Time: 1 hour 45 minutes **Signature:**

COMPUTER ARCHITECTURE 1ST MIDTERM

QUESTION 2: (30 Points)

A RISC CPU has an instruction pipeline with 3 stages: I (Fetch), A (Decode/ALU), D (Data).

The given program on the right performs a multiplication (C=AxB) and clears the registers that hold the source operands A and B.

A starts in memory at the address $500, B at $502, and C at $504.

Remember R0 of the CPU always contains zero.

```
START:
      LDSU  (R0)$500, R10    ; R10 <-- A (16-bit)
      ADD   R0, 0, R11       ; R11 <-- 0
      LDSU  (R0)$502, R11    ; R11 <-- B (16-bit)
      ADD   R0, 0, R12       ; R12 <-- 0 (C)
LOOP:
      ADD   R10, R12, R12    ; R12 <--R12+A
      SUB   R11, 1, R11      ; Decrement R11
      JMPR  BHI, LOOP        ; If higher than zero
      ADD   R0,R0,R10        ;R10 <-- 0 (Clear A)
      STL   (R0)$504,R12     ;M[504]<-- Result (C)
```

a) Examine the given program by drawing the timing diagram of the 3-stage pipeline.
Is there any data conflict?

b) Explain the control (branch) conflict encountered in the given program. What happens if this branch conflict is not solved?

c) Find a software solution to the branch problem without decreasing the performance of the pipeline. Do not change the algorithm.

INSTRUCTION SET:

ADD	Rs,S2,Rd	*Rd* ← *Rs* + S2	
SUB	Rs,S2,Rd	*Rd* ← *Rs* - S2	
LDSU	(Rs)S2,Rd	*Rd*←*M[Rs + S2]*	Short unsigned
STL	(Rs)S2, Rm	*M[Rs + S2]* ← *Rm*	Store long
JMPR	COND,Y	*PC*←*PC + Y*	Jump relative

Solution:

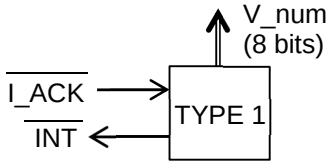
İTÜ Computer Engineering Department **Number:**
28.03.2013 **Name Surname:**
Time: 1 hour 45 minutes **Signature:**

COMPUTER ARCHITECTURE 1ST MIDTERM

QUESTION 3: (35 Points)

In a MC68000-based system there are 3 interrupting devices of two different types.

TYPE 1: To send an interrupt request it asserts the INT′ (active low) output. When its I_ACK′ input becomes zero it puts its vector number on the V_num lines. The maximum delay between I_ACK′ and V_num is 20 ns.
When the I_ACK′ input is one the V_num lines are in 3rd state (high impedance).



TYPE 2: This type of device has all lines of TYPE 1 device (INT′ , I_ACK′ , V_num). In addition it has an V_ACK′ output. When its I_ACK′ input becomes zero it puts its vector number on the V_num lines and asserts the V_ACK′ output (active low). When the I_ACK′ input becomes one the V_ACK′ output is deactivated and the V_num lines transit to the 3rd state (high impedance).
Note that the delay between I_ACK′ and V_num is not determined for this type.

An interrupting device (A) of TYPE 1 and two devices (B, C) of TYPE 2 will be connected to the 2nd interrupt level of the MC6800 using a serial interrupt priority controller (*daisy chain*). A has the highest priority and C the lowest. Assume that each logic gate has the propagation delay of 5 ns.

a) Design and draw the internal structure only of a single link of the daisy chain that can be used with the given interrupting devices.

b) Design and draw the entire system. Do not show the internal structure of a link of the daisy chain.

c) What happens if the device A sends an interrupt request while the interrupt service routine (ISR) of the device C is running.