

Department of Electrical and Computer Engineering  
The University of Texas at Austin

EE 460N Fall 2018  
Y. N. Patt, Instructor  
Chirag Sakhuja, John MacKay, Aniket Deshmukh, Mohammad Behnia, TAs  
Exam 2  
November 19, 2018

Name: \_\_\_\_\_

Problem 1 (20 points): \_\_\_\_\_

Problem 2 (25 points): \_\_\_\_\_

Problem 3 (25 points): \_\_\_\_\_

Problem 4 (30 points): \_\_\_\_\_

Total (100 points): \_\_\_\_\_

Note: Please be sure that your answers to all questions (and all supporting work that is required) are contained in the space provided.

Note: Please be sure your name is recorded on each sheet of the exam.

Please read the following sentence, and if you agree, sign where requested: I have not given nor received any unauthorized help on this exam.

Signature: \_\_\_\_\_

**GOOD LUCK!**

Name: \_\_\_\_\_

**Problem 1 (20 points):** Answer the following questions.

**Part a (5 points):** A 16KB, 2-way set associative, physically-indexed, physically-tagged cache has a line size of 64B. We wish to use it with x86 ISA, which has a page size of 4KB. Assuming no help from the Operating System, can we design the cache such that the TLB, Tag Store, and Data Store accesses can all be made at the same time?

Yes/No (Circle one).

Explain.

**Part b (5 points):** A computer implements IEEE Floating Point, with the one exception that each data element is represented with 12 bits. Six bits are used for the exponent.

What is the smallest positive normalized number that can be represented exactly? Hint: Show result as power of 2.

What is the smallest positive number that can be represented exactly? Hint: Show result as power of 2.

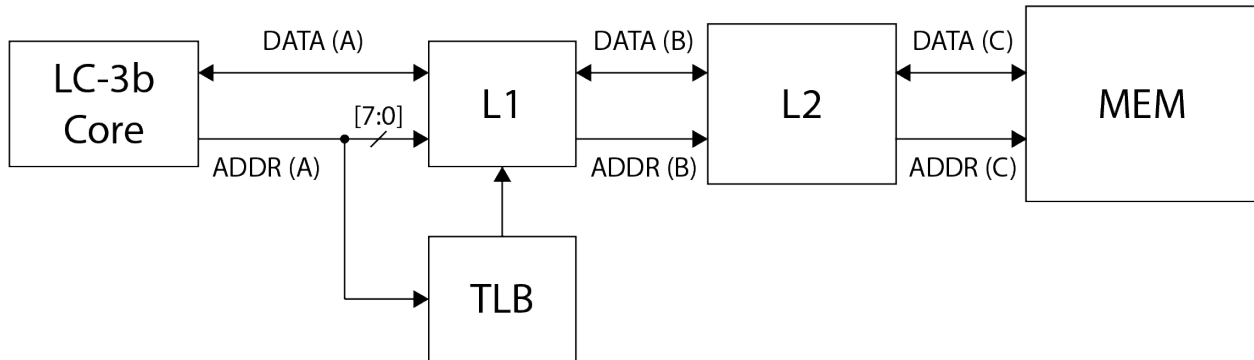
**Part c (5 points):** The microarchitecture of the VAX-11/780 has a 32-bit register containing the value in hex: 0x66666666. Could this register be of any use in performing BCD arithmetic. Yes/No. Explain.

**Part d (5 points):** Interrupts and Exceptions both interrupt the normal execution of a program, put the machine in a consistent state, and go to a service routine for handling. There are, however, many differences between interrupts and exceptions, for example when they are carried out, their priority level, the context within which they operate, etc., mostly due to the fact that interrupts are caused by events that are

while exceptions are caused by events that are

Name: \_\_\_\_\_

**Problem 2 (25 points):** We have augmented the LC-3b with the memory hierarchy shown below.



Addr(A), Addr(B), and Addr(C) represent addresses that access respectively L1, L2, and Memory. Data(B) and Data(C) each transfer a full cache line.

The table below shows a sequence of five memory accesses from the LC-3b core. If ADDR(A) misses in L1, an access is required to L2. If ADDR(B) misses in L2, an access is required to Memory. Each of the five requests from the LC-3b core must complete before the next access from the LC-3b core is initiated.

ADDR(A)	ADDR(B)	ADDR(C)	Read/Write
0x3000			Read
	0x100		Read
		0x100	Read
0x3003			Write
0x3004			Read
	0x104		Read
0x3008			Read
	0x108		Read
		0x108	Read
0x8000			Read
	0x100		Write
	0x200		Read
		0x200	Read

You may make the following assumptions:

- Virtual addresses are 16 bits and the page size is 256B
- The TLB has 2 entries and is fully associative
- All accesses to the TLB are hits
- The L1 and L2 are both physically-indexed, physically-tagged
- The L1 contains 64 sets
- If a cache line is present in the L1, it will also be present in the L2 (although the contents of L2 may not be correct)
- The caches are initially empty

Name: \_\_\_\_\_

**Part a (3 points):** How many bytes are in an L1 cache line?

 Bytes

**Part b (3 points):** How many bytes are in an L2 cache line?

 Bytes

**Part c (3 points):** Fill in the VPNs and the PFNs of the two TLB entries.

VPN	PFN

**Part d (4 points):** Is the L1 cache write through or write back? (Circle one)

**Write Through / Write Back**

Explain

**Part e (4 points):** Is the L2 cache write through or write back? (Circle one)

**Write Through / Write Back**

Explain

**Part f (4 points):** What is the associativity of the L1 cache?

 Way(s)

Explain

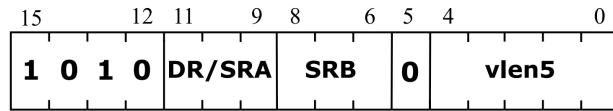
**Part g (4 points):** What is the minimum possible associativity of the L2 cache?

 Way(s)

Explain

Name: \_\_\_\_\_

**Problem 3 (25 points):** Let us use one of the unused opcodes to add an instruction DOTPRODUCT (i.e., dot product) to the LC-3b ISA. Its format will be



The DOTPRODUCT of two vectors is computed as shown below:

$$\sum_{i=0}^{n-1} A[i] \times B[i]$$

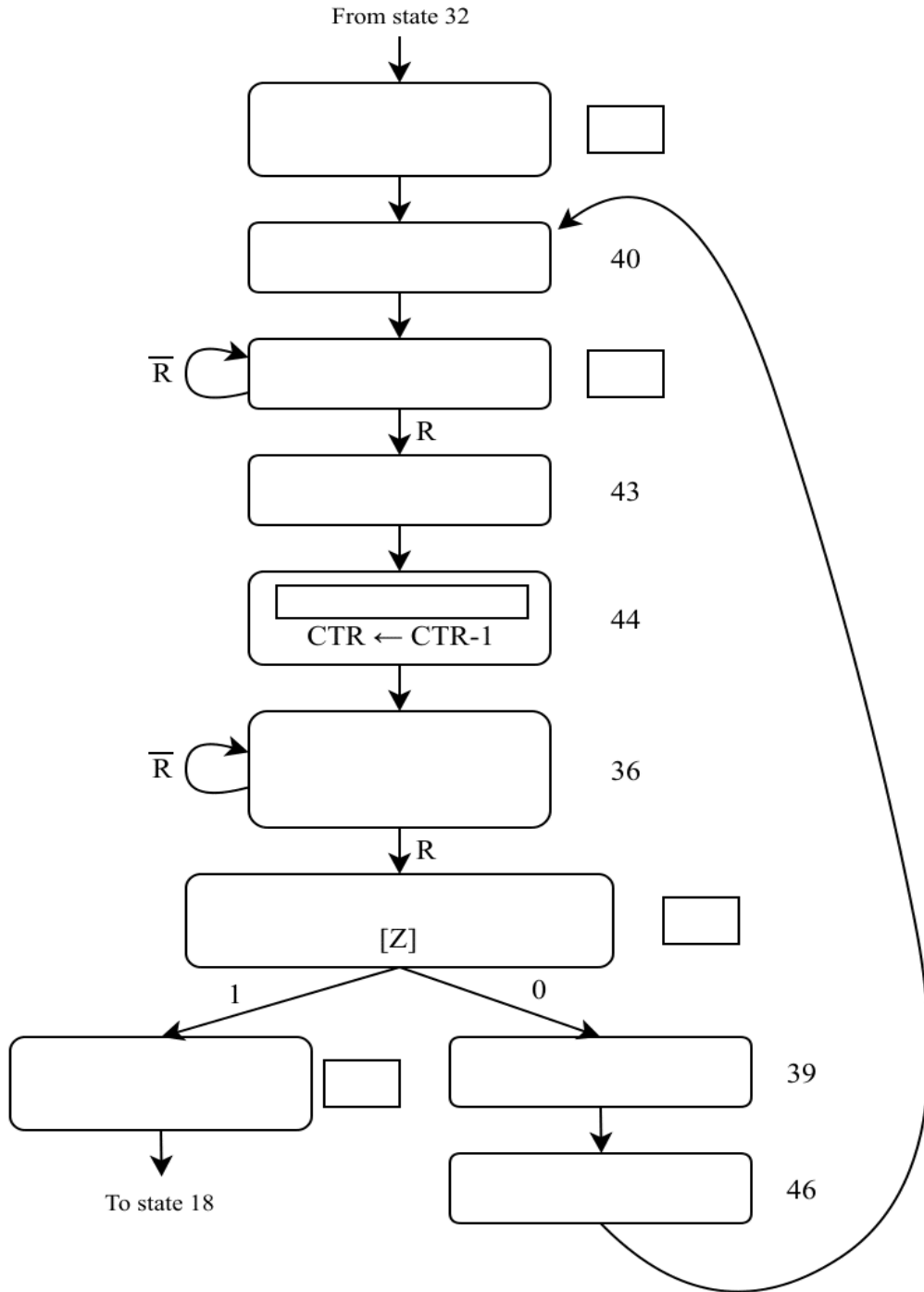
The two vectors are stored in memory. Their starting addresses are contained in SRA and SRB, and their length is specified as an immediate 5-bit value (vlen5). The instruction stores the result of the dot product in the register specified by DR. Assume vlen5 is not zero.

For this problem, you can assume no overflow will occur. Note: execution of this instruction will destroy the initial contents of SRA and SRB.

Your job: augment the LC-3b state machine, data path and microsequencer shown on the next three pages to add DOTPRODUCT to the LC-3b ISA.

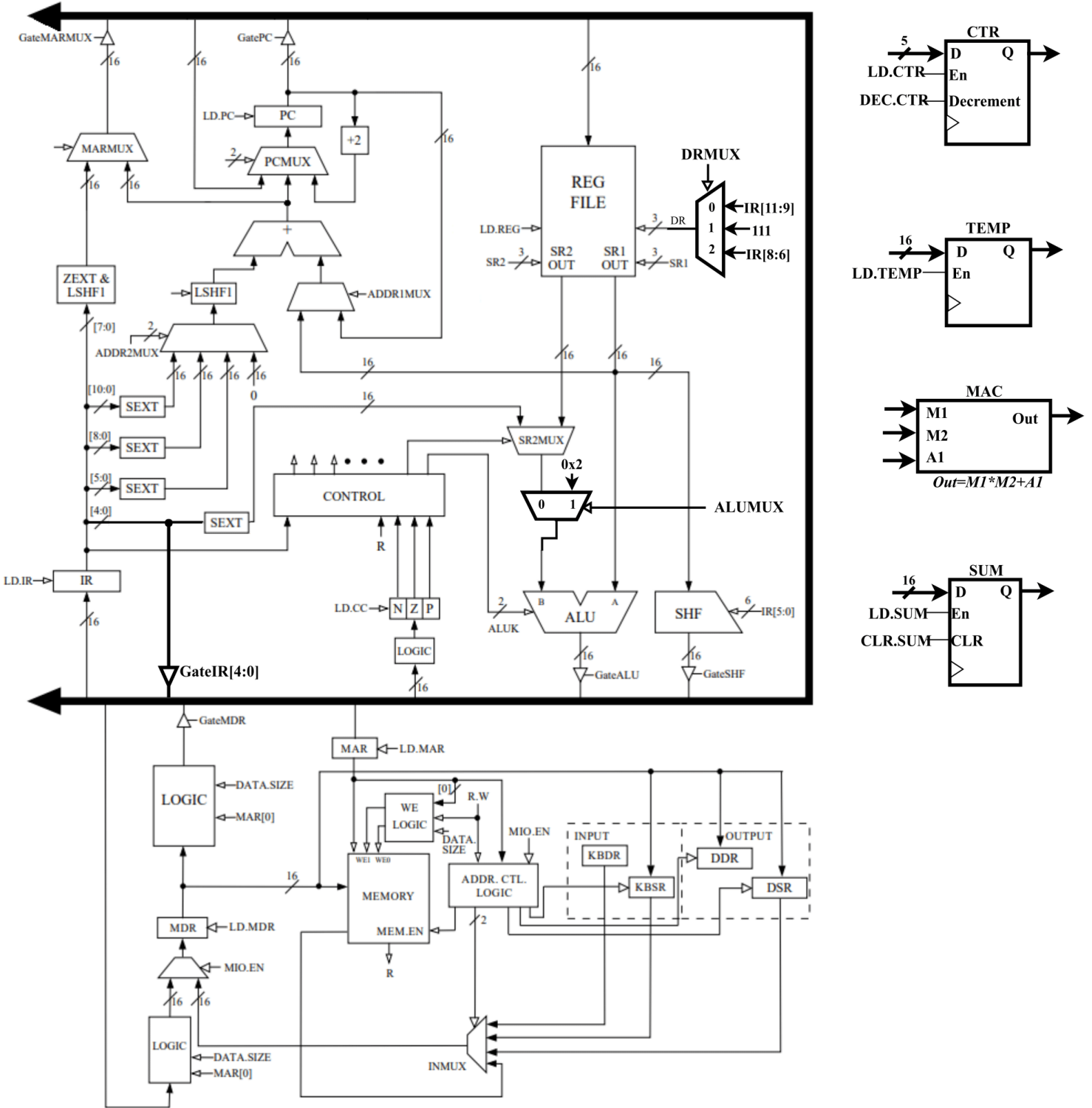
Name: \_\_\_\_\_

**Part a, The state machine (12 points):** From decode (state 32), ten states are needed to complete the execution of DOTPRODUCT. One of the states (state 44) has been partially specified. Your job is to complete the specifications of all the states and add the missing state numbers.



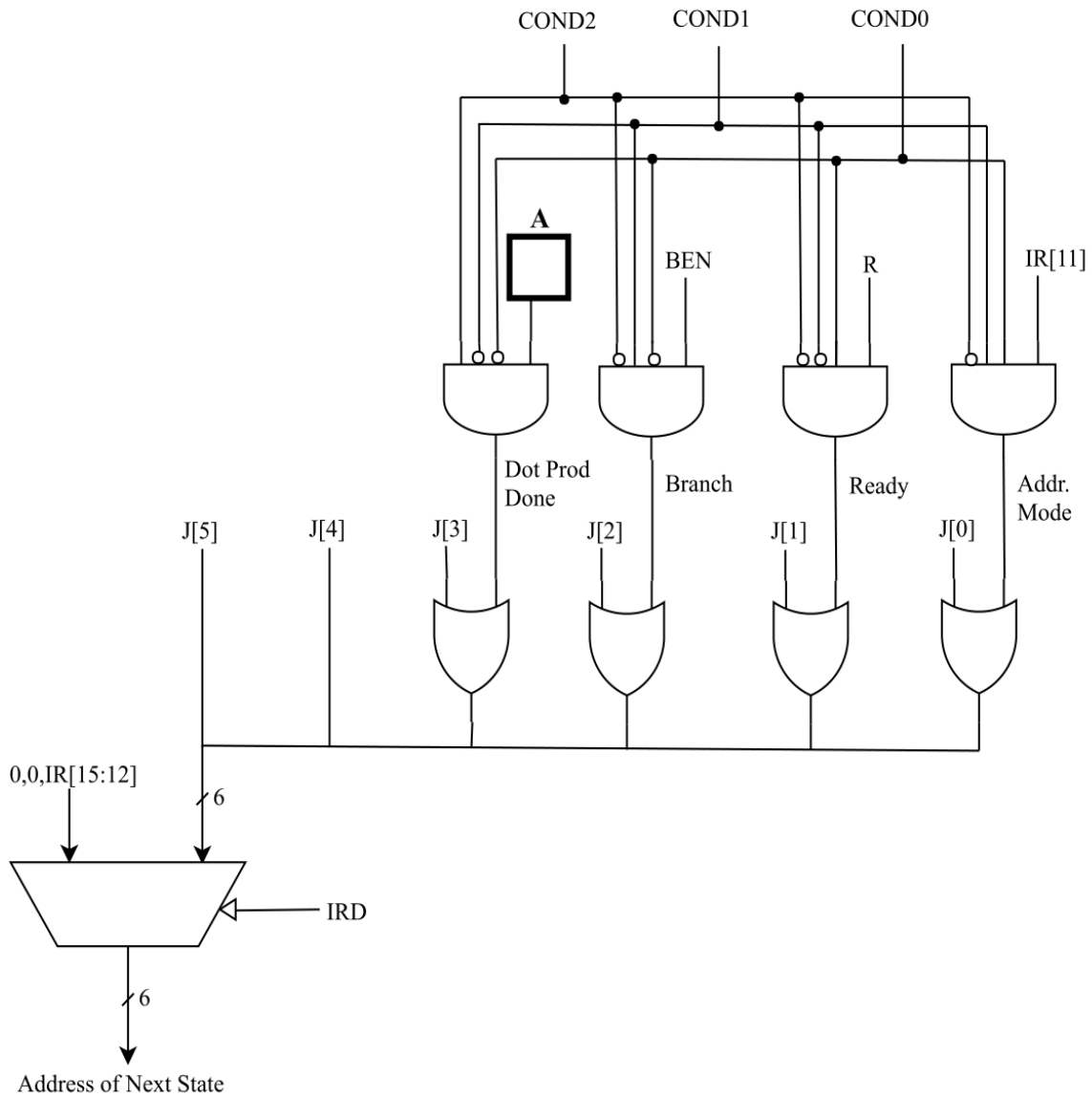
Name: \_\_\_\_\_

**Part b, The data path (10 points):** We have added GateIR[4:0], ALUMUX, made changes to DRMUX, provided registers for CTR (with built-in decrement functionality), TEMP, and SUM as well as a multiply-and-accumulate (MAC) unit. The MAC computes  $M1 \times M2 + A1$ . Your job is to implement the changes you made in Part a by connecting the necessary structures to the LC-3b datapath. You are free to add control signals and tri-state buffers as needed.



Name: \_\_\_\_\_

**Part c, The microsequencer (3 points):** To make this work, we need to add a COND2 control signal to the microsequencer. The only thing missing to complete the change to the microsequencer is the box labeled A. Your job: fill in the box labeled A.





Name: \_\_\_\_\_

**Problem 4 (30 points):** Suppose the LC-3b ISA had a 12-bit, byte-addressable, virtual address space with two levels of virtual to physical translation, similar to the VAX.

A PTE is shown below:

V	M	ACC	0...0	PFN
---	---	-----	-------	-----

It includes a Valid bit, a Modify bit, a 2 bit Access Control field, some number of unused bits (i.e., 0..0), and the PFN. The low bits of the PTE are used for the PFN.

The access control bits are defined as follows:

- 00: none
- 01: read-only
- 10: read-write
- 11: —

The virtual address space is divided evenly into two regions. The first half is user space, the second half is system space.

The user space page table starts at the beginning of a page. The system space page table starts at the beginning of a frame. We require 1/4 of physical memory to store the entire system page table.

A user program fetches and executes one LC-3b instruction, resulting in six accesses to physical memory, as shown by the following table:

VA	PA	Data
—	x00EA	x9...4
x08B0		x9...2
x0100		
—	x00EE	x9...5
x08F6		x9...1
x0572		x10A0

Note: Since the size of the PTE has not been given, entire PTEs are not shown in the above table.

**Part a (17 points):** Fill in the entries for the following:

	Physical Address Space	<input type="text"/>
PTE Size	<input type="text"/>	Page Size <input type="text"/>
SBR	<input type="text"/>	UBR <input type="text"/>

Name: \_\_\_\_\_

**Part b (3 points):** After the instruction is executed, the register file is as shown:

Register	Value
R0	x0550
R1	x0590
R2	x00A0
R3	x0200
R4	xFFA0
R5	x000C
R6	x0010
R7	x0100

What LC-3b instruction was executed?

**Part c (10 points):** Complete the entries in the memory access table shown on the previous page.