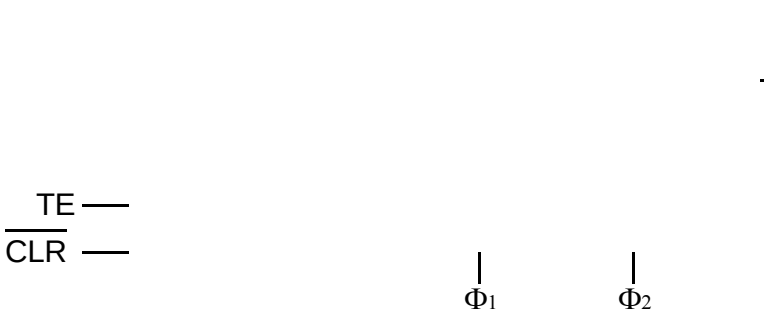
Instructions: This is a closed book, closed note exam. Calculators are not permitted. If you have a question, raise your hand and I will come to you. Please work the exam in pencil and do not separate the pages of the exam. For maximum credit, show your work.
Good Luck!

Your Name (***please print***) _____

1	2	3	4	total
24	18	30	28	100

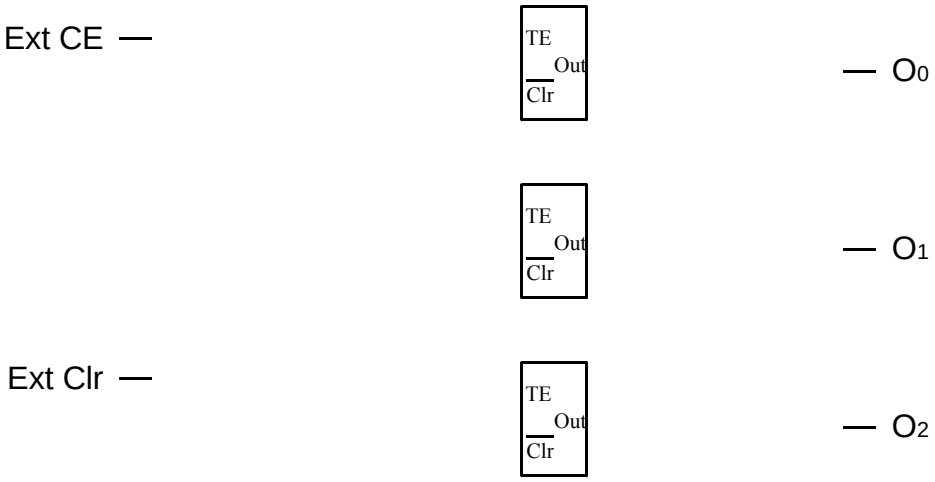
Problem 1 (2 parts, 24 points) Counters

Part A (12 points) Design a toggle cell using transparent latches and basic gates. Use an icon for the latch. Your toggle cell should have an active high toggle enable input **TE**, and an active low clear input **$\overline{\text{CLR}}$**, clock inputs Φ_1 and Φ_2, and an output **Out**. The **$\overline{\text{CLR}}$** signal has precedence over **TE**. Label all signals. Also complete the behavior table for the toggle cell.



TE	$\overline{\text{CLR}}$	CLK	Out
0	0	↑↓	
1	0	↑↓	
0	1	↑↓	
1	1	↑↓	

Part B (12 points) Now combine these toggle cells to build a **divide by six** counter. Your counter should have an external clear, external count enable, and three count outputs O_2, O_1, O_0. Use any basic gates (AND, OR, NAND, NOR, & NOT) you require. Assume clock inputs to the toggle cells are already connected. *Your design should support multi-digit systems*.



Problem 2 (2 parts, 18 points) Datapath Elements

Part A (9 points) Consider the following input and output values for a shift operation. Determine the shift *type* and *amount* required to achieve the listed transformation. I/Os are in hexadecimal.

Input Value	Output Value	Shift Type	Shift Amount (signed decimal value)
87654321	43210000		
87654321	76543218		
87654321	00008765		

Part B (9 points) Consider the following input and output values for a logical operation. Determine the *logical function* and *function code* (in hexadecimal) required for the operation.

X Input	Y Input	Output	Logical Function	Function Code
87654321	0000FFFF	00004321		
87654321	0000FFFF	8765FFFF		
87654321	0000FFFF	8765BCDE		

Problem 3 (3 parts, 30 points) Memory Systems

Part A (10 points) Consider a DRAM chip organized as **512 million addresses** of **eight bit words**. Assume both the DRAM cell and the DRAM chip is square. The column number and offset concatenate to form the memory address. Using the organization approach discussed in class, answer the following questions about the chip. **Express all answers in decimal**.

number of columns _____

column decoder required (*n* to *m*) _____

type of mux required (*n* to *m*) _____

number of muxes required _____

number of address lines in column number _____

number of address lines in column offset _____

Part B (10 points) Consider a **4 Gbyte** memory system with **256 million addresses** of **16 byte words** using DRAM chips organized as **64 million** addresses by **16 bit words.**

word address lines for memory system _____

chips needed in one bank _____

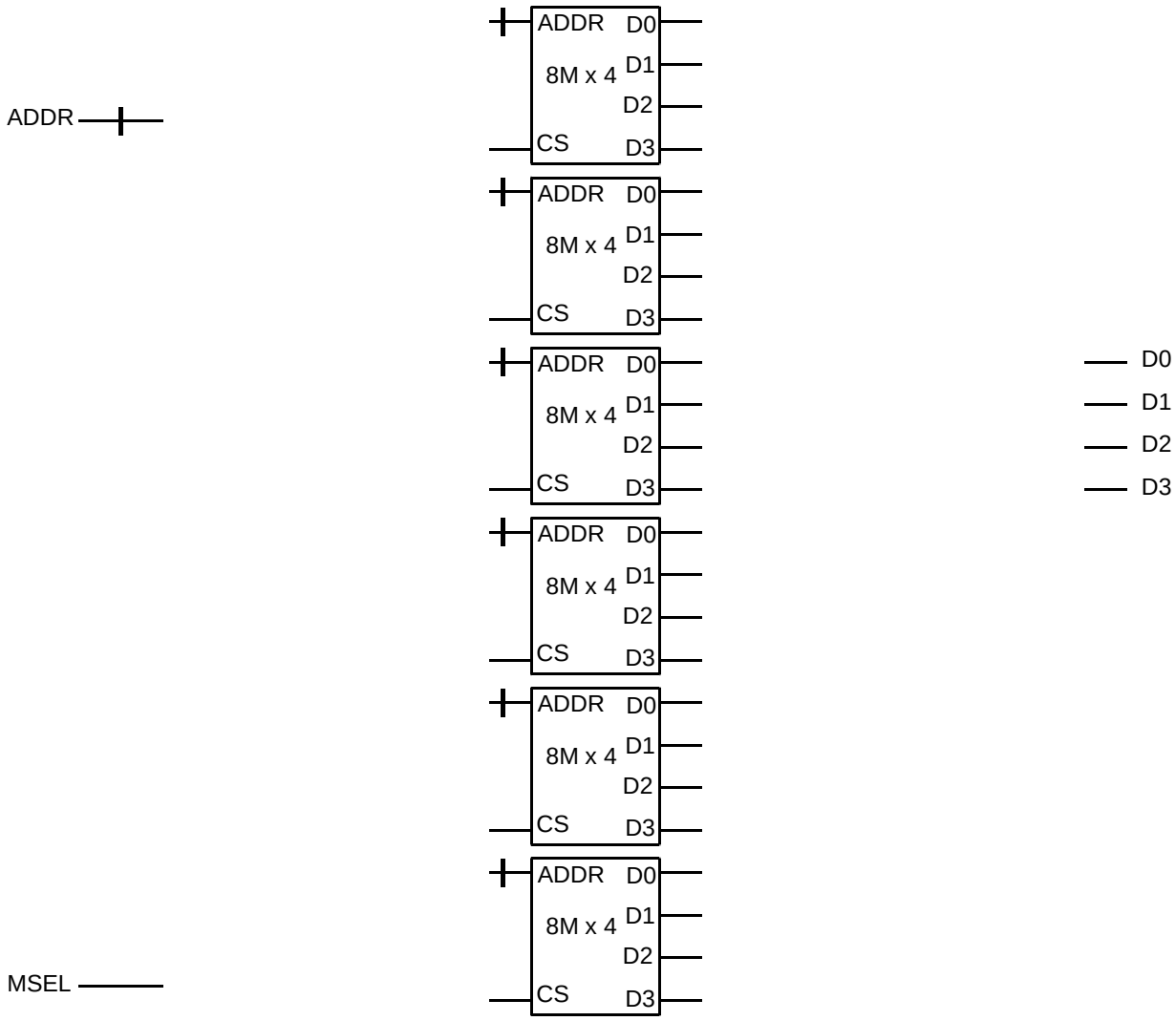
banks for memory system _____

memory decoder required (*n* to *m*) _____

DRAM chips required _____

Part C (10 points) Design a 48M address x 4 bit memory system with six 8M address x 4 bit memory chips. **Label all busses and indicate bit width**. Assume R/W is connected and not

shown here. Use a decoder if necessary. Place a star on the chip(s) that contain address 26,000,000.



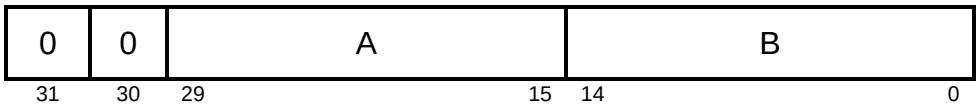
Problem 4 (2 parts, 28 points) Microcode

Using the supplied datapath, write microcode fragments to accomplish the following procedures. Express all values in hexadecimal notation. Use 'X' when a value is don't cared. For maximum credit, complete the description field.

Part A (14 points) $R_7 = \frac{3 \times R_5}{16} - 256 \times R_6$ **Modify only R_5, R_6 and R_7.**

#	X	Y	Z	rwe	im en	im va	au en	-a /s	lu en	lf	su en	st	ld en	st en	r/ -w	msel	description
1																	
2																	
3																	
4																	
5																	
6																	
7																	

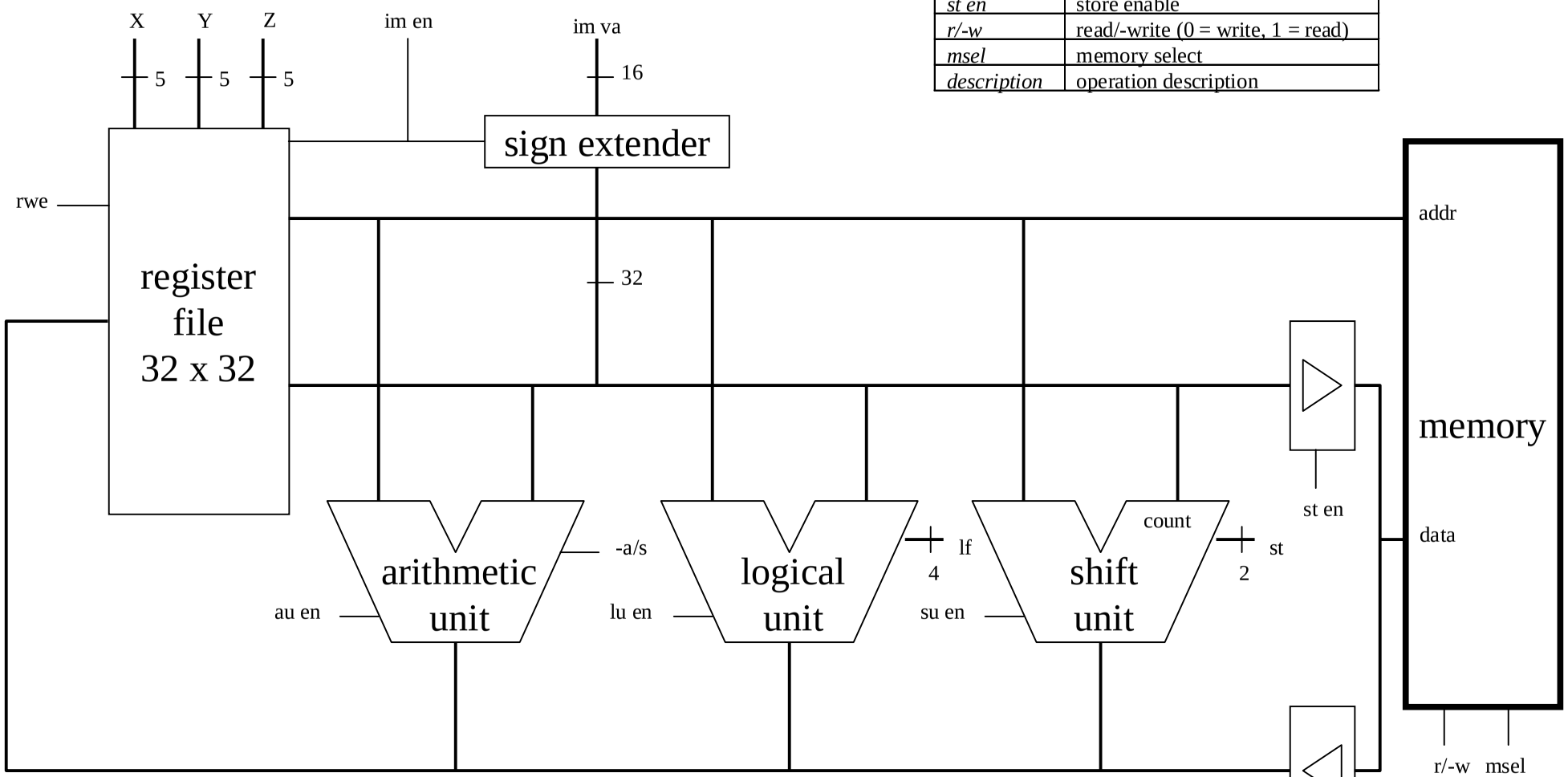
Part B (14 points) Write a microcode sequence that loads a 32 bit word from memory location 0x4000, unpacks and averages two 15 bit unsigned values (A and B), and then stores the result back to memory location 0x4000. Assume the most significant two bits of the register are zero. **Modify only R_1, R_2, and R_3.**



#	X	Y	Z	rwe	im en	im va	au en	-a /s	lu en	lf	su en	st	ld en	st en	r/ -w	msel	description
1																	
2																	
3																	
4																	
5																	
6																	
7																	
8																	

cycle	cycle number
X	register driven onto X bus
Y	register driven onto Y bus
Z	register written from Z bus
rwe	register write enable
im en	immediate enable on Y bus
im va	immediate value

au en	arithmetic unit enable
-a/s	-add / sub (0 = add, 1 = subtract)
lu en	logical unit enable
lf	logical function
su en	shift unit enable
st	shift type
ld en	load enable
st en	store enable
r/-w	read/-write (0 = write, 1 = read)
msel	memory select
description	operation description



logical functions

X	Y	out
0	0	lf_0
1	0	lf_1
0	1	lf_2
1	1	lf_3

shift types
0 = logical
1 = arithmetic
2 = rotate
+ count shifts right
- count shifts left