1. (11 pts) **We're going to play Final (exam) Jeopardy!** Associate the following answers with the appropriate question. (You are given the "answers": Pick the "question" that goes best with each "answer".) The first two have been done for you.

"Answers:"

1 **(u)** Anywhere but here.

2. A Flip Flop
3. A technique in which the occurance of one event on a bus follows and depends only on the occurance of a previous event.
4. When an I/O device is allowed to write directly into memory.
5. To maximize the efficient use of an expensive system.
6. A structure that holds recent mappings of virtual to physical addresses.
7. An unscheduled subroutine call.
8. What happens when a page is referenced but not in memory.
9. A binary digit appended to a group of binary digits to make the sum of all the digits either always odd or always even.
10. To make Memory appear to cost as much as the cheapest element and perform as well as the fastest element.
11. A structure that holds the contents of recently referenced memory locations.
12. Technique used in CD-ROM Drives.

"Questions:"

a) What is synchronous timing?
b) What is asynchronous timing?
c) What is an Address Translation Lookaside Buffer?
d) What is a circuit that exhibits purely sequential behavior?
e) What is the Data Bus?
f) What is a Parity error?
g) What is a Register?
h) What is a Cache?
i) What is DMA?
j) What is an interrupt?
k) What is the goal of the Memory Heirarchy?
l) What is a Page?
m) What is a Floating Point number?
n) What is a Page Fault?
o) What is a Parity bit?
p) What is a Sign bit?
q) What is polling?
r) What is Constant Linear Velocity?
s) What is the goal of multiprogramming?
t) What is this question?
u) Where would I rather be right now than where I am?

3. (6) Caches can be either Virtually Addressed or Physically Addressed. Explain the difference, and give an advantage and a disadvantage for each.

4. (5) What does a Virtual Memory system do?

5. (5) Explain the principle of Locality and why it is important.

6. (4) What is the worst case path of a circuit?

7. (6) Given a logical 19-bit address and a 64K-byte physical memory for a byte-addressable machine,

   How big is the physical address space?

   How big is the virtual address space?

   Assuming 4K-byte pages, how many page frames are there? How many pages?

   Assuming 512-byte pages, how many page frames are there? How many pages?

8. (3 pts) For the following 8-bit patterns, add a 9th bit to create:

   Even Parity: 00001011_

   Odd Parity: 11101101_

   What is the Hamming distance between these two codewords (including the parity bit)?

9. (12 pts) Here is a 12-bit Error Correction code format (same one used in class):

$$d_8\ d_7\ d_6\ d_5\ C_4\ d_4\ d_3\ d_2\ C_3\ d_1\ C_2\ C_1$$

   a. Given the *data* bit pattern

   0 1 0 1 0 0 0 1

   in a machine using the above ECC code, what bit pattern gets sent to memory? (No credit will be given without work being shown.)

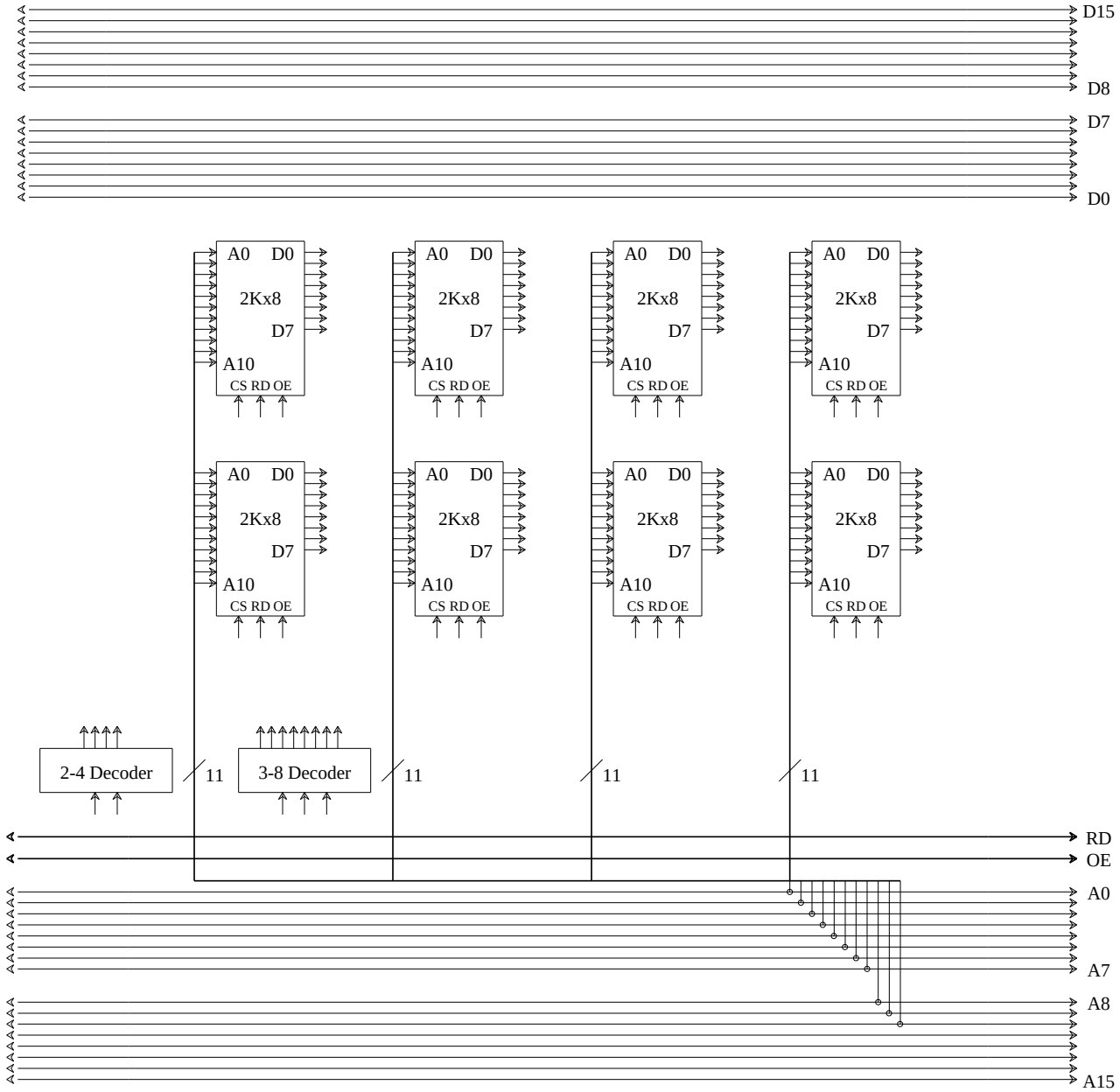
   b. In this same machine, the following bit pattern is retrieved from memory:

   1 0 1 1 0 1 0 1 0 0 1 0

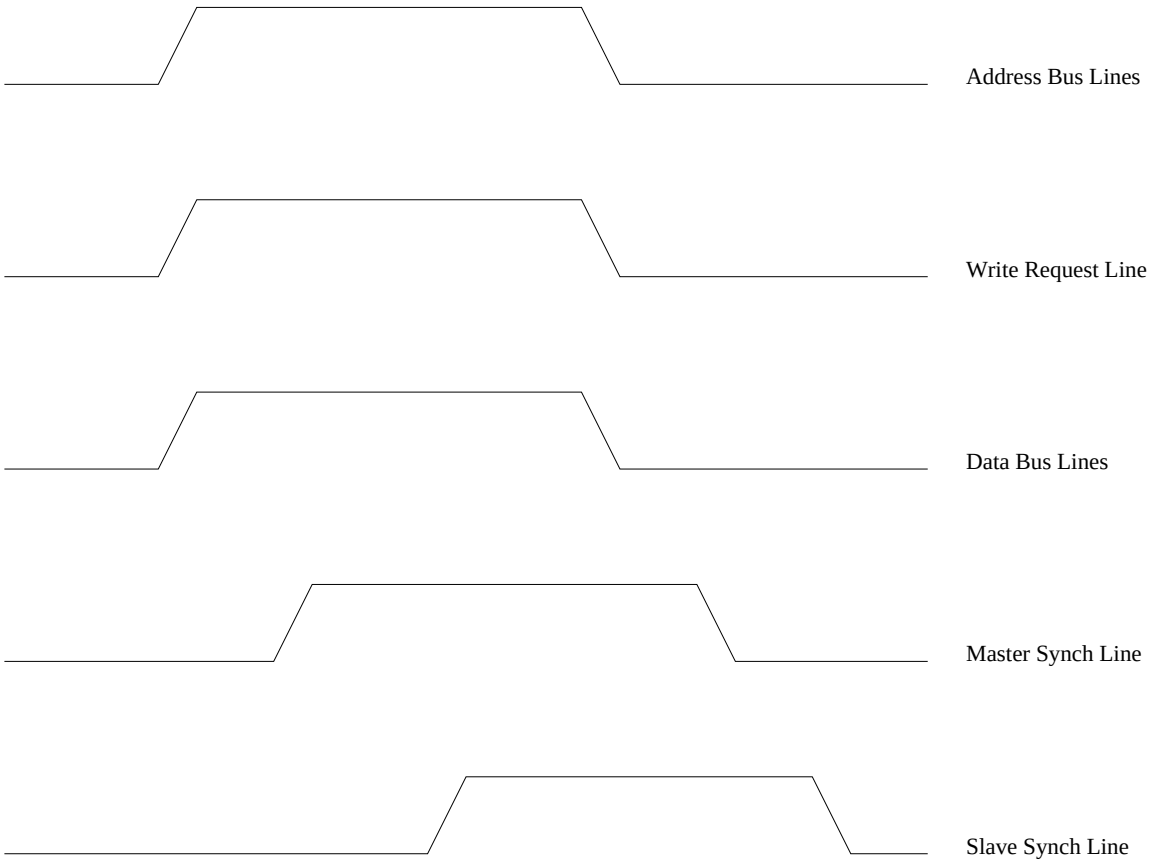
   Assuming the above Error Correction code format, identify and correct any errors that may have occurred during transmission or storage. (No credit will be given without work being shown.)

10. (7 pts) Add the connections to the following diagram necessary to create a 4Kx16 memory. Not all of the hardware shown is required to perform this task.

CS - Chip Select
OE - Output Enable
RD - Read (Read/Write, technically)



11. (5 pts) Add arrows to the following diagram to indicate which signal transitions **cause** other transitions to occur.



Now explain in words (briefly) what the arrows you have added are doing. (What is the sequence of events in words?)

12. (20) The Megatron Corporation has decided to go small. They are planning on rolling out the "Megamax Weenie", a byte-addressable computer with a 32-bit word size. In this machine the bus to main memory is 16 bits wide, and servicing a cache miss takes 20 clock cycles (in addition to the time necessary to do a cache lookup). In order to improve performance, they are considering adding a 64-byte Direct-Mapped cache with a line size of 1 word and an access time of 1 cycle. Given the following address reference sequence (in Hex):

0x12,0x14,0x16,0xD1,0x13

a) Write down how you are partitioning each address (which bits are the Tag, offset, etc.)

b) In the table below, fill in the proposed Cache's Tag values after each memory reference has been processed. If it is a hit, mark the entry number to indicate this, and if it is a miss enter what the new tag should be. (X indicates the entry is invalid).

Tag Array	Contents of Tag Array after processing address (Time ->)					
Entry Number	Initial Contents	0x12 (00010010)	0x14 (00010100)	0x16 (00010110)	0xD1 (11010001)	0x13 (00010011)
0	X					
1	X					
2	X					
3	X					
4	X					
5	X					
6	X					
7	X					
8	X					
9	X					
A	X					
B	X					
C	X					
D	X					
E	X					
F	X					

What is the Average Memory access time for this sequence of references?

Now fill in the contents of the Data array after processing the given address references. Write down only the ones that change.

Data Array	Data Array Contents after processing address
Entry Number	0x14
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
A	
B	
C	
D	
E	
F	

Memory Contents at Hex Address **XY**																
Most Significant Digit (**X**)	Least Significant Digit (**Y**)															
	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	23	20	6d	61	74	74	27	73	20	67	76	69	65	77	20	73
1	68	65	6c	6c	20	73	63	72	69	70	74	0a	73	65	74	20
2	67	66	69	6c	65	20	3d	20	24	31	0a	09	65	63	68	6f
3	20	2d	6e	20	22	67	67	72	61	70	68	20	24	67	66	69
4	6c	65	2e	2e	2e	22	20	0a	09	65	63	68	6f	20	22	2e
5	73	70	20	32	22	20	3e	21	20	2f	74	6d	70	2f	74	65
6	6d	70	24	24	2e	6e	72	0a	09	65	63	68	6f	20	22	2e
7	70	6f	20	2b	30	2e	35	69	22	20	3e	3e	20	2f	74	6d
8	23	7b	92	08	22	41	85	32	69	73	11	35	97	54	31	48
9	88	73	48	72	98	21	42	85	62	65	90	84	31	56	55	83
a	43	64	84	36	59	3c	8a	95	3b	8f	0e	41	7a	40	2b	3c
b	4c	d4	c7	82	a0	38	f9	c6	29	a3	d0	9c	7d	41	2b	75
c	54	69	9c	3b	b0	2a	d9	3e	45	72	6e	f0	f9	3f	a0	0a
d	60	89	43	d8	c0	e7	49	76	59	21	2c	c8	a8	f2	87	43
e	76	8f	2e	a9	ff	38	ae	65	dd	cf	21	84	ce	e4	34	51
f	8a	65	30	2f	c9	3a	58	72	3e	a0	4f	38	96	47	21	80

13. (20) Megatron decided to experiment with using a 48-byte 3-way Set Associative Cache (instead of the Direct-mapped Cache). Remember, the Megamax Weenie is a byte-addressable machine with a 32-bit word size, a line size of 1 words, a 16-bit bus to main memory, and a Main Memory access time of 20 cycles (in addition to the time necessary to to a cache lookup). The Cache access time is still 1 cycle. Given the same address reference sequence (in Hex) as before:

0x12,0x14,0x16,0xD1,0x13

a) Write down how you are partitioning each address (which bits are the Tag, offset, etc.)

b) In the table below, fill in the proposed Cache's Tag values after each memory reference has been processed. If it is a hit, put an "H" in the tag field, and if it is a miss write down what the new tag should be. Use an LRU replacement scheme, and after each address is processed be sure to indicate the age of the references. (MRU = Most Recently Used, LRU = Least Recently Used).

Tag Array				Contents of Tag Array after processing address (Time ->)									
Set #	Entry #	Initial contents		0x12 (00010010)		0x14 (00010100)		0x16 (00010110)		0xD1 (11010001)		0x13 (00010011)	
		Age	Tag	Age	Tag	Age	Tag	Age	Tag	Age	Tag	Age	Tag
0	0		1101										
	1	LRU	0111										
	2	MRU	0011										
1	0	LRU	0010										
	1		1001										
	2	MRU	0011										
2	0		1100										
	1	LRU	0111										
	2	MRU	0011										
3	0	LRU	0010										
	1		1001										
	2	MRU	0011										

What is the Average Memory access time for this sequence of references?

14. (11 pts) The following tables contain some of the information about a segmented, paged virtual memory system and certain select memory locations. Total physical memory size is 8K bytes, and the page size is 2048 bytes. All numbers in this table are in Hex unless otherwise noted.

Segment Table		
Entry Number	Presence Bit	Page Table
0	1	5
1	0	0
2	1	1
3	1	7
4	1	2
5	1	2
6	1	2
7	1	2

Memory	
Address	Contents
0x00A4	0x76
0x01A4	0x73
0x02A4	0x32
0x03A4	0x46
0x14A4	0x30
0x2AA4	0x29
0x05A4	0xa9
0x09A4	0x74
0x1BA4	0x05
0x3CA4	0x23
0x0DA4	0xE3
0x17A4	0xAE
0x2EA4	0x92

Page Table 0			
Entry Number	Present? (1=Yes)	Disk Addr	Frame Number
0	1	1234123	0x4
1	0	0893748	0x7
2	1	2489567	0x1
3	1	9623873	0x5

Page Table 2			
Entry Number	Present? (1=Yes)	Disk Addr	Frame Number
0	1	1234123	0x1
1	0	0893748	0x3
2	1	2489567	0x5
3	1	9623873	0x7

Page Table 5			
Entry Number	Present? (1=Yes)	Disk Addr	Frame Number
0	1	1234123	0x2
1	0	0893748	0x3
2	0	2489567	0x4
3	1	9623873	0x4

Page Table 7			
Entry Number	Present? (1=Yes)	Disk Addr	Frame Number
0	1	1234123	0x5
1	0	0893748	0x6
2	1	2489567	0x1
3	1	9623873	0x2

For each of the following convert the virtual address into a physical address (if possible) and write down the value of the memory location corresponding to the address. If it is not possible to do so, explain why.

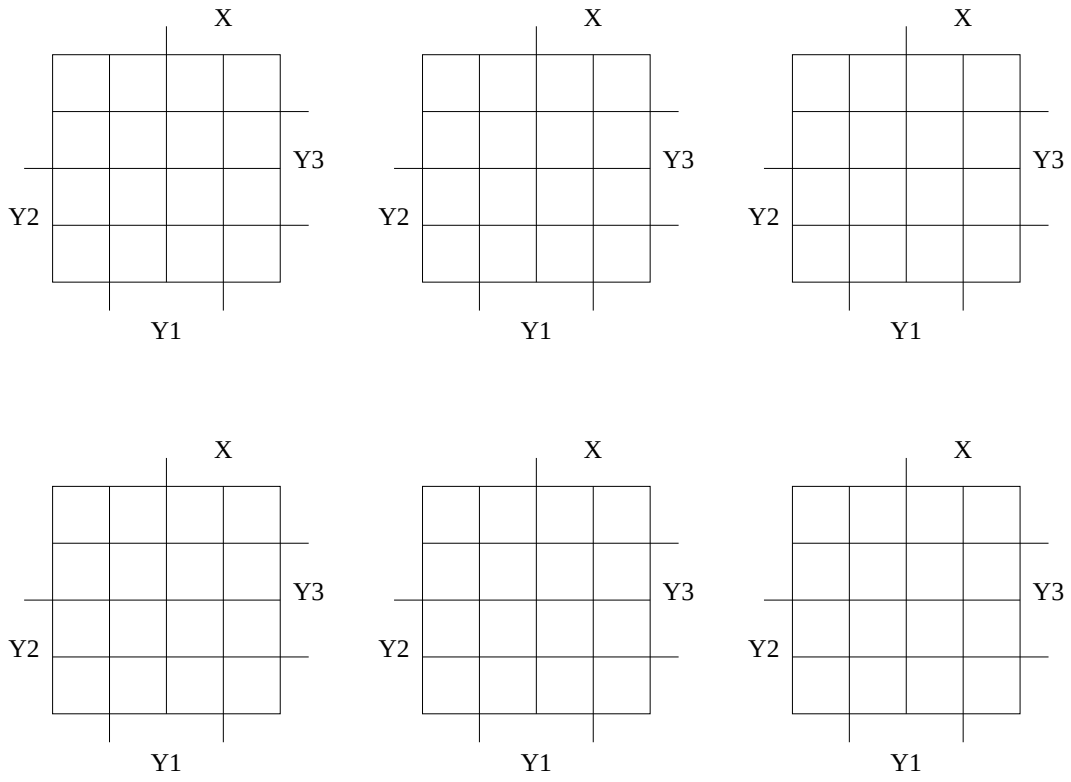
0x96A4 (**1 0 0 1 0 1 1 0 1 0 1 0 0 1 0 0** in binary).

0x0EA4 (**0 0 0 0 1 1 1 0 1 0 1 0 0 1 0 0** in binary).

0x28A4 (**0 0 1 0 1 0 0 0 1 0 1 0 0 1 0 0** in binary).

15. (20) Given the following table, draw the Karnaugh maps for Y1', Y2', and Y3' and Z in terms of X, Y1, Y2 and Y3, and then write **minimum** boolean equations for each.

Present State (Y1 Y2 Y3)	Next State X=0 (Y1' Y2' Y3')	Next State X=1 (Y1' Y2' Y3')	Output X=0	Output X=1
000	101	111	0	0
001	101	111	0	1
010	101	101	0	1
011	010	010	1	1
100	100	100	0	0
101	000	010	0	1
111	010	010	1	1



16. (15 pts) Given the following Karnaugh maps, implement the sequential machine using an SR FF for Y1, a JK FF for Y2, and a T FF for Y3. You do not need to draw the gates, but you do need to write down the **minimized** input equations for each of the inputs of each of the Flip Flops in the circuit.

(Note the change in the order of Y1' and Y2'!!!)

Y2'

			X	X	
	1			1	
		1	1	d	Y3
Y2	1	d		1	Y3
Y2		1	1		
		Y1	Y1		

Y1'

			X	X	
	d				
	1	d	1	1	Y3
Y2			d		Y3
Y2	1	1	1	1	
		Y1	Y1		

Y3'

			X	X	
	1			1	
		1	1		Y3
Y2		1	1		Y3
Y2	1			1	
		Y1	Y1		

			X	X	
					Y3
Y2					Y3
Y2					
		Y1	Y1		

			X	X	
					Y3
Y2					Y3
Y2					
		Y1	Y1		

			X	X	
					Y3
Y2					Y3
Y2					
		Y1	Y1		

			X	X	
					Y3
Y2					Y3
Y2					
		Y1	Y1		

			X	X	
					Y3
Y2					Y3
Y2					
		Y1	Y1		

			X	X	
					Y3
Y2					Y3
Y2					
		Y1	Y1		