

# ECE 2020

## Digital Systems Design

### Quiz IV

November 29, 2012

This quiz is closed book and closed note and no calculators.

There are four questions. Do read them over before you start to work. If you need to make any assumptions, state them.

The meaning of each question should be clear, but if something does not make any sense to you, please ask for clarification. If you run out of room, please continue on the back of the previous page.

**Good Luck!**

Name (Please print) Solution

This quiz will be conducted according to the Georgia Tech Honor Code. I pledge to neither give nor receive unauthorized assistance on this exam and to abide by all provisions of the Honor Code.

Signed \_\_\_\_\_

Question	Score	Max
1		20
2		30
3		30
4		20
<b>Total</b>		<b>100</b>

1. (20) Logic and barrel shifter units

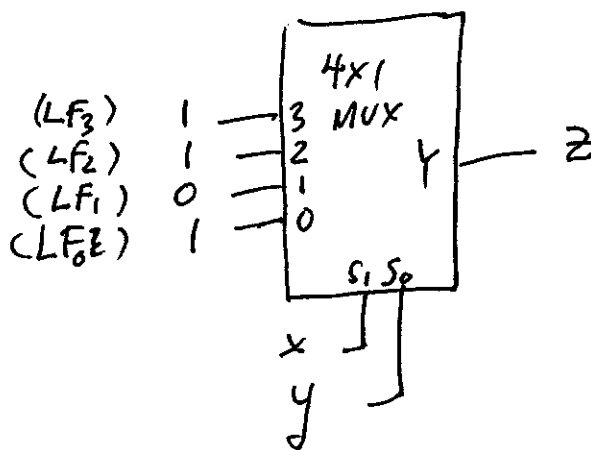
(a) (5) What is the result of an arithmetic right shift by 10 bits of (the half word) 0xA752.

$A752 = 1010\ 0111\ 0101\ 0010$   
 arith shift right  $\rightarrow 1111\ 1111\ 1110\ 1001$   
 $= \text{F F E 9}$   
0x FFE9

(b) (5) What is the result of a rotate left by 8 bits of (the half word) 0x3D7F.

3D7F  
 $\swarrow \searrow$   
 7F3D  $\leftarrow$   
0x7F3D

(c) (10) Show the design for one bit of a logic unit which can compute any function of two boolean variables (as designed in class). You may build the unit out of any multiplexor, decoder, or encoder module, or any gates you need. Show how to configure your design to compute the function  $z = x + \bar{y}$ . You must show what signals or values are connected to which pin of your design. You must label each pin inside the module.



x	y	$z = x + \bar{y}$
0	0	1
0	1	0
1	0	1
1	1	1

2. (30) Memory

(a) (5) Consider a 4Kx4 memory chip. Suppose there are 128 bits per row? What is the specification of the column mux?

32 nibbles / row : column mux:  $\boxed{4 - 32 \times 1}$

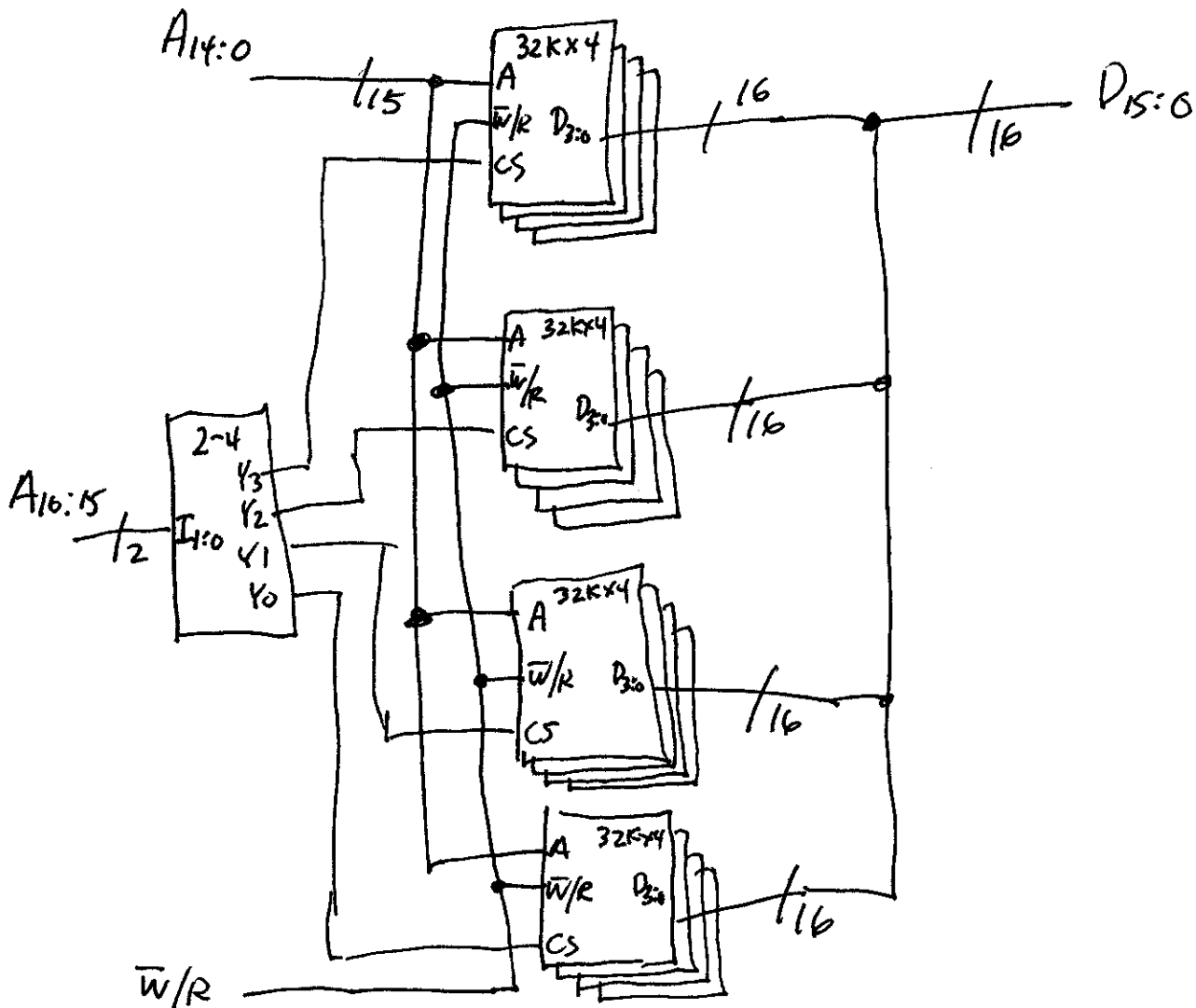
(b) (5) For the chip in (a), what is the specification of the row decoder?

$4K \times 4 = \frac{2^{14} \text{ bits/chip}}{128 / \text{row}} = 2^7 \text{ rows/chip}$   
 row decoder:  $\boxed{7 - 128}$

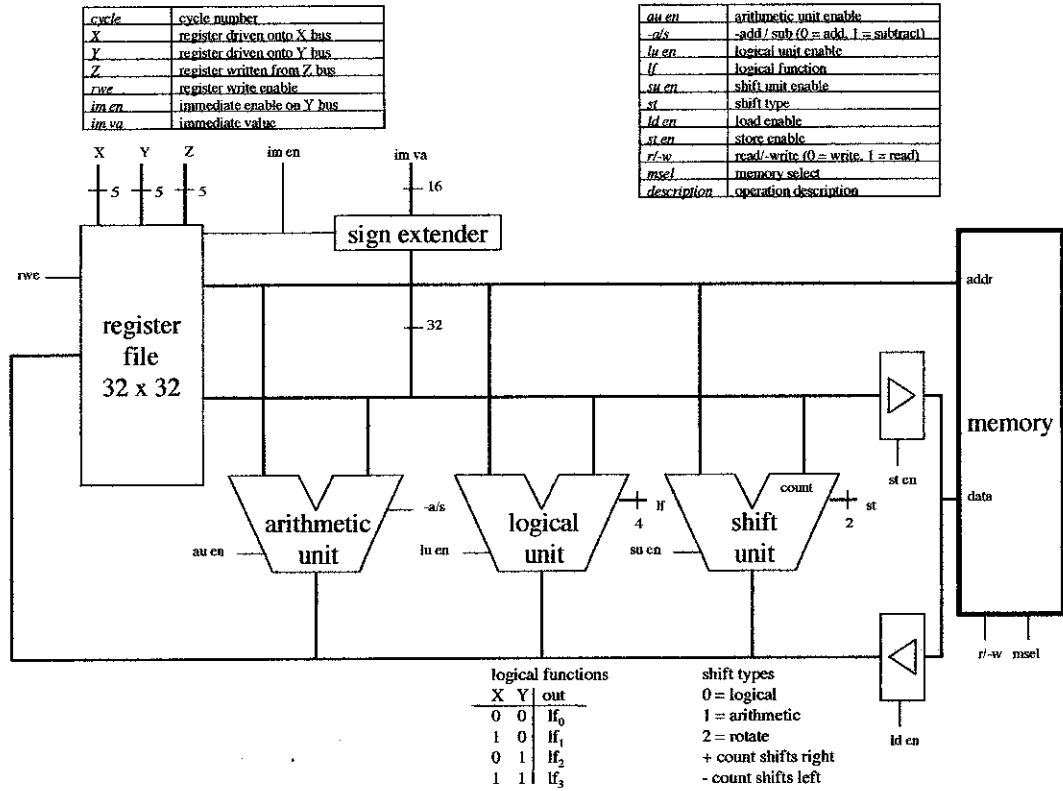
(c) (5) Which is more dense (bits/area), SRAM or DRAM? Why?

DRAM. DRAM has 1T/bit vs 6T/bit for SRAM.

(d) (15) Design a 128Kx16 memory out of 32Kx4 chips. You must identify all address, data, and control bits/busses and their connections for full credit.



### 3. (30) ISA/Datapath



Xra	Yra	Zwa	rwe	im en	im va	au en	a/s	lu en	LF	su en	ST	st en	ld en	r/~w	m sel
0B	0D	07	1	0	XXXX	1	0	0	XXXX	0	XX	0	0	X	0
05	07	02	1	0	XXXX	0	X	1	1000	0	XX	0	0	X	0
08	02	XX	0	0	XXXX	0	X	0	XXXX	0	XX	1	0	0	1
0C	XX	0B	1	1	0002	0	X	0	XXXX	1	01	0	0	X	0

a) (20) Write the microcode for the following sequence of RTL statements in the table above. Register addresses and immediate values should be in hex. All other fields must in binary and be the correct number of bits. Assume that register i (Ri) is initialized to i. Assume that all arithmetic operands are signed (twos complement) numbers.

$R7 = R11 + R13$        $R7 \leftarrow 23$   
 $R2 = R5 \text{ AND } R7$        $R2 \leftarrow 0$   
 $\text{MEM}[R8] = R2$        $\text{MEM}[8] \leftarrow 0$   
 $R11 = R12/4$  (division by 4)       $R11 \leftarrow 3$

- b) (5) What is the value of the data that is loaded into Mem[8]? 0
- c) (5) What is the final value in R11? 3

4. (20) Instruction Set Architecture

a) Consider the MIPS assembly language statement: `addi $12, $4, -7`

If the opcode for `addi` is `010110`, what is the hexadecimal encoding of this instruction? Show the encoding of each field and the overall hex encoding.

010 | 110 | 0100 | 0100      -7 → FFF9

0x5984FFF9

b) Consider the MIPS assembly language statement: `XOR $16, $24, $8`

If the opcode for `XOR` is `100100`, what is the hexadecimal encoding of this instruction? Show the encoding of each field and the overall hex encoding.

100 | 100 | 1000 | 0100 | 000000 ...

0x92184000

c) In a couple of sentences, why is `R0 = 0` in MIPS (why did the architects decide to do this)?

Since there is almost always a "0" register needed, they provided one, and since it's hardwired, the programmer does not need to initialize it.

d) What is the rationale for immediate type instructions.

Immediate operands are <sup>constants</sup> encoded within immediate inst. Therefore, there is no need to allocate a register to hold that constant, which would require at least one additional operation, and might also force a needed value out of the register file taking even more instructions.