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

EE 306, Fall 2021
Yale Patt, Instructor
TAs: Sabee Grewal, Ali Fakhrzadehgan, Ying-Wei Wu, Michael Chen, Jason Math, Adeel Rehman
Exam 1, October13, 2021

Name: Student

Problem 1 (25 points): 25

Problem 2 (15 points): 15

Problem 3 (15 points): 15

Problem 4 (20 points): 20

Problem 5 (25 points): 25

Total (100 points): 100

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.

I will not cheat on this exam.

____
Signature

GOOD LUCK!

Name:____

Problem 1. (25 points):

Part a. (5 points): Which of the 15 opcodes in the LC-3 set condition codes?

AND, ADD, NOT, LD, LDR, LDI

Part b. (5 points): In class we learned that a bit is a number with one of two possible values: 0 or 1. Similarly, a *trit* is a number with one of three possible values: 0, 1, or 2. Given n trits, how many unique items can we represent?

3^n

Part c. (5 points): 42 is a base-5 number. We want you to convert it to binary, decimal, and hexadecimal. Fill the boxes below.

binary: 10110 decimal: 22 hexadecimal: x16

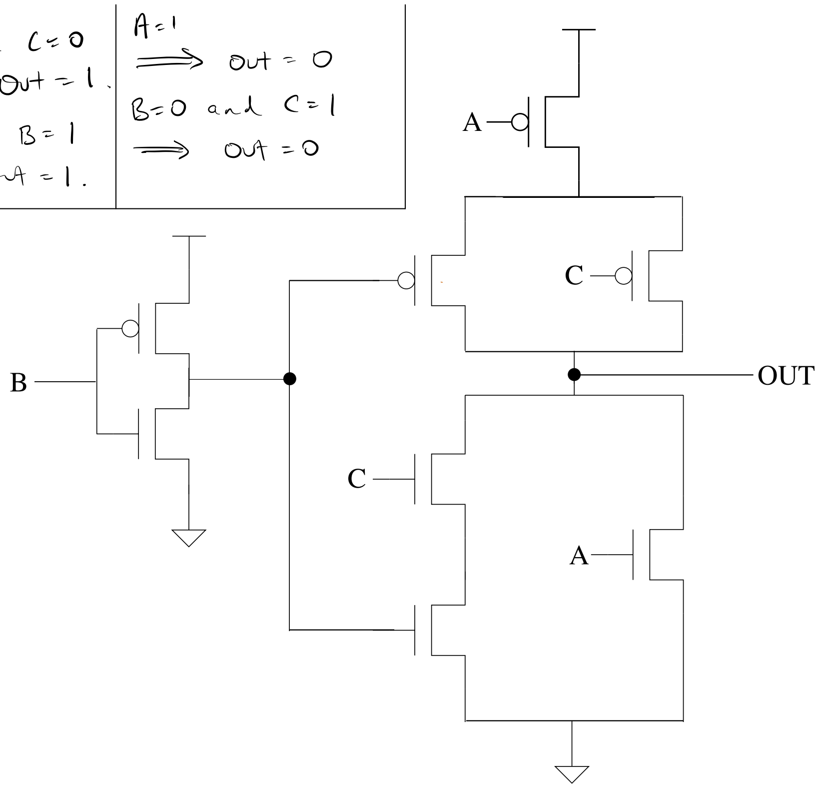
Part d. (5 points): For the transistor-level circuit below, fill in the truth table.

A=0 and C=0 $\Longrightarrow$ Out = 1.

A=0 and B=1 $\Longrightarrow$ out = 1.

A=1 $\Longrightarrow$ out = 0

B=0 and C=1 $\Longrightarrow$ out = 0



A	B	C	OUT
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

Name:____

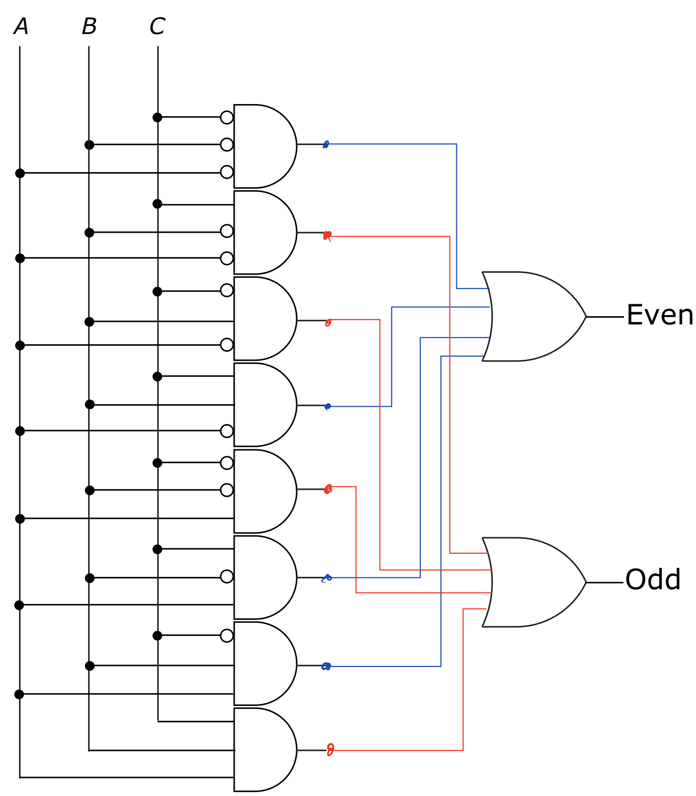
Part e. (5 points): We want to design a logic function with three 1-bit inputs (A, B, C) and two 1-bit outputs (Even, Odd) with the following behavior:

- Even = 1 if an even number of input bits are set to 1.
- Odd = 1 if an odd number of input bits are set to 1.

Complete the truth table for this logic function. (Note: zero is an even number.)

A	B	C	Even	Odd
0	0	0	1	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	0	1

We can implement this logic function with a PLA. Complete the PLA shown below by connecting the necessary outputs of the AND gates to the appropriate inputs of the OR gates.



Name:____

Problem 2. (15 points):

An LC-3 program is stored in memory locations x4000 to x4005. Note that the branch instruction in memory location x4002 has an unspecified PCoffset9, denoted as X.

Address	Instruction
x4000	0101 000 000 1 00000
x4001	0001 000 000 1 00001
x4002	0000 011 X
x4003	0001 000 000 1 00010
x4004	0001 000 000 1 00011
x4005	1111 0000 0010 0101

The program starts executing with PC = x4000.

Your job: In the table below, for each value of X, answer the question: "Does the program halt?" (Yes or No). If your answer is "Yes", answer the question: "What value is stored in R0 immediately after the instruction at x4004 completes execution?" If your answer is "No", put a dash in the column labeled "Value stored in R0".

X	Does the program halt?	Value stored in R0
000000010	Yes	1
000000001	Yes	4
000000000	Yes	6
111111111	No	—
111111110	Yes	x8005 $(=-2^{15}+5)$

Name:____

Problem 3. (15 points):

Rock/paper/scissors is a two-person game which most of you played in your childhood. During each round of the game, the two players simultaneously form one of three SHAPES (rock, paper, or scissors) with an outstretched arm. If one player shows rock and the other shows scissors, the player showing rock wins (since rock crushes scissors). Similarly paper covers rock, and scissors cuts paper. If both players choose the same shape, the round ends in a draw. The players continue to play rounds as long as they wish to play the game.

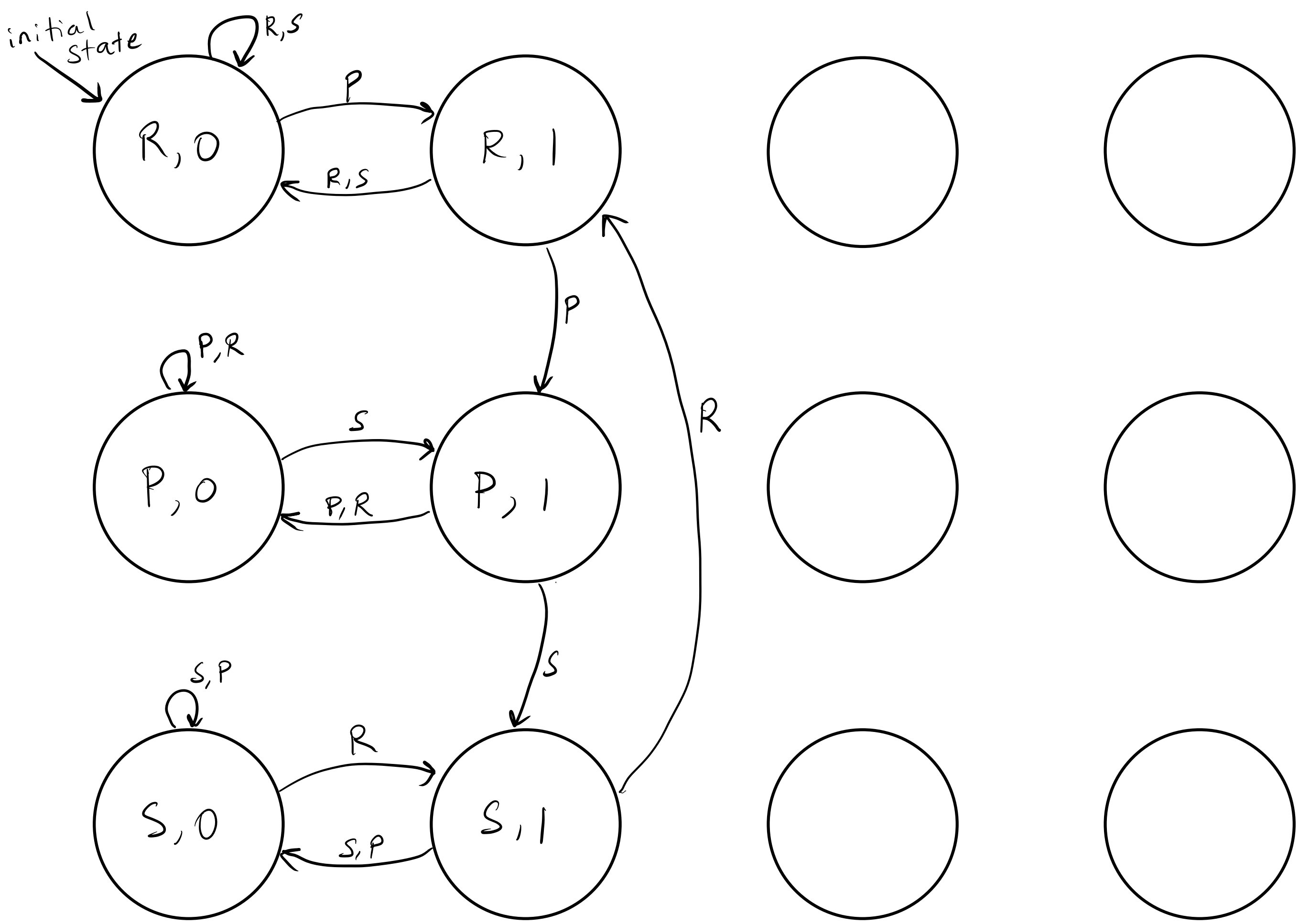
Dr. Patt has devised a strategy for rock/paper/scissors. The strategy is as follow:

- He always plays “rock” in the first round.
- He plays the same shape again unless he lost the last two rounds. In that case, he plays the shape his opponent played in the previous round.

Your job: Construct a finite state machine for what Dr. Patt should play in the current round. The input is the shape that his opponent plays in the current round. On your finite state machine designate “rock” as “R”, “paper” as “P”, and “scissors” as “S”. (We have provided 12 states. You will not need all of them. Use as many as you need.)

Note: Your finite state machine will not provide any signals to designate the winner or loser of a round. It will also not designate when the players stop playing the game.

Hint: If Dr. Patt plays the same shape as his opponent, the round ends in a draw (i.e., Dr. Patt did not lose that round).



Name:____

Problem 4. (20 points):

The program below sets the nth bit of the 16-bit bit vector stored in x30A0. Before the program executes, a separate program stores n into R0, which designates the bit in the bit vector that this program is asked to set. The program loads the bit vector into a register, sets the nth bit, and stores the bit vector back into x30A0. You will note that the program is missing four instructions.

For example, if R0 contains the value "2" and the bit vector 0101000001010000 is contained in memory location x30A0, then the program will store the bit vector 0101000001010100 in memory location x30A0.

Your job: Complete the program by filling in the instructions needed in memory locations x3002, x3007, x300A, and x300B to make the program work as specified above.

Alternative solutions for x3007:

1. 0000 001 111111101
2. 0000 001 111111100
3. 0000 101 111111100
4. 0000 011 111111100
5. 0000 111 111111100
6. 0000 001 111111011
7. 0000 101 111111011
8. 0000 011 111111011
9. 0000 111 111111011

Address	Value	
x3000	0010 001 010011111	R1 ← M[x30A0]
x3001	0101 010 010 1 00000	R2 ← ∅
x3002	0001 010 010 1 00001	R2 ← R2+1
x3003	0001 000 000 1 00000	R1 ← R1 + ∅
x3004	0000 010 000000011	Branch to x3008 if Z=1.
x3005	0001 010 010 0 00 010	R2 ← R2 + R2
x3006	0001 000 000 1 11111	R0 ← R0 − 1
x3007	0000 101 111111101	Branch to x3005 if R0 ≠ ∅.
x3008	1001 011 010 111111	R3 ← NOT(R2)
x3009	0101 001 001 0 00 011	R1 ← R1 & R3
x300A	0001 $\boxed{001}$ 001 0 00 010	R1 ← R1 + R2
x300B	0011 $\boxed{001}$ 010010100	M[x30A0] ← R1
x300C	1111 0000 0010 0101	HALT.

These could be any register, as long as they are the same.

Note: at x3003, it is also acceptable to have put R0 ← NOT(R2). In that case, the answer to part b would be "No, since the program does not have a Halt instruction."

Name:____

Problem 5. (25 points):

Memory locations x3000 to x3005 contain six instructions. Note they are partially filled in.

Also acceptable at x3002: 0101 000 010 000 001

Address	Instruction
x3000	0010 0010 1111 1111
x3001	0010 0100 1111 1111
x3002	0101 000 001 000 010
x3003	1001 000 000 111 111
x3004	0011 0000 0000 0000
x3005	0000 0000 0000 0000

If memory location x3100 is the only memory location that contains x0FFF and memory location x3101 contains xCFDA and the program starting at location x3000 starts executing, the contents of the PC, MAR, MDR, IR, R0, R1, and R2 **AFTER** each instruction is executed are shown below:

	PC	MAR	MDR	IR	R0	R1	R2
Initially	x3000	----	----	----	x0000	x0000	x0000
After instruction at x3000	x3001	x3100	x0FFF	x22FF	x0000	x0FFF	x0000
After instruction at x3001	x3002	x3101	xCFDA	x24FF	x0000	x0FFF	xCFDA
After instruction at x3002	x3003	x3002	x5081 x5042	x5081 x5042	x0FDA	x0FFF	xCFDA
After instruction at x3003	x3004	x3003	x903F	x903F	xF025	x0FFF	xCFDA
After instruction at x3004	x3005	x3005	xF025	x3000	xF025	x0FFF	xCFDA

Part a. (20 points): Fill in the remaining entries in both tables above.

Part b. (5 points): If the program starts executing with the initial value of the PC = x3000, will the program halt (Yes/No)? Explain in 15 words or fewer. Overly-generic answers will earn zero points.

Yes, the instruction at x3004 stores HALT to x3005, which is then executed.