EECS 270

Final Exam

For all questions, show all work that leads to your answer.

Problem #	Possible Points	Points Earned
1	5	
2	15	
3	17	
4	18	
5	25	
6	20	
Total	100	

1. *Short Answer/Multiple Choice: 5 Points Total*

(a: 1 pts) In base-3 arithmetic, what arithmetic operation is performed when all digits are shifted twice towards the least significant bit, dropping the two least significant bits and adding zeros for the two most significant bits?

_____

(b: 2 pts) Assuming equally optimized implementations, which combinational circuit will have fewer gates? (circle one)

A. A 16-bit ripple-carry adder

B. A 16-bit comparator (output = 1 if and only if A == B)

C. A 16-to-4 encoder

Note: When counting gates, you can use inverters and 2-input AND, NAND, OR, NOR, XOR, XNOR gates (and no other gates). Larger gates will have to be decomposed into 2-input gates.

(c: 2 pts) Which of the following equations can be used as a feedback equation in an LFSR? (circle all correct answers)

D. $F = X_0 + X_1 + X_2$

E. $F = X_0$ XOR X_3

F. $F = X_0 \bullet X_2 \bullet X_4$

G. $F = X_0' \bullet X_1' \bullet X_2 \bullet X_3'$

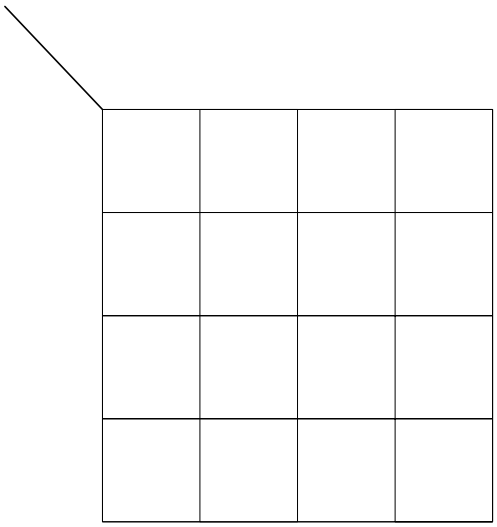
2. *K-Maps: 15 Total Points*

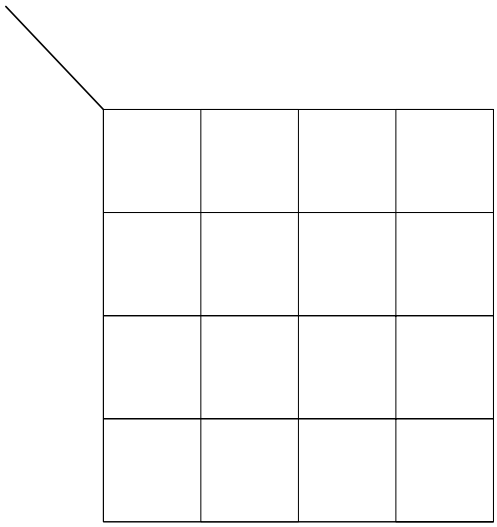
Consider the following function:

$$F = \Sigma_{ABCDE}(0, 1, 2, 3, 10, 11, 14, 15, 19, 23, 24, 25, 27, 28, 30) + d(8, 12, 17, 26, 31)$$

(a: 5 pts) Complete the K-map below for this function and neatly circle all prime implicants.

(b: 5 pts) Mark all essential prime implicants with an X.





(c: 5 pts) Construct the minimal SOP expression for F.

3. *Combinational Circuit Design: 17 Total Points*

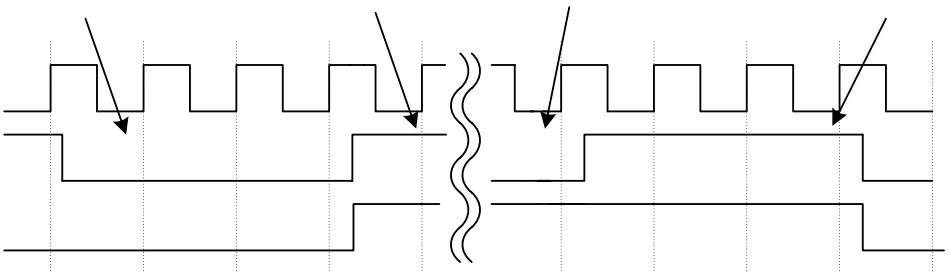
Construct a circuit with four inputs, A_1, A_0, B_1, B_0, and three outputs, S_2, S_1, S_0. Let A_1A_0 represent a binary number A from 0 to 3, B_1B_0 represent a binary number B from 0 to 3, and $S_2S_1S_0$ represent a binary number S from 0 to 7. The outputs of your circuit should always be such that S = A + B. Your design may use any of the following, each instance counting as one component: 2-to-4 decoders, 4-to-2 encoders, 8-3 encoders, and AND/OR/INVERTER gates. ***Neatly*** construct your circuit, ***using as few of the listed components as possible, with a maximum of 20 components.***

4. *Sequential Design with Shift Registers: 18 Total Points*

Consider a serial channel with a packet-oriented data transfer protocol. An arbitrary number of data bits can be sent in a packet, with 000 denoting the beginning of the packet, and 111 denoting the end of the packet. Assume that 000 and 111 will never occur as part of the data bits. The task for this problem is to implement sequential circuits with one input DATA_IN, and two outputs DATA_OUT and VALID.

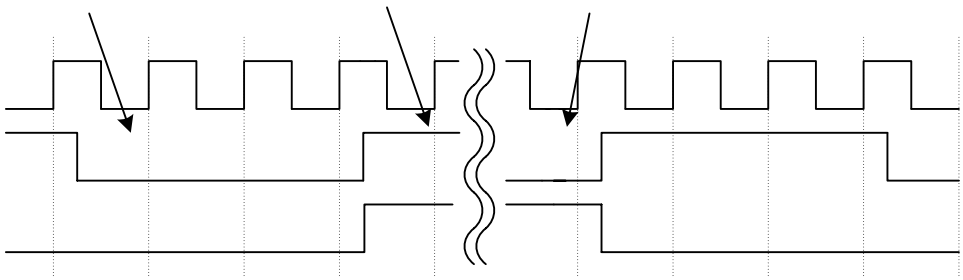
DATA_IN will read a serial data stream, and DATA_OUT will be equal to DATA_IN, but it can be delayed as many clock cycles as necessary. VALID will denote when data should be read within the packet, according to the following:

(a: 6 pts) Assume that VALID should transition to 1 *after* 000 appears on DATA_OUT, and transition to 0 *after* 111 appears on DATA_OUT, as illustrated in the timing diagram below: *(note that DATA_IN is not shown)*



Design this circuit, using a 4-bit shift register with serial-in and clock inputs, a SR latch, and any combinational gates required. Make sure that the shift direction of your shift register is clear.

(b: 6 pts) Assume that VALID should transition to 1 *after* 000 appears on DATA_OUT, and transition to 0 *before* 111 appears on DATA_OUT, as illustrated in the timing diagram below: *(note that DATA_IN is not shown)*



Design this circuit, using a 6-bit shift register with serial-in and clock inputs, a SR latch, and any combinational gates required. Make sure that the shift direction of your shift register is clear.

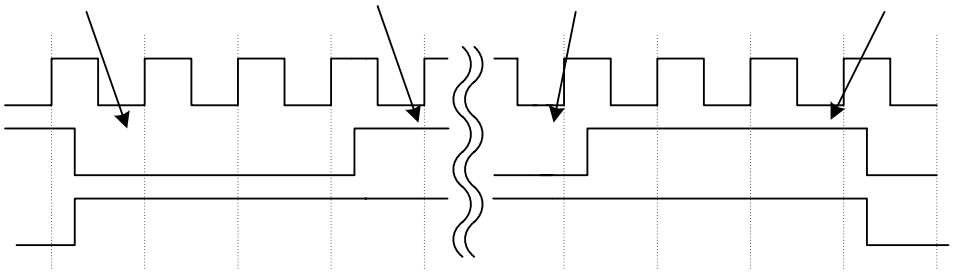
VALID should remain lo
for all of 000 marker.

CLK

DATA_OUT **0**

VALID

(c: 6 pts) Now assume that VALID should transition to 1 *before* 000 appears on DATA_OUT, and transition to 0 *after* 111 appears on DATA_OUT, as illustrated in the timing diagram below: *(note that DATA_IN is not shown)*



Design this circuit, using a 6-bit shift register with serial-in and clock inputs, a SR latch, and any combinational gates required. Make sure that the shift direction of your shift register is clear.

First bit of 000 mark
VALID should be h

CLK

DATA_OUT **0**

VALID

5. *"Silicon Chef" Sequential Design: 25 Total Points*

Your task is to design a control circuit for a microwave oven with a timer. The cook time, up to 1023 seconds, is loaded into a 10-bit binary *down*-counter, with ENABLE and CLR_L inputs. The outputs of the timer would be conceptually connected to a display, but you do not need to show this in your design. Your design can assume that the timer value is already loaded into the counter, i.e., you will not need to worry about the load and data inputs of the counter.

(a: 8 pts) Design a circuit with one active-high input, START, and one output, GENERATOR_ON. When START is asserted, GENERATOR_ON should be asserted, and remain asserted until the counter value reaches 0. At that time, GENERATOR_ON should be de-asserted, and the counter should stop counting. Your design may use the 10-bit *down*-counter, one or two latches of any type, and any combinational logic gates (AND/OR/etc…).

(b: 8 pts) Extend your circuit to incorporate two additional inputs: DOOR_OPEN, and CLEAR (both active high). When DOOR_OPEN is asserted, GENERATOR_ON should immediately be de-asserted, the counter should be paused, and the START input should be blocked. Once DOOR_OPEN is de-asserted, asserting START will continue the cooking sequence. When CLEAR is asserted, the counter should be cleared.

(c: 9 pts) Reusing your design from part (b) as a block diagram, extend the controller to account for four power modes, encoded with two bits as follows: boil (00), cook (01), reheat (10), and defrost (11). While the microwave generator can only be on or off, you can regulate the average power by *periodically* turning the generator on and off:

In boil mode, GENERATOR_ON should be asserted at all times.
In cook mode, GENERATOR_ON should be asserted 3/4 of the time.
In reheat mode, GENERATOR_ON should be asserted 1/2 of the time.
In defrost mode, GENERATOR_ON should be asserted 1/4 of the time.

The power setting is stored in a two-bit register. Your design does not need to take into account the manner in which values are loaded into the register; it only must take into account the actual contents of the register. In addition to the register and circuit block from (b), your design can also use one 2-bit binary counter, one MUX of any size, and less than 6 one- or two-input combinational gates.

6. *Lab: 20 Total Points*

(a: 8 pts) How many machine cycles (clocks) are required to implement the following operations?

Instruction fetch: _____

Immediate fetch: _____

Yr generation: _____

Ym generation: _____

BRN execution: _____

What register holds the result of a Ym generation? _____

What data paths are required for a Yr generation? _____

What operation in the BRN instruction loads the negative flag register?

_____

(b: 5 pts) True/False
The ALU is used to increment the PC.

True **False**

R0 is always the destination register and R1, R2, and R3 are always source registers.

True **False**

The instruction register contains the source and destination registers Ra and Rb.

True **False**

The RTL expression for STR is MEM[Ym] ← Ra

True **False**

The RTL expression for Yi is Yi ← Rb + n

True **False**

(c: 7 pts) Consider the following ABEL code segment:

```
Line 1      State S0:
Line 2      N = 0; M = 1;
Line 3      If (A) then S1 with { x := 1; y:=0 }
Line 4      Else if (B & C==0) then S2 with { x:=0; y:=1 }
Line 5      Else S3
```

What lines are Moore Outputs? _____

What lines are Mealy Outputs? _____

When debugging this code you discover that line 4 does not transition to S2 when B is 1 and C is 0. Rewrite the transition statement so that this transition will now occur:

_____