Homework 3 Logisim

You may use all components under wiring and all components under gates **except** for the parity checkers. From the memory tab you may use all Flip-Flops and the register. You may use no other components besides these unless specified in the problem.

A Note About Testing and the Halt Pin

All of the testing circuits have a halt pin that will **attempt** pause the testing on the first test you get wrong. I say attempt because there are two things that prevent it from working exactly like that

1. I can only pause the testing **after** you miss a test so when it does pause it will show the inputs for the next test case. For example suppose that the input you miss is 5. The tester will then pause once you get to test 6. So to see the actual test you missed you'll have to reset the circuit and then tick through until you get to test 5.
2. Your circuit keeps taking input even though I'm not giving you new ones. Because the clock keeps ticking this means that you'll keep transitioning states based on the paused input. So let's say that you were supposed to output a 1 on this test but you output a 0. The tester will stop feeding you new input but your circuit will keep on taking in the old paused input over and over. If this eventually causes you to output a 1 the tester will then move on. This means that your grade with the halt pin set to 1 and set to 0 could be different. We will grade your submission with halt set to 0.
3. Debugging sequential circuits is a pain and while not perfect this is the best I could figure out to help you locate any errors you might have in your solution.

1. (15 points) Implement a 3 bit synchronous up/down counter that stops counting when the minimum/maximum count is reached. For example if the count is at 111 and you are told to count up you should stay at 111. Count_Up and Count_Down will never both be 1 at the same time. If both Count_Up and Count_Down are 0 the count should not change.
    1. Inputs: Count_Up, Count_Down.
        1. If Count_Up is high the value increases by 1 unless the count is at 111.
        2. If Count_Down is high the value decreases by unless the count is 000.
    2. Output: Count
2. (15 points)Implement a **Moore** model circuit that calculates the even parity over a **group** of 4 bits. The circuit should output a 1 if after the last bit in the group is received, the number of 1's in the group of 4 bits is even and 0 at all other times.
    1. Example inputs/output
        1. Input: 0111 1000 1010
        2. Output: 0000 0000 0000 1
3. (Credit for this problem goes to Sean Davis 20 points) Implement a **Mealy** model circuit to control a coin operated vending machine. This machine only accepts quarters, dimes, and nickels. Coins are inserted until a total of 30 cents or more is deposited. X1 = quarter, x2 =

dime, X3 = nickel. Only 1 coin is deposited at a time. The output signal, z1 indicates whether merchandise should be provided or not; z1 = 0 indicates no merchandise and z1 = 1 means merchandise should be given. Coincident with the last coin input, the change outputs are to be set. Assume that the machine can give a dime, z2 = 1, and/or a nickel, z3 = 1. If the customer does something silly like entering a quarter followed by a quarter, correct change does not have to be provided but the maximum amount of change should be given. Note that only 1 input can be high at a time.