Mission Overview

Your mission is to modify the “Load Before Launch Sequencer (LBLS) that you built last week for the USS Harry S. Truman to handle more jets. As you recall, the LBLS system monitors the number of planes in the queue by counting them as they roll over a sensor in the deck. The new system must:

- Count the number of jets in the launch queue, from 0 to 12. Use one momentary switch as the deck sensor to add a plane. Use the second momentary switch to indicate a jet being launched (subtracted) from the queue.
- Refuse another “count up” if the number of jets in the queue is already 12. Similarly, if the number of jets in the queue is already 0, the counter must not accept another “count down”.
- Indicate the number of jets in the queue using two 7-segment decimal display units.
- Build a *status panel* with three LEDs:
  - Status light 1 is "on" only when the queue is empty (Q = 0).
  - Status light 2 is "on" only when the is neither full nor empty (0 < Q < 12).
  - Status light 3 is "on" only when the queue is full (Q = 12).
- You must also use one of the regular SPDT switches as a “clear” switch, which sets the counter to zero.
- Whenever a jet is added to the queue, run the motor clockwise and light a green LED. The motor should stay on for one second. Also, the motor may start after a small delay to allow the circuit to use a 1 Hz clock signal. This motor indicates that the large elevator is coming up to the flight deck from the hangar deck.
- Whenever a jet is launched, run the motor counter-clockwise and light a red LED. The motor should stay on for one second. Also, the motor may start after a small delay to allow the circuit to use a 1 Hz clock signal. This motor indicates that the large elevator is returning to the hangar deck from the flight deck, to prep another plane.

Input Summary: The final system must use 1 switch, and 2 momentary switches.
Output Summary: The final system must use 3 red LEDs, two 7-segment displays, one motor, a yellow LED, and a green LED.

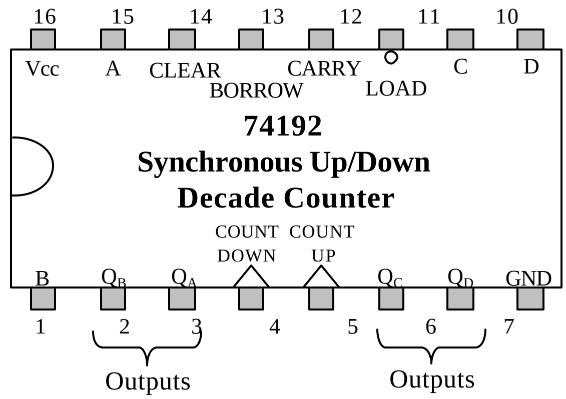
Equipment: You may use any chips that we have learned about this semester.

Suggested Process:

1. Build a 00 to 99 up/down counter using two 74192 chips, two displays, and two momentary switches).
2. Build the "Clear" feature (requires only additional wires, and no new chips).
3. Build the 3 status lights (can take a variety of forms).
4. Use the status light outputs to restrict the counting range from 00 to 12 (can be done with only two gates!).
5. Add the motor control system (see page 3).

Discussion of New Equipment

Last week, we used a group of JK Flip-Flops to count by ones. Specifically, we counted the number of times that a switch was thrown. As you might expect, people don't really use JKFF's to count stuff. Instead, we have a specialized chip called a 74192 "up/down counter" (see layout).

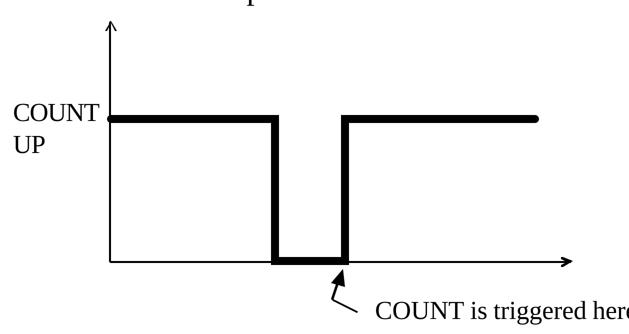


Obviously, these chips must have a power pin (pin #16) and a ground pin (pin #8). Like the JKFF, they have an edge-triggered CLK (pin #5). However, this input is activated by a *rising* edge, not a falling edge. In other words, this input, called "COUNT UP", is activated when it *changes* from low to high (↑).

The outputs of the chip, $Q_DQ_CQ_BQ_A$, are found on pins 7, 6, 3, and 2. These can be connected directly to a 7-segment display decoder package. These 4 outputs represent a "BCD" (Binary Coded Decimal), with D being the most significant bit (MSB), and A the least significant bit (LSB).

This chip has a couple of other features. Like the JKFF, this chip has a CLEAR input (pin #14; **active high**), which lets you force all four outputs to low, indicating a count of zero. Internally, this chip is made from 4 Flip Flops connected together, so it has 4 separate PRESETs (pins #15, 1, 10, and 9). That means you can force the chip to accept any values of ABCD that you want; you could apply a 0101 (which is a decimal 5), or 1001 (a decimal 9), or any other value. The preset lines are called ABCD (note that they are different than Q_A, Q_B, Q_C, and Q_D). However, merely changing the inputs to ABCD does not itself activate these presets. The chip never looks at the preset inputs unless the LOAD input is also active. This LOAD input is pin #11, and it is **active low**. So, whenever LOAD = 0, the chip forces $Q_A \leftarrow$ A, etc. Note that the LOAD input is *not* edge-triggered. If you keep LOAD low, then the counter continually looks at the inputs ABCD, and will keep adjusting Q_A to match A, until you turn off the LOAD input. In any case, we won't be using LOAD today.

Another cool feature of the chip is that it can also count *down*. If you send a rising edge trigger to the input at pin #4 ("COUNT DOWN") instead of to pin #5, the chip *reduces* the value of the BCD represented by $Q_DQ_CQ_BQ_A$ output by 1. You cannot use the COUNT UP input unless the COUNT DOWN input is held high. Similarly, you cannot use the COUNT DOWN input, unless the COUNT UP is held high. This means that the "normal state" for each input must be high. That is, when you are *not* counting, both count inputs should normally be high. Since the "count" inputs are high most of the time, and because they also require a rising edge, the input to COUNT UP must look like this:



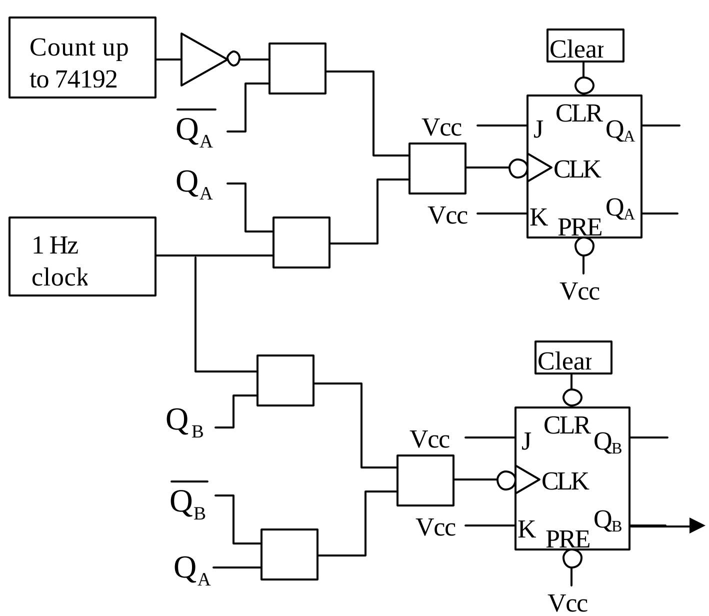
Suppose we are counting up (using pin #5), and the output at one time is a 7_{10} (0111_2). Now, we give a rising edge (↑) to pin #5. The output changes to 8_{10} (1000_2). The next ↑ gives us a 9_{10} (1001_2). The next ↑ gives us a 0_{10} (0000_2) for the output. This kind of counter is therefore sometimes called a "decade" counter, because it counts *ten* digits, instead of 4, 8, or 16. This counter chip is specifically designed to be used with the 7 segment displays.

Again, if the output is currently at 9_{10} (1001_2), and we activate the COUNT UP pin, the output goes to zero. While we're discussing this "rollover" behavior, I want to draw your attention to another really cool feature on this chip, called "CARRY". Whenever the total output rolls over from nine to zero, the CARRY output changes from 0 to 1. You could connect this CARRY output to the COUNT UP input on a second 74192 chip, which is itself connected to a second 7-segment display. Then, when you send a bunch of rising edges to the first counter chip, you might see the display like this: 07, 08, 09, 10, 11, 12, etc. You can make a chain of as many 74192 chips as you, each one storing the value of one digit of the output.

The final feature of this chip is called the "BORROW" output, which is similar to the CARRY output. Suppose the output of a pair of two decade counters is currently 43; if you count down on the LSD, the output changes to 42, then 41, then 40. So far, only the LSD has changed. The next time we count down, the LSD will change from 0 to 9, and we desire the display to read 39. However, unless we give the system a way to alert the "4" that it needs to change to a "3", we'll just get a 49. To solve this problem, the BORROW output from the LSD can be connected to the COUNT DOWN input on the next higher digit's 192 chip. BORROW becomes active whenever the output changes from a 0 to a 9.

Motor Control

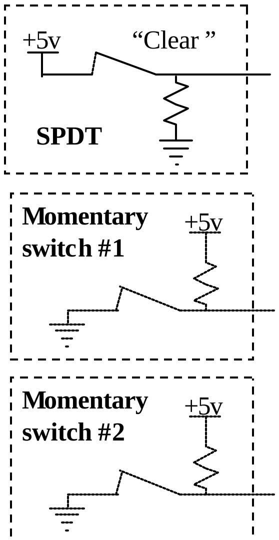
As with lab 3, you need to build two control signals "A" and "B" to control the motor. As with that lab, each relay should be active when its control signal is low. The following circuit can be used to create control "A", although you'll need to determine what gates to use to replace the rectangular boxes! (*Hint*: all six are either AND or OR gates. If you can't figure out which is which, you might be able to simulate this circuit in Digitalworks to discover the solution by trial and error). A similar circuit is needed for control "B".

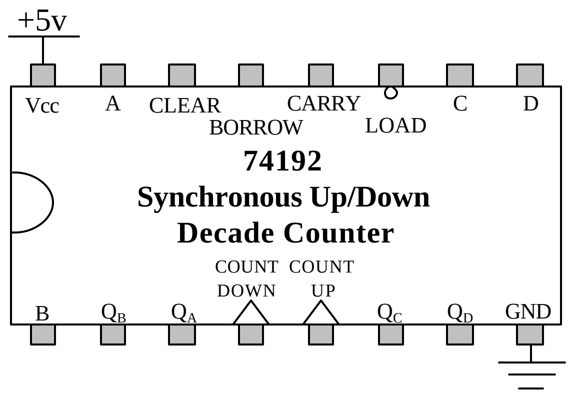


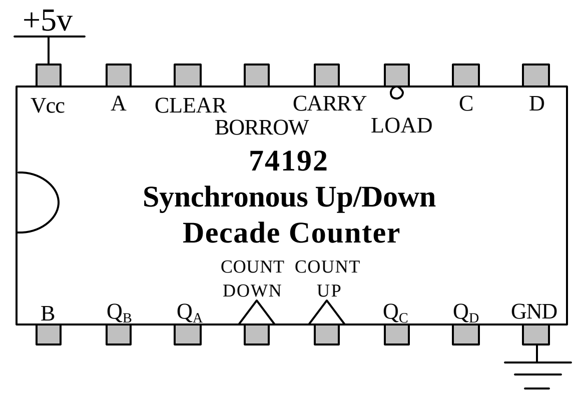
Name: ____________________________________

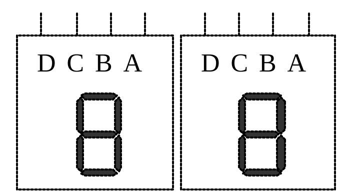
You MAY NOT write on this sheet in pen!

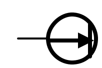
Your circuit:

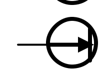


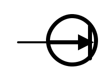







"Full"


"Some"


"Empty"