Mission Overview

Your mission is to make an improved version of the "Load Before Launch Sequencer" (LBLS) that you built last week, so that the USS Harry S. Truman can 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.
- Ignore add signals if there are already 12 jets in the queue. Similarly, ignore launch signals if the queue already contains 0 jets.
- Indicate the number of jets in the queue using two 7-segment decimal display units.
- Build a *status panel* with three red 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).
- Use one of the regular SPDT switches as a "clear" signal, which sets the counter to zero.
- Whenever a jet is added to the queue, run a motor clockwise and light a green LED. The motor should stay on for exactly one second, but may start after a small delay, to allow the circuit to use a 1 Hz clock signal for timing. (This motor represents the large elevator coming up to the flight deck from the hangar deck.)
- Whenever a jet is launched, run the motor counter-clockwise and light a yellow LED. Timing requirements are the same as above. (This motor represents the large elevator is returning to the hangar deck from the flight deck, to prep another plane. Except… wait, that doesn't make any sense; multiple planes can be brought up into the queue before one is launched. Aw, shucks, it's just a cool Digital Electronics thing…)
- While the circuit is waiting for a motor to start, or while a motor is running, both "count up" and "count down" signals should be ignored.

Input Summary: The final system must use 1 switch, and 2 momentary switches.
Output Summary: The final system must use two 7-segment displays, 3 red LEDs (on the PAD), one motor, and a yellow and a green LED (discrete components).
Equipment: You may use any chips that we have learned about this semester, plus the handy 74192 chips described below.
Suggested Procedure:

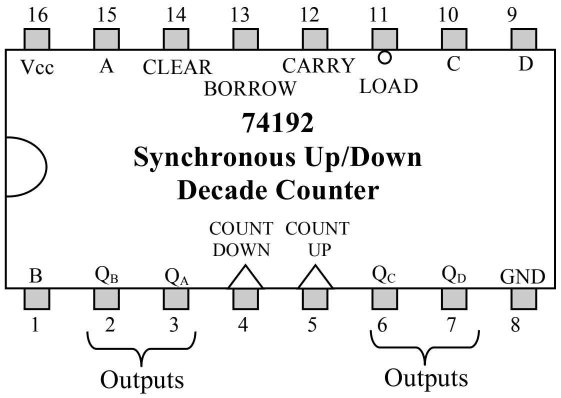
1. [8 points] Build a 00 to 99 up/down counter using two 74192 chips, two displays, and two momentary switches).
2. [3 points] Build the "Clear" feature (requires only additional wires, and no new chips).
3. [6 points] Build the 3 status lights (can take a variety of forms; don't forget MUXes).
4. [7 points] Use the status light outputs to restrict the counting range from 00 to 12 (can be done with only two gates!).
5. [4 points] Add the motor control system (see page 3).
6. [4 points] Use the "arm" signal from the motor control to block counting during motor operation (only three more gates!).

New Equipment

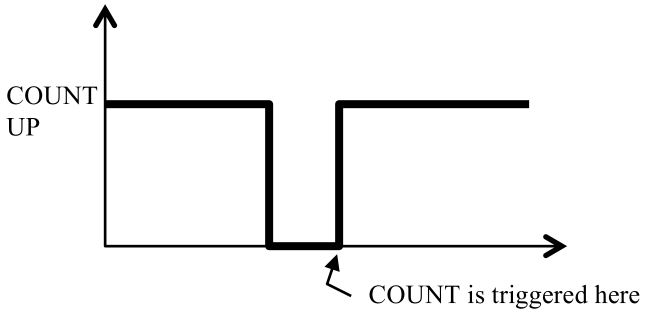
Last week, we used a group of JK Flip-Flops to count by ones. Specifically, we counted the number of times that a switch was thrown. As you might expect, people don't often use JKFF's to count. Instead, we have a specialized chip called a 74192 "up/down counter" (see layout).

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

Like the JKFF, it has an edge-triggered input (pin 5), called "COUNT UP." However, this input is activated by a *rising* edge, not a falling edge. When this input *changes* from low to high (↑), the outputs increment to the next BCD digit.



The chip can also count *down*. If you send a rising edge trigger to the "COUNT DOWN" input (at pin 4 rather than pin 5), the chip *reduces by 1* the value of the BCD digit represented by $Q_DQ_CQ_BQ_A$. You cannot use the COUNT UP input unless the COUNT DOWN input is held high, and 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 be high. Therefore, the COUNT signals must be short downwards pulses like this:



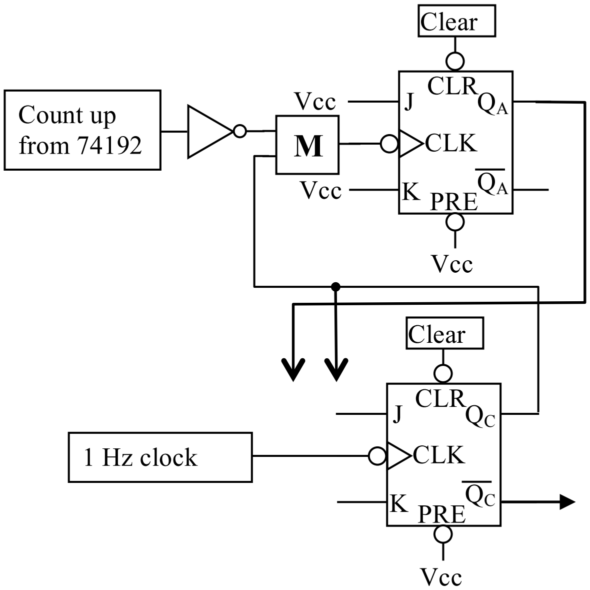
Like the JKFF, this chip has a CLEAR input (pin 14, **active high**), which lets you force the count to zero (all four outputs to low). Internally, this chip is made from 4 Flip Flops connected together, so it has 4 separate PRESETs (pins 15, 1, 10, and 9). That means you can force the chip to any value that you want. The preset lines are called DCBA (note that they are different from Q_D Q_C Q_B Q_A). However, these preset inputs have no effect unless the LOAD input is also active (similar to a D-latch ENABLE). This LOAD input is pin #11, and it is **active low**. So, whenever LOAD = 0, the chip forces Q_A to A, etc. Note that the LOAD input is *not* edge-triggered; if LOAD stays low, then the counter continually holds the outputs equal to the presets, and count signals have no effect. In any case, we won't be using LOAD today.

Suppose we are counting up, and the output at one time is a 8_{10} (1000_2). The next rising edge (↑) on pin #5 causes the output to change to 9_{10} (1001_2). The next ↑ after that causes the output to roll over to 0_{10} (0000_2). This kind of counter is therefore 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.

So if the output is currently at 910 (10012), and we activate the COUNT UP pin, the output goes to zero. Now notice 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 would then act as the next most significant digit. A bunch of rising edges to the COUNT UP of the first counter chip, might cause the combination to run through 08, 09, 10, 11, 12, etc. You can make a chain of as many 74192 chips as you like, 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 "C" and "CC" to control the motor. (You can review that lab to see how they should control the relays. Remember that these control signals are active low.)

The following circuit can be used to create control "C", although it is not quite complete. The basic idea is that the upper JK-FF creates an "arming" signal, indicating that the action has been requested but not yet completed. Q_A=1 allows the clock to turn the lower JK-FF on and then off, at which point the upper FF is disarmed.

For the worksheet:

1. What logic gate at **M** will result in a triggering edge only when Q_C is low?
2. What wiring will ensure that the lower FF will only go on (Q_C=1) when the arming signal is on (Q_A=1)?
3. What wiring will ensure that the lower FF will turn off (Q_C=0) whenever it has been on (Q_C=1) during the previous second?

A similar circuit is needed for control "CC." Note that, conveniently, the "arming" signal is high precisely at the times when we want to ignore further count signals.