

Part I

0. You investigate here two counters: the '163 (a fully synchronous, presettable binary counter) and the '191 (an up/down binary counter with asynchronous load). (Note: some protoboard 'debounced pushbuttons' will produce multiple pulses each push. If your counting seems anomalous, try clocking with the TTL function generator.)
1. Press in, and power "at the counters", the following TTL chips (from top to bottom): 74191 (up/down counter), 7400 (quad NAND), and two 74163s (counter).
2. Using the top '163, tie ENABLE T high, take ENABLE P, CLEAR, LOAD, and the data inputs D, C, B, and A to data switches and the outputs: CARRY OUTPUT, Q_D , Q_C , Q_B , and Q_A to LEDs. Begin by clocking with a pushbutton (remember the pull-up resistor). Note that to count, ENABLE P, CLEAR, and LOAD must all be high. Does the '163 count positive-going edges or negative-going edges? Describe when CLEAR and LOAD need to be set and when the corresponding action occurs. Describe exactly how you would use LOAD to set the counter to 7. Carefully sketch (stacked scope trace format) the CARRY OUTPUT, CLOCK and counter value (converted to decimal) as you move towards and beyond the top count. Move the clock input from the pushbutton to the protoboard's function generator (on TTL). Set the clock rate to ~ 1 Hz and observe the up-counting action.
3. Make a decade counter by detecting a 9 count output using a NAND and feeding the NAND output to CLEAR. Does your decade counter briefly hit 10 before clearing? Use another NAND to detect a 10 count and look for glitches with a logic probe. Make 7,8,9 cycle by enabling a load of 7 following the detection of 9. Draw these circuits!
4. How can the carry be used to achieve synchronous 8-bit counting shared between two '163s? Note that using the carry as a clock to the second chip fails as the second clock ticks one count too soon. (On your above scope trace, label where such clocking would occur.) Inverting the carry would work, but would lead to a ripple counter. If two ICs are to be synchronous they must use exactly the same clock. (Hint: If the chips are clocked by the same signal, but one chip's count changes and the other's count does not, the inactive chip must be disabled.) Make a synchronous binary counter with 8 bits of output. Now make your two-163 counters have a top count of 20. Show the result to your instructor. Draw the circuit! (Your solution should be easily modifiable to make any number, e.g., 160, the top count.)
5. Investigate the behavior of the '191 as in 2. Note that LOAD must be high and ENABLE G low to count. Note that there is no CLEAR. Don't worry about the two types of carry: MAX/MIN and RIPPLE CLOCK. Describe how the up/down pin affects the counting. What sort of edge triggers the '191? What evidence shows that the '191 is asynchronous whereas the '163 is synchronous?

Part II

0. In this part you will investigate a higher level of function integration available in the 74143. This chip is a 4-bit counter, a 4-bit latch, and a seven-segment decoder/driver all in one package. This chip is designed to drive common-anode displays (N.B., *not* the common-cathode displays we used in the last lab). In a common-anode display, the high side of all the segments are internally connected together and that common-anode pin needs to be connected to +5 V. An individual segment is lit by connecting the low side of that segment “to ground.” *Recall that you never actually connect a segment between ground and power as it will burn out in a quick fizzle.* The current must be somehow limited (in this case to about 15 mA). This current-limiting feature is built into the '143.
1. After removing the chips from Part I, press in (top to bottom) and power up (at the corners for the chips) a '08 (quad AND), a '143, a seven-segment display, another '143 and another display. Read the spec sheet and decide how to wire-up the various inputs so that the '143 will continuously display and count the clock input. Make a table reporting the logic level required on each input for this normal counting. Does the '143 count positive or negative edges? Note particularly the PARALLEL COUNT ENABLE INPUT, CLEAR, and the LATCH STROBE INPUT. Describe how/when each of these works. As you did for the '163, carefully sketch (stacked scope trace format) the MAX COUNT, CLOCK and counter value (converted to decimal) as you move towards and beyond the top count.
2. Figure out how you can chain your two '143s to get a two-decimal-digit count. Provide a brief circuit diagram of your solution. Is your circuit synchronous or asynchronous?
3. Make a simple stop watch using the AND as a clock gate. Wire-up CLEAR and LATCH STROBE INPUT as in a normal stop watch (i.e., include a lap-time, a reset, and start/stop options). (If you want to be fancy and include a pushbutton toggle-action start/stop, you'll need a JKFF.) Demonstrate the circuit to your instructor.