Activity 3.2.2

Asynchronous Counters: Small-Scale Integration (SSI) Modulus Counters Using PLTW S7

INTRODUCTION

In the last activity we saw how easy it was to design **asynchronous counters** using either the D or J/K flip-flop. However, these designs had two big limitations.

First, the count limit had to be a power of 2 (2, 4, 8, 16, 32, and so on).

All counts also started or ended at a count of 0. In the real world, we frequently need to set the count limit to some arbitrary value (10, 25, 85). More often than not, the starting or ending value will not be 0. For this reason we must design asynchronous **modulus** counters.

An asynchronous modulus counter, or mod-counter, uses the addition of simple combinational logic to a standard asynchronous counter to set the count limit and starting point. In this activity we will simulate and build a mod-5 counter that has a starting count of 1.

This activity will also introduce using a clock signal with a PLD.

EQUIPMENT

- Circuit Design Software (CDS)
- Digital MiniSystem (DMS)
- PLTW S7 FPGA Module
- #22-gauge Solid Wire

RESOURCES

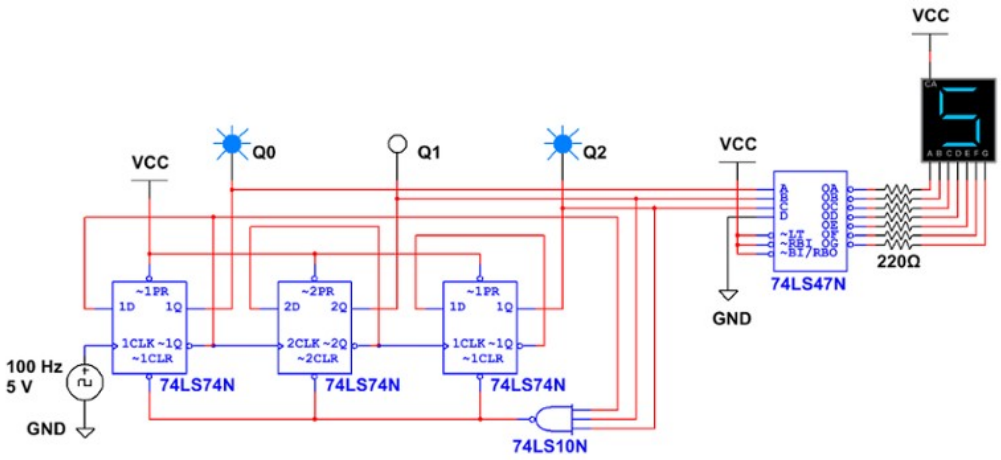
Asynchronous Counter

Procedure

Simulation (Design Mode)

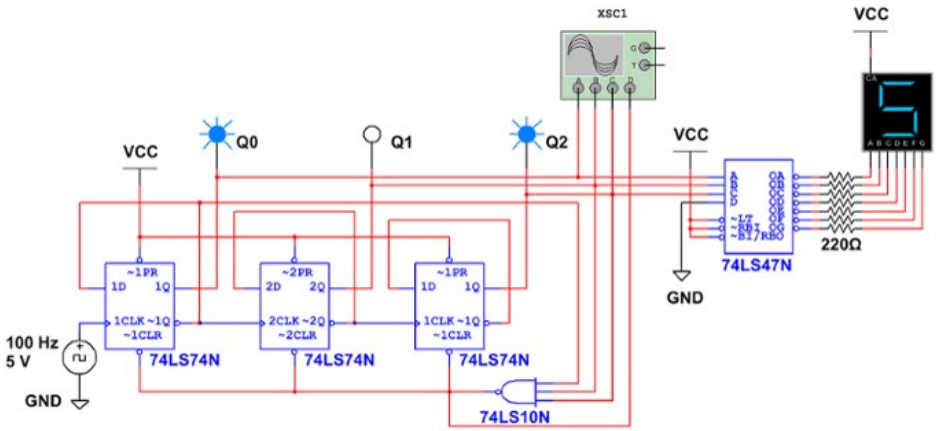
1

The circuit below is a 3-Bit Mod-6 **Up Counter** implemented with 74LS74 D flip-flops. In this design the count will be displayed on a common anode seven-segment display using a 74LS47 encoder.
This design will count from 0 to 5 and then repeat (Mod 6).



3-Bit Mod-6 Up Counter with D Flip-Flops

a. Use Design Mode of the CDS to enter the 3-Bit Mod-6 Up Counter. Add a four-channel oscilloscope to monitor the signals Q0, Q1, Q2, and the output of the NAND gate. Run the simulation and capture a full count cycle (0–5) of the signal. Verify that the circuit is working as expected. If the results are not what you expected, review your circuit and make necessary corrections.

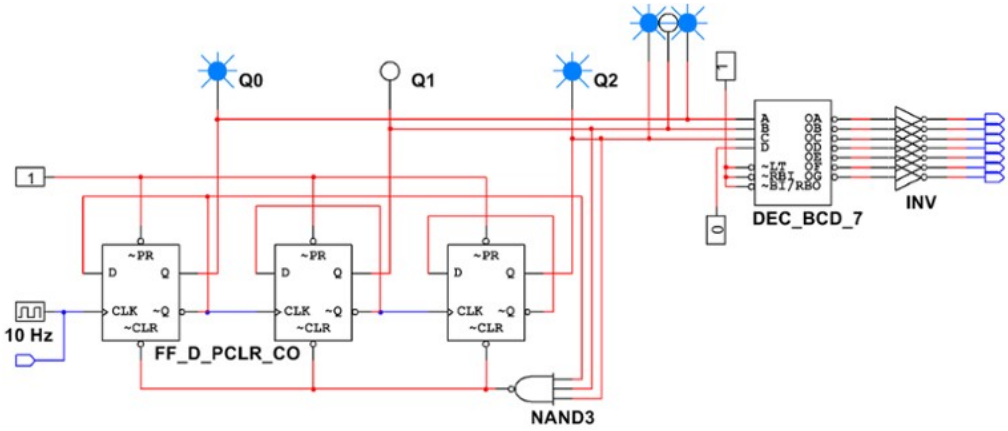


b. Adjust the time-base of the oscilloscope to zoom in to the point in time when the counter changes from a count of 5 (101) to 0 (000). Obtain a printout of these waveforms.

c. Make the modifications to this circuit to change the count from 2 (010) to 6 (110). This is now a Mod-5 Up Counter with a start of 2 (010). Run the simulation and verify that the circuit is working as expected. If not, review your circuit, make necessary corrections, and retest. Use a 74LS48 and a common cathode seven-segment display for this simulation in preparation for the next step.

Simulation (PLD Mode)

2 The circuit below is the same 3-Bit Mod-6 Up Counter implemented with 74LS74 D flip-flops (only it is created in PLD Mode).



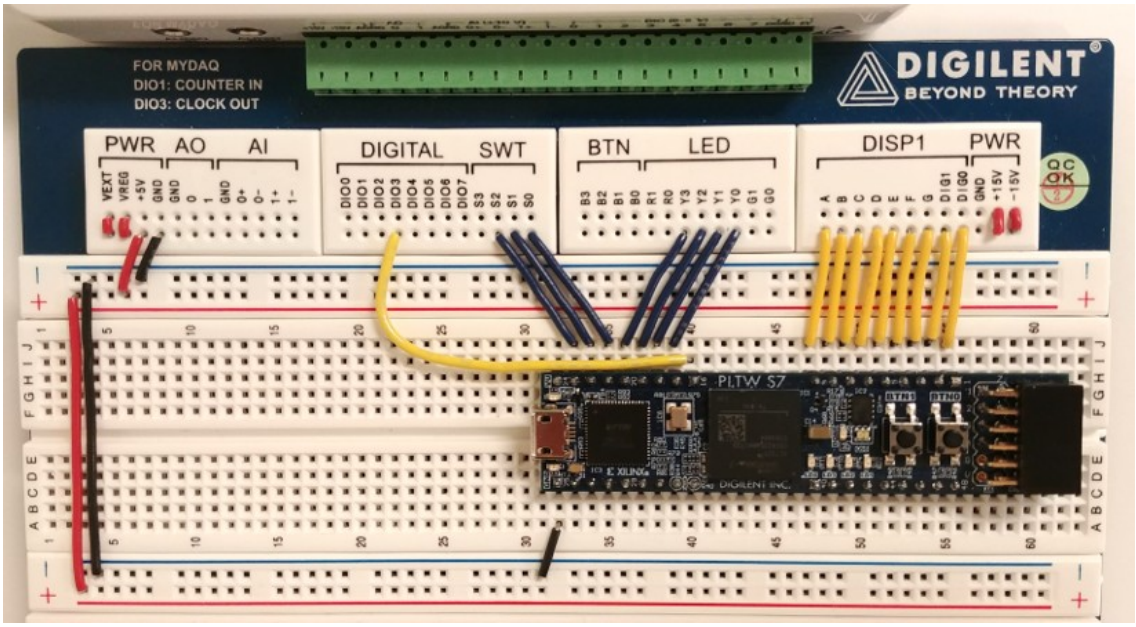
3-Bit Mod-6 Up Counter with D Flip-Flops

3. The counters in Design Mode were implemented with 74LS74 D flip-flops and a 74LS47 display driver. ***These exact components do not exist in the PLD Mode.***

4. In the PLD mode, the 74LS74s will be replaced with **FF_D_PCLR_CO** (Flip-Flop, D, Positive Edge Clock, Clear, Preset, Complementary Output).

5. In the PLD mode, the 74LS47 Display Driver must be replaced with the **DEC_BCD_7** (Decimal, Binary Coded Decimal, Seven-Segment Display). This is the only BCD-to-decimal decoder available in the PLD mode. The **DEC_BCD_7** is the PLD Mode equivalent of the 74LS47 Display Driver (designed for common anode SSDs). No decoder exists in PLD Mode to represent the 74LS48 (designed for common cathode). Since the DLB has common cathode seven-segment displays, inverters must be added between the decoder and the outputs.

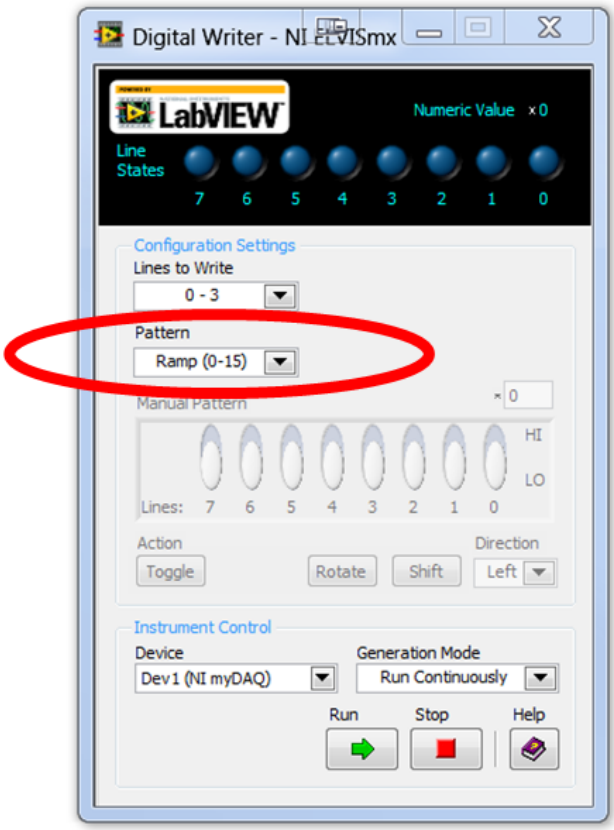
Clock Signal



6. The DMS implementation will use the myDAQ DigOut Digital Writer to send a clock signal to the PLD. (See the image below.)

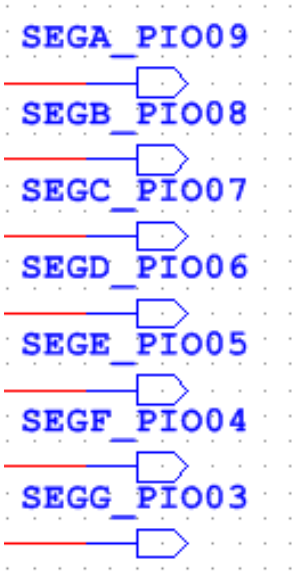
7. DIO3 from the myDigital Protoboard should be wired to PIO16 of the PLD Module (PLTW S7 FPGA Module). PIO16 is the only input designated for CLK in.

8. External Clock. Using the NI™ Elvis Digital Writer on the myDAQ, generate a clock signal. (Write lines 0–3 Ramp 0–15.)

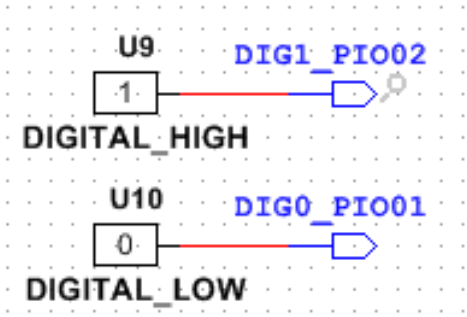


9. To adjust the frequency of the clock, move the wire from DIO3 to DIO2 and so on to speed up the clock signal.

10. This PLD Mode design will have:
    a. Map the design's seven outputs (A, B, C, D, E, F, and G) to PIOs shown in the image.

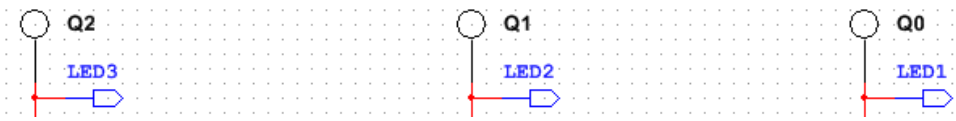


b. Add a digital low and digital high to DIG0 and DIG1 respectively as shown.



c. Seven-segment displays that are Common Cathode. Add inverters to the outputs of the DEC_BCD_7 decoder used to convert BCD to decimal, or the output will be the exact opposite of what you want.

d. Three outputs from the three flip-flops (Q2, Q1, and Q0) going to the DMS's LEDs (yellow). These signals will be useful in the event that your design needs to be debugged. See the image.



e. Use a wire to connect this PIO16 pin to the CLK signal on the DMS since the only PIO that can accept the CLK signal is PIO16.

11 Create and test the 2-to-6 Mod-5 Up Counter on the DMS.

12 Have your teacher verify that the circuit works by capturing and presenting the circuit as instructed by your teacher.

CONCLUSION

1 The asynchronous modulus counters examined in this activity were all designed using D flip-flops. Design a 3-Bit Mod-6 Up Counter (0–5 count) using the 74LS76 J/K flip-flop.

2 Explain why a counter with an upper limit of five (101) resets at six (110).

3 When designing a Mod-13 Up Counter (0–12 count), how many flip-flops are needed?

4 What values must be on the Qs of the flip-flops to cause the counter to reset? Explain.

Proceed to next activity