HARDWARE 2:
POSTNET AND XS3

STUDENT RUBRIC

DEMO RUBRIC

Hardware 2 is a massive step up in difficulty from hardware 1, and will almost certainly be a source of frustration for those of you who have not yet gotten comfortable with wire stripping, circuit building, etc. This is the first lab, as well, where you must design the circuit in its entirety. Start by using Karnaugh maps to minimize each equation, then create a LogicWorks mock-up, and then you can build the circuit itself. It may behoove you to "demo" to TAs at each step to ensure that you're on the right track.

When testing your hardware, **do NOT dismantle the whole circuit if it doesn't work.** Try using your logic probe to test each connection to the faulty output(s) in reverse order to find the source of the error. Confirm that you're using the right chips, that they're in the right direction, and that you are using the correct diagrams to assist in your wiring. Most of all, keep it organized. This is a reasonably large design, so creating a mess will result in much headache for you and the TA trying to debug it.

Completion Requirements:

- ✓ Create your equations from a series of K-maps.
- ✓ Create a LogicWorks mockup of your design before building.
- ✓ Finally, confirm that the hardware implementation is without error. Look at some non-valid inputs too. We may consider partial credit if only one or two states aren't working, but don't count on it.

REPORT RUBRIC

This is another full lab report. Make sure to include everything indicated below. Additionally, while there's no grade for specific formatting conventions, keep your submission professional looking and well organized, with the sections labelled and your photos well-lit. **We will take off points if the end result is highly messy, illegible, or replete with spelling and grammatical errors.**

Scoring (out of 3 points):

- ✓ **[0.7 points]** Theory:
  - [0.4] Display an understanding of minimizing equations with Karnaugh Maps. How are they set up and how do you use them? Why do we need "don't care" states (boxes marked with an X)?
  - [0.3] What are POSTNET (2 out of 5, 74210 weighting) and XS3? Keep it brief.

- ✓ **[1.4 points]** Deliverables:
  - [0.3] Include your full table of POSTNET (VWXYZ) to XS_3 (DCBA) conversions from decimal 0 to 9.
  - [0.8] Show all four of your 5-variable Karnaugh maps and the work you did to minimize each equation. This can just be a photo of your work, but try to keep it readable. Also include the final, minimized equations for D, C, B, and A.
  - [0.3] Make a POSTNET symbology bar code for your home area. If you'd rather not say, you can select any location.

- ✓ **[0.4 points]** Discussion section. Should conform to standard lab report guidelines.

- ✓ **[0.6 points]** Questions:
  - [0.2] Q1: How many non-valid input codes are there? Your answer to this question should correctly derive how many total states exist, and extrapolate how many of those are not used.
  - [0.4] Q2: Would your design be simpler if there was no zero input? You do not *need* to do another set of K-maps to answer this question, although you may choose to… it can be answered theoretically, although ultimately you must provide some proof. Recall that the POSTNET "zero" input is 11000. By "how does your answer conflict with your design", we mean that you should compare the complexity of your original equations to what you would have without zero.