Project Proposal: Hamming Encoder & Decoder
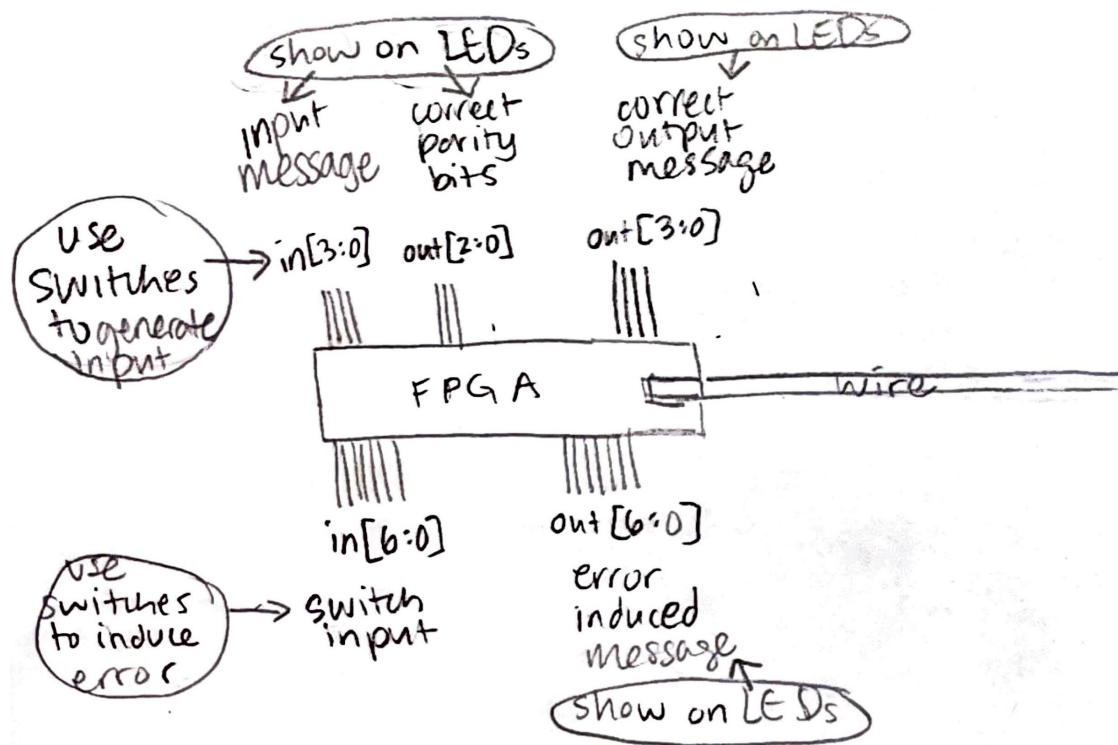Computer Architecture
Emma Mack and Lilo Heinrich

Abstract

We will make a hamming encoder and decoder with switches to set the message and the corruption, and LEDs to visualize the correct message, the corrupted message, and the corrected message. An FPGA will handle all of the logic.

Block Diagram

Project Plan

Our MVP is a working Hamming (7,4) encoder and decoder circuit, where you can flip the switches to create a 4-bit message, introduce a 1 bit error, and that bit error will be corrected in the final output message.

We will create the physical circuit showing led output, writing five verilog modules (encoder, error inducer, decoder, corrector, and main) and a writeup about hamming codes, error correction, and what it is we implemented.

Stretch goal: Researching and doing a writeup (and maybe a schematic / block diagram) for methods for multi-bit error correction. These are entirely different algorithms!

The steps to the MVP are:
- An encoder which lights up LEDs based on calculated parity bits. We can write a test file to simulate this and since there are only 16 input possibilities we can test to exhaustion.
- Electrically hooking up the switches to introduce an error and creating the module to turn positive values from a switch into errors. We can test this visually since it is simple.
- The decoder module: we will write a test file that takes the corrupted signal and outputs the error indicators. We will test this with enough examples that we feel confident it works.
- The corrector module: Calculates the original (uncorrupted) signal. We will test this by visually checking that the corrected output is the same as the uncorrupted input.

Pitfalls / questions: We predict that setting the pins on the fpga and writing our own main will be the trickiest since we haven't done it before. I don't foresee the logic or the breadboarding to be an issue. We will come for help if we have pin-setting problems, but we don't have questions at the moment.

Bill of Materials

- FPGA
- 18 LEDs
- 18 pull-down resistors
- 3x 4-position dip switches (we checked that the stockroom has them)

Sources

This entire project is walked through in this video: https://youtu.be/h0jloehRKas