

CPSC 121: Models of Computation
Lab #7: Circuit Design: Coding Theory

Objectives

Coding theory is the study of methods used in determining the correctness of transmitted data. Think about the game of telephone in which one person whispers a message to another, who then passes that message on to the next person and so on and so forth. The “fun” in the game arises when the message makes it around the room and back to the original sender, now changed in some (humorous) way from the actual message, presumably having been altered somewhere along the way. Messages such as emails or phone conversations, which are sent on computer networks are subject to these kinds of transmission errors. In this lab, you will be designing and implementing an error-checking circuit that catches mistakes made when data is relayed.

1 Prelab: Coding Theory

The two biggest categories of transmission errors are *substitution errors* and *transposition errors*. **TODO (prelab): Define these two terms and give an example of each. Also define a *checksum*.**

1.1 Prelab: design

TODO (prelab): Research and design an error-checking circuit that can identify substitution errors or transposition errors. Provide a circuit diagram for your design and cite any sources you use. To help you get started, consider parity checking and repetition codes. You are welcome to use any of these for your circuit. If you use the parity check your circuit must be able to identify errors in a message of at least four parallel bits, where one of the bits is the checksum. For all other designs, aim for a circuit that can handle three serial bits.

Please do not design an overly complex circuit that will take you too long to implement or test.

1.2 Prelab: methods

TODO (prelab): Determine or describe a method to test each of the following:

1. Will this circuit catch substitution errors?
2. Will this circuit catch transposition errors?

1.3 In-lab: Your circuit

TODO: Implement your circuit on Logisim, and then test it using your methods from the prelab. Record your results/observations.

TODO: Scale your circuit up: modify your circuit so that it can detect errors on messages one bit larger than your current implementation. Then test your circuit again using your own methods. Record your results/observations. Show your circuit to your TA.

TODO: What errors did you find your circuit could catch? Do you confirm or reject your hypothesis? How well did your circuit scale up? Discuss your answers with your TA.

1.4 Real-world applications

Credit cards use an error-detection scheme known as the Luhn algorithm. The codes used for identifying books, ISBN (International Standard Book Number) also uses a checksum-based scheme. **TODO: Look up these schemes: how does each one work? What types of errors will each detect?** Discuss your findings with your TA.

TODO: How does the ISBN error detection scheme compare to the Luhn algorithm? Give an example of an error the ISBN scheme would not catch that the Luhn algorithm would catch, and vice versa.

1.5 In-lab: Analysis and Discussion

In 2003, NASA began a robotic space mission to explore Mars. Two unmanned semi-autonomous rovers, *Spirit* and *Opportunity* were deployed on the surface of Mars. *Spirit* is no longer in use but *Opportunity* is still functioning (past its designed lifespan). While the rovers can operate on their own, occasionally they require human control which needs to be communicated via radio signals from Earth.

Radio signals travel at the speed of light ($c = 3 \cdot 10^8 \text{ m/s}$) and Earth and Mars are on average about $225 \cdot 10^6 \text{ km}$ apart from each other. Thus, it takes approximately 750 seconds (12.5 minutes) to send a control signal from Earth to Mars. **TODO (further analysis): How would the time delay, in between signal broadcast and reception, influence our design of how data is transmitted to the rovers on Mars?**

2 End of Lab Survey

TODO: To help us improve these labs both this term and for future offerings, complete the survey at <http://www.tinyurl.com/cs121labs>.

3 Challenge problem

TODO (challenge): Implement a small circuit that corrects errors upon detecting them. You may want to first read up on forward error correction (FEC).

4 Marking scheme

All labs are out of ten marks, with two marks for prelabs, and eight marks for in-lab work. In more detail:

- Two marks - Prelab questions
- Five marks - In-lab questions. It is one mark for your results, one for scaling up, one for your findings; and two for the real world applications.
- Two marks - Further analysis questions
- One mark - End of lab survey.

TAs may at their discretion award one bonus mark, such as for completing a challenge problem. It is expected that most students will achieve 6-8. If you feel you're heading for 0-5, **get immediate help from the TAs!**