

Introduction to Computer Communications

Programming Assignment1: Noisy Channel

Submitted by

Rony Kositsky 205817893

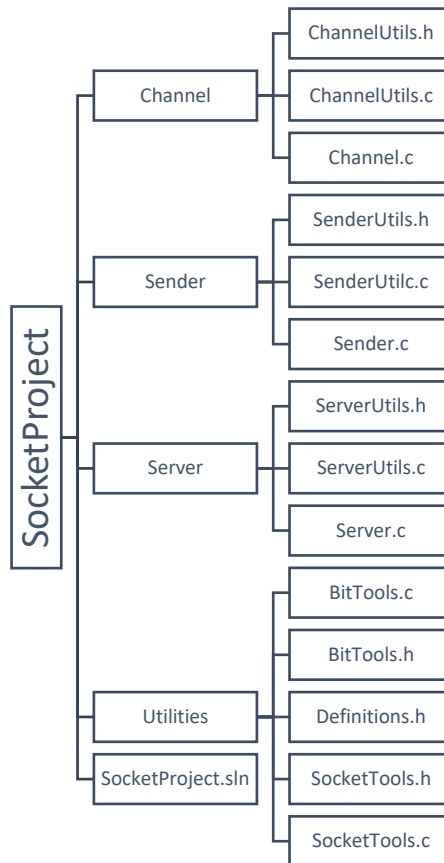
Jonathan Matetzky 205949753

1. User manual

- a. Running our project is exactly as requested in the documentation.

2. General description of the code structure

- a. Project's files hierarchy (the graph below does not contain configuration files such as '.vcxproj')



- b. We have divided the project into 3 parts; **Sender, Channel and Server**, each part has a generic repetitive structure as seen in the graph above containing relevant modules.
<part>Utils.c contains utilities which we have developed and are relevant for this certain part of the project. The **<part>Utils.h** is a header file containing constants and functions declarations. **<part>.c** implements the essence of the logic of this part of the project – includes executable code which is mainly using the designated utility function.
- c. In Addition, we have created a general **Utilities directory** which contain reusable utility functions and tools which we have been using along the different parts of the project. Among those function are implementations of logic operations such as XOR, and a function that adds parity bits to a given message. The Definitions.h header file mainly contains predefined constants such as buffers sizes and masks to extract certain bits of a given message.
- d. Code we haven't written / libraries we have used: winsock2, stdin, stdlib, string, stdbool, stdio, stdint

3. A few more points to note

- a. The ports are selected randomly and according to their availability status
- b. The channel component acts as a server which listens over IP 0.0.0.0 (according to the instruction given via the course forum discussion). The channel's address is known to the user in advance.

4. Known bugs and exercise limitations

- a. In case the message file is not in a 26-bit chunks format. It won't be able to be read.
- b. The program doesn't handle illegal arguments as was defined in the instructions.

5. Bonus extras

- a. Our implementation is independent of the length of the file, therefore meets the requirements and supports any size of file. It sends it as one message and let the OS to handle everything.