CSE 140L Lab 4

(NG Zhe Wee, A16389707)

Academic Integrity

Your work will not be graded unless the signatures of all members of the group are present beneath the honor code.

To uphold academic integrity, students shall:

- Complete and submit academic work that is their own and that is an honest and fair representation of their knowledge and abilities at the time of submission.
- Know and follow the standards of CSE 140L and UCSD.

Please sign (type) your name(s) below the following statement:

I pledge to be fair to my classmates and instructors by completing all of my academic work with integrity. This means that I will respect the standards set by the instructor and institution, be responsible for the consequences of my choices, honestly represent my knowledge and abilities, and be a community member that others can trust to do the right thing even when no one is watching. I will always put learning before grades, and integrity before performance. I pledge to excel with integrity.

(NG Zhe Wee. A16389707)

Free Response

Please answer the following questions.

1. Please explain, at a high level, how encryption using a LFSR works. (4 pts)

Word limit: 200 words

The LFSR is responsible for generating a pseudo-random bit pattern, which can have a significantly long repeating pattern, effectively resembling a random sequence. To initialize and set the state of the LFSR, an initial value is required, and the resulting random pattern will be the same for a given initial value and state. This similarity between generating random patterns and encryption keys is notable. By applying XOR with a random pattern, it becomes possible to encrypt a message. Conversely, by knowing the initial value and state of the LFSR, the encrypted message can be decoded back into its original form by reversing the XOR operation.

2. Please explain, in detail, the behavior of a LFSR. (4 pts)

Word limit: 200 words. Think about the input, output, timing, and behavior of the module.

The LFSR requires an initial value, often referred to as a random seed. As the LFSR's output is deterministic, its generated random pattern depends on its current and previous states. Since the register size is finite, the LFSR will eventually enter a looped state, resulting in a repetitive random pattern. Therefore, the pattern generated by the LFSR is considered pseudo-random.

3. How did you manipulate your address pointers (both read and write) to implement your design? (4 pts)

Word limit: 200 words

Both the read and write pointers are monitored using an enable-controlled counter. The read pointer is held at 0 until it reaches the value of pre_len - 1. This ensures that the underscore is read from the data memory. Meanwhile, the write pointer increments by 1 with each cycle, allowing the encrypted data to be written to a new address.

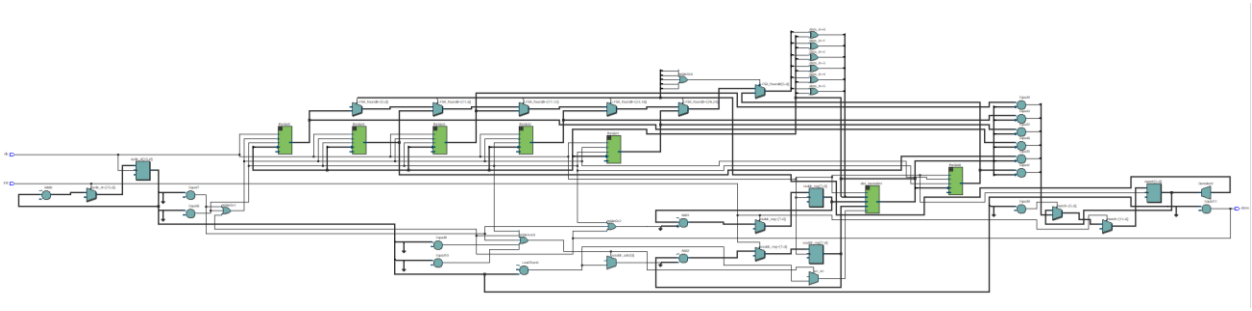
4. What is the purpose of making our messages have a preamble in our design? (3 pts)

Word limit: 200 words

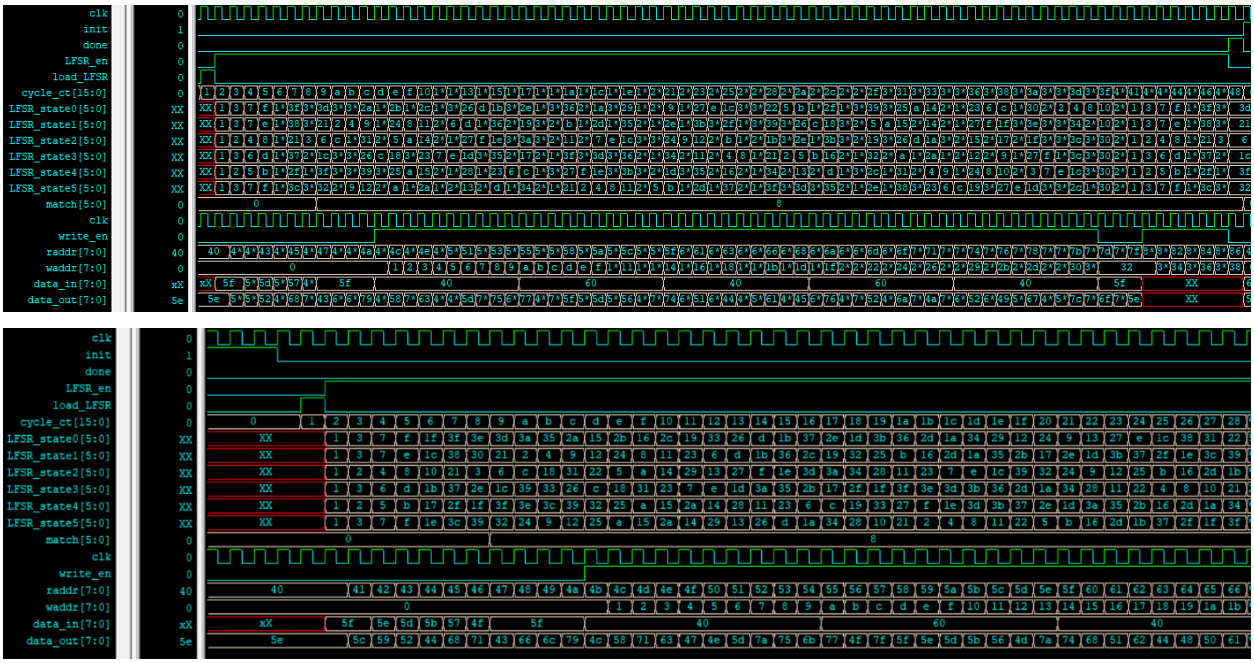
Basically, the preamble aids in the identification of the actual encrypted or decrypted message within a data stream. It acts as a reliable marker that helps the module differentiate between the message and any unrelated data. Lastly, the inclusion of a preamble enhances processing efficiency. This allows for better effenciency.

Screenshots

Screenshot of the RTL viewer top level schematic/block diagram in Quartus Or submit your Mentor Precision netlist file if using EDA Playground (5 pts)



Screenshot of your waveform viewer, including variables from both the testbench and the design under test. Or submit your Mentor Precision netlist file if using EDA Playground (5 pts)



Please edit your testbench to pipe the transcript to an output file in your submission, and name the output file “output.txt” (5 pts)

We will be looking for a text file with that name specifically, so be sure to rename it. Nothing is required in the writeup for this question.