ECE 385

Fall 2022

Final Project

Twofish Encryption/Decryption Interface

Udit Pai, Frank Cai

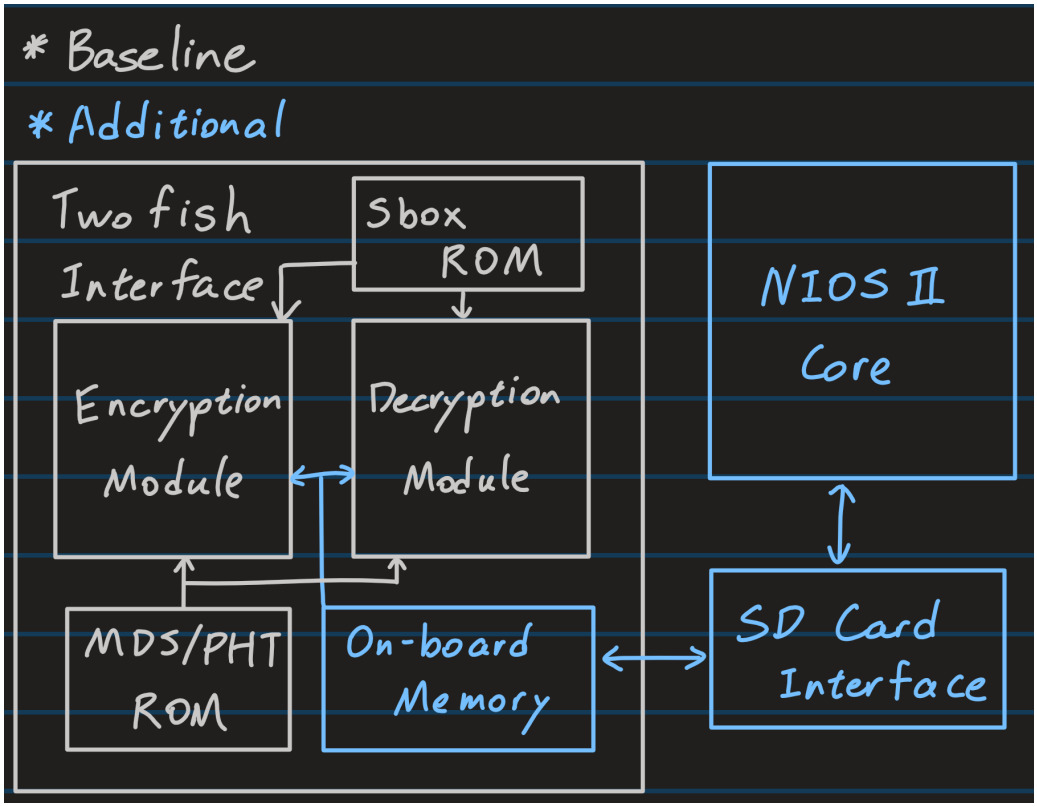
TA: Hongshuo Zhang

1. **Idea and Overview**

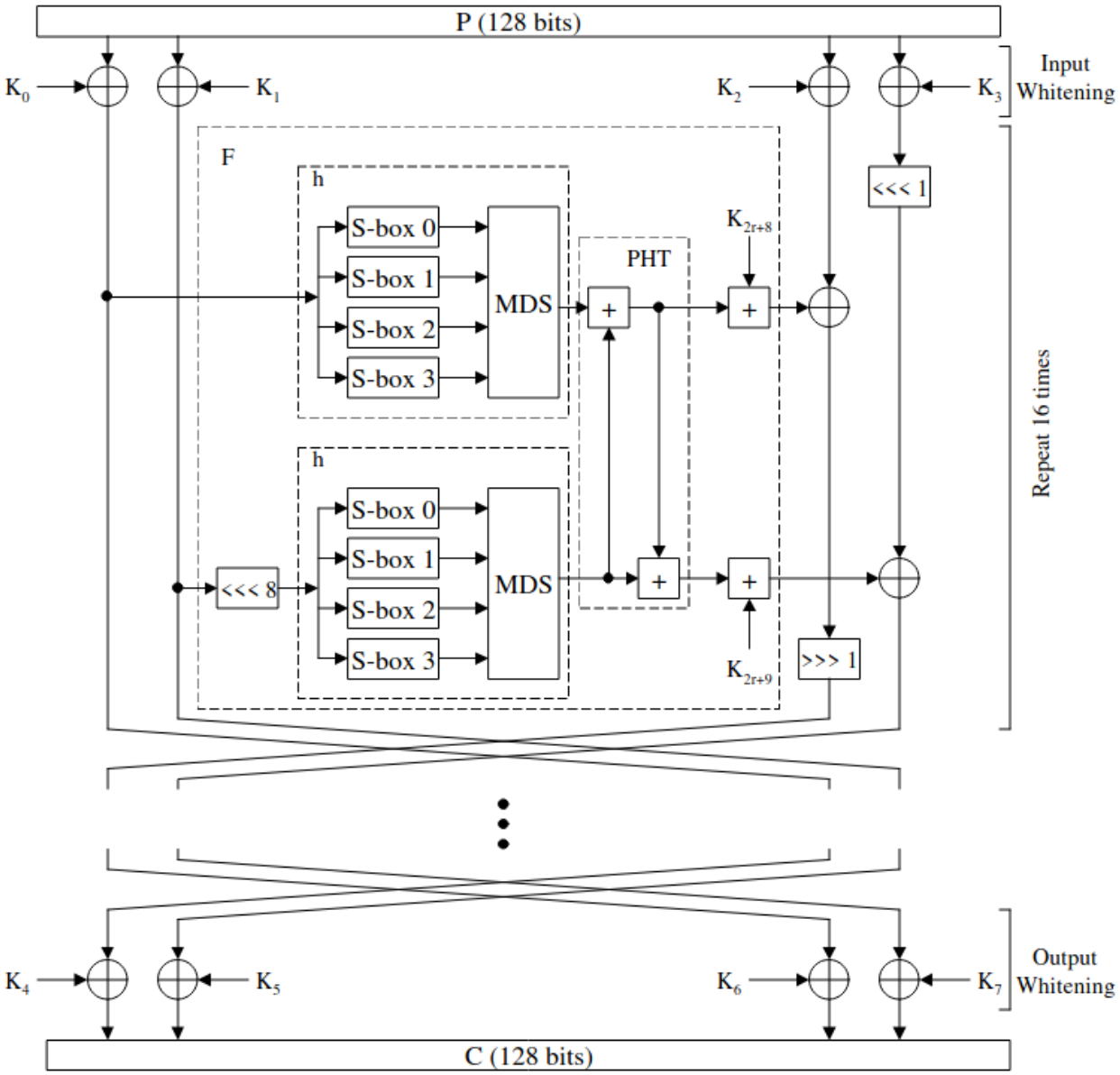
   We propose to design and implement a Twofish core interface that will be able to encrypt and decrypt raw binary data per the standard using the ECB (Electronic Code Book) method. Twofish was one of the finalists in the AES contest along with Rijndael, Serpent, RC6 and MARS. The ciphering process consists of multiple steps such as key scheduling, keyed s-box, MDS (Maximum Distance Separable) matrix, etc to cipher the given raw data. The entirety of the interface will be implemented in SystemVerilog. However, as it is just an interface, a soft-core processor such as the NIOS II will need to be used to input and output meaningful data besides ones given in simulation. The reason why we chose this project is because we believe that the FPGA is very suited for cryptography due to its customizability as well as less overhead compared to software encryptors/decryptors. At 50 MHz, the maximum throughput we can expect is approximately 400 Mbps. If we increase the clock to 115 MHz, the throughput will be at approximately 920 Mbps. We could also unroll the rounds and implement the module in purely combinational logic to achieve significantly faster throughput. However, it will be incredibly costly at about 16 times the logic.

2. **Block Diagram**

   Top Level:



Detailed encryption module:



3. List of Features

A. Baseline features

1. The interface will be capable of encrypting and decrypting according to the Twofish standard
2. As it is an interface, it will output signals signifying the completion of the process
3. The entirety of the encryption and decryption process will be able to be simulated via ModelSim

B. Additional features

1. On-board memory to store encrypted/decrypted data

2. Communication with the NIOS II to print data via stdout
3. Read and/or write data to SD card

4. **Expected Difficulty**

   We expect the baseline features to be a 8 in difficulty as the encryption/decryption algorithm is fairly mathematically and logically intensive, also given the fact that we have one week less to implement this due to complication with our proposal. It is also something outside of the scope of the teaching in ECE 385, meaning we have to start every part of this project from scratch compared to game projects where the majority of the concepts such as VGA were discussed over the course and code for VGA interface and controller were given. We believe the communication with NIOS II and on-board memory should be fairly easy and would warrant only another point in difficulty, bringing the total to 9. We also believe that reading and writing data from and to the SD card would potentially be the biggest challenge and should warrant a two point bump in difficulty.

5. **Proposed Timeline**

   Due to the rejection of our initial proposal and second proposal, we do not believe we will have any meaningful progress by this week. However, we believe that we will be able to finish all baseline features (with margin of error for bugs) and show a working simulation for the Twofish standard by Friday, 9/18, after which we can start working on the additional features. We plan to have the on-board memory finished by the end of Friday, 10/2. Given only a week's time as well as the fact that it's nearing final time, we are not confident to promise the delivery of more additional features such as NIOS II and SD card and those will only be done if we have sufficient time.