Project Report

Implementation of Cryptographic Algorithm on FPGA

Anjana sharma(2018H1400143G)

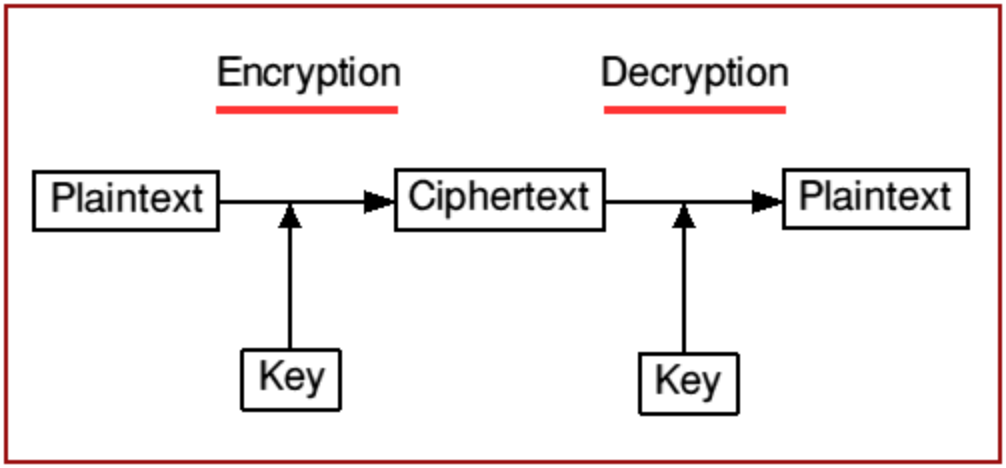
Renuka Ramrakhiani (2018H1400122G)

Abstract

Considering the complex nature of advance encryption standard (AES) algorithm, it requires a huge amount of hardware resources for its practical implementation. The extreme amount of hardware requirement makes its hardware implementation very burdensome. During the implementation of project, a FPGA scheme is introduced which is highly efficient in terms of resource utilization. In this scheme implementation of AES algorithm is done as a finite state machine (FSM). Verilog is used as a programming language for the purpose of design. Data path and control unit are designed for both cipher and decipher block, after that respective data path and control unit are integrated using structural modeling style of Verilog.

Brief Overview of AES Algorithm

In cryptography, the Advanced Encryption Standard (AES), also known as Rijndael, is a block cipher adopted as an encryption standard by the US government. The cipher was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, and submitted to the AES selection process under the name "Rijndael", a portmanteau comprised of the names of the inventors.



AES has a fixed block size of 128 bits and a key size of 128, 192 or 256 bits, whereas Rijndael can be specified with key and block sizes in any multiple of 32 bits, with a minimum of 128 bits and a maximum of 256 bits. This project has been implemented with 128 bit. AES operates on a 4×4 array of bytes, termed the state. For encryption, each round of AES (except the last round) consists of four stages.

- *Byte Substitution*

Each byte of the state is substituted with a 8-bit value from the S-box. The S-box contains a permutation of all possible 256 8-bit values. It is a nonlinear operation. The Sbox is gained by a multiplicative inverse over GF(2^8) and an affine transform.

- *Shift Row Operation*

State is the intermediate cipher result that can be pictured as a rectangular array of bytes, having four rows. In the direct ShiftRows transformation, the first line of State remains the same, the second line, third line and fourth line respectively perform circular shift right 1byte, 2 bytes, and 3bytes.

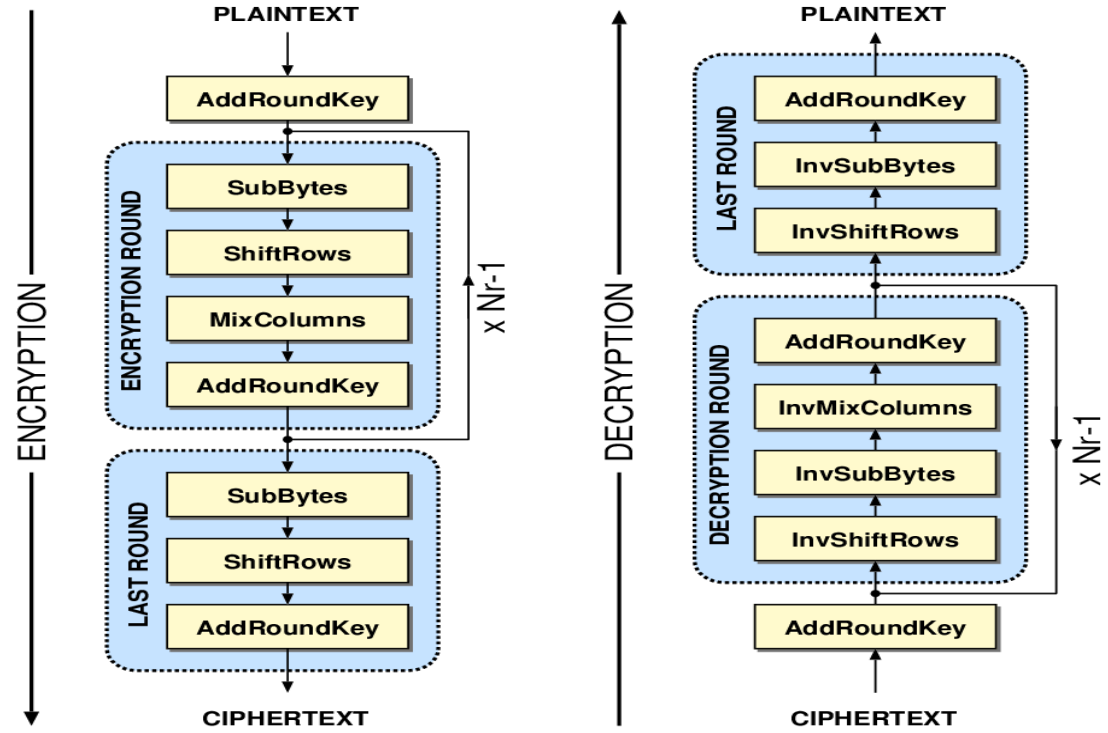
- *Mixcolumn*

MixColumn operation performs on the state column by column and performs multiplication with a predefined 4*4 matrix .As a result of this multiplication, the new four bytes in a column is generated

- *Addroundkey*

The transformation in the cipher and inverse cipher in which a round key is added to the state using an XOR operation. Round keys are values derived from the cipher key using the Key Expansion routine.

.....



.Hardware Specification

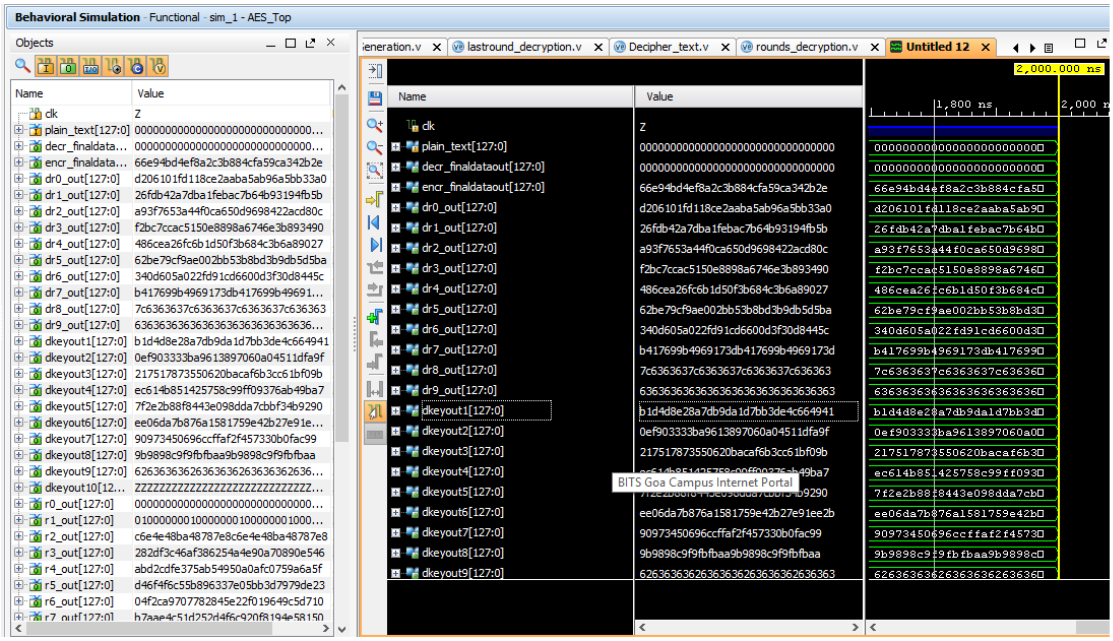
Xilinx zed b

oard Zync7000 -xc7z020dg484-1

.Software Specification:

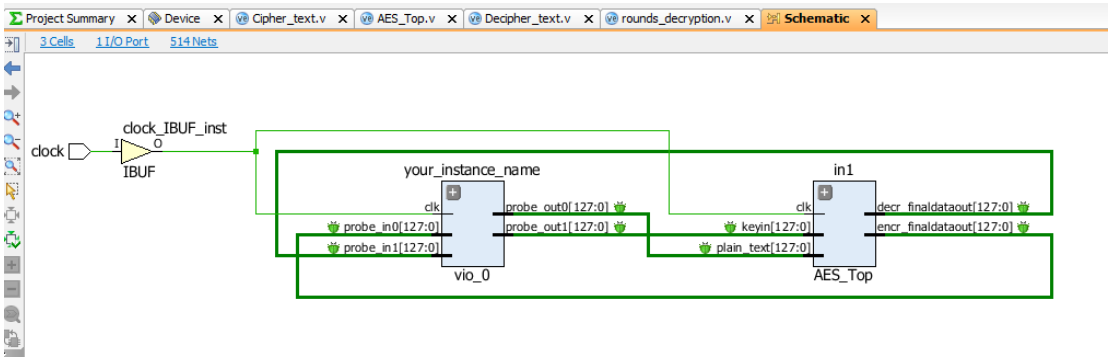
Xilinx Vivado IDE 2016.4

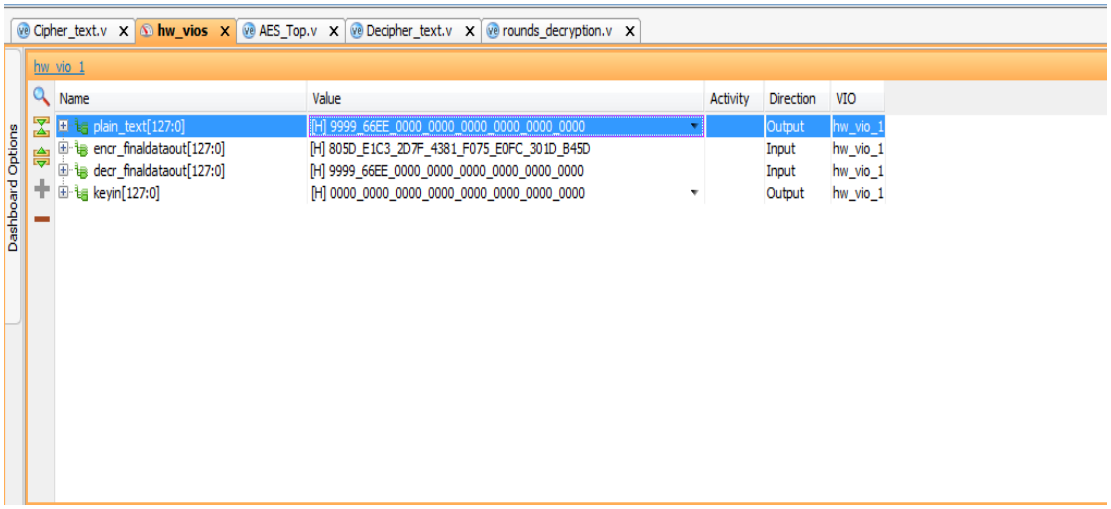
Simulation Results:



Output:

.





Conclusion:

Optimized and Synthesizable VERILOG code is developed for the implementation of both encryption and decryption process. Each program is tested with some of the sample vectors and output results are perfect with minimal delay. Therefore, AES can indeed be implemented with reasonable efficiency on an FPGA, with the encryption and decryption taking an average of 320 and 340 ns respectively (for every 128 bits). The time varies from chip to chip and the calculated delay time can only be regarded as approximate. Adding data pipelines and some parallel combinational logic in the key scheduler and round calculator can further optimize this design.