California State University, Northridge

Department of Electrical & Computer Engineering



ECE 526L

Final Project Report

(AES Cryptography)

By

Avinash Damse

CSUN ID- 203131064

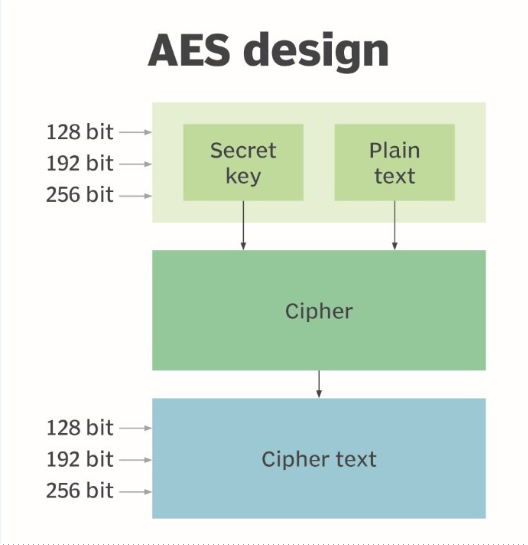
**1: Introduction**

The Advanced Encryption Standard (AES) is a symmetric [block cipher](https://www.techtarget.com/searchsecurity/definition/block-cipher) chosen by the U.S. government to protect classified information.

AES is implemented in software and hardware throughout the world to [encrypt](https://www.techtarget.com/searchsecurity/definition/encryption) sensitive data. It is essential for government computer security, cybersecurity and electronic data protection.

The objective of this project is to understand the working of AES (Advanced Encryption Standard) and implement AES using verilog and showing successful demonstration.

Block Diagram For AES Algorithm.



### Working of AES

### 1. Dividing data into blocks

First, we have to keep in mind that AES is a **block cipher**. Unlike stream ciphers, it encrypts data in **blocks of bits** instead of bit-by-bit.

Each of its blocks contains a column of 16 bytes in a layout of four-by-four. As one byte contains 8 bits, we get 128-bit block size (16x8=128).

Thus, the very first step of AES encryption is dividing the plaintext into blocks(4X4 Matrix).

### 2. Key expansion

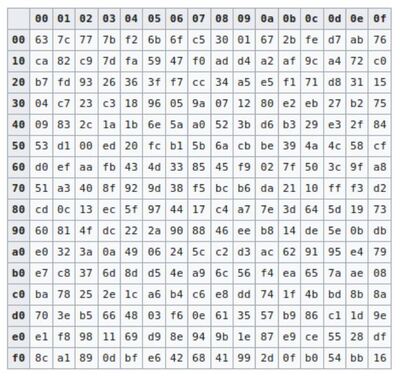
This is an important step of AES encryption. It produces new 128-bit round key.

### 3. Adding round key

This is the very first round of AES encryption.

### 4. Byte substitution

Now, the AES algorithm substitutes every byte with a code according to a pre-established table called the  **S-box**. It looks like this:

.

### 5. Shifting rows

In this step, the AES algorithm shifts the rows of the block it got during the byte substitution process.

### 6. Mixing columns

Talking in mathematical terms, this step multiplies each column by a predefined matrix, giving us a brand new block of code.

### 7. Adding round key

In this process we apply the round key we got in the key expansion section

### 8. Rinse and repeat

**2: Implementation**

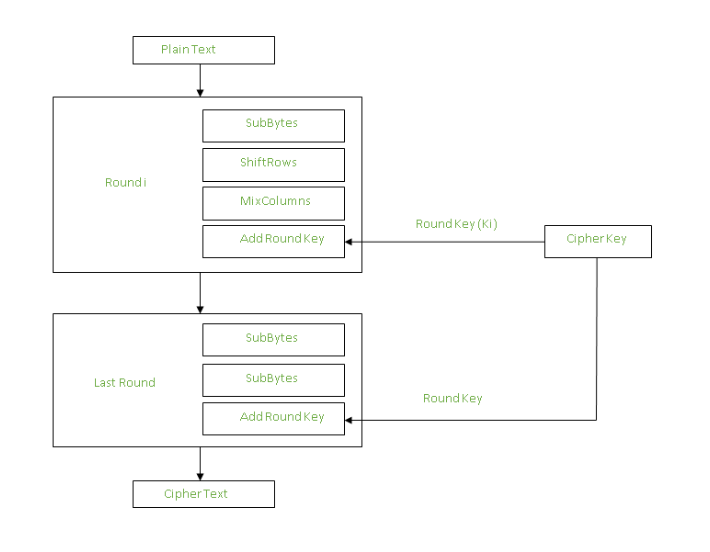
**a. Part 1: Creating AES\_Core Module**

In this module I have instantiated the different modules which are required for the AES implementation.

The modules are.

1. keyGenerationandExpansion
2. subBytes
3. shiftRows
4. mixColumn
5. sbox
6. 9rounds
7. Lastround

These modules are separately created to perform specific function according to following flowchart.



**b. Part 2: Creating Testbench**

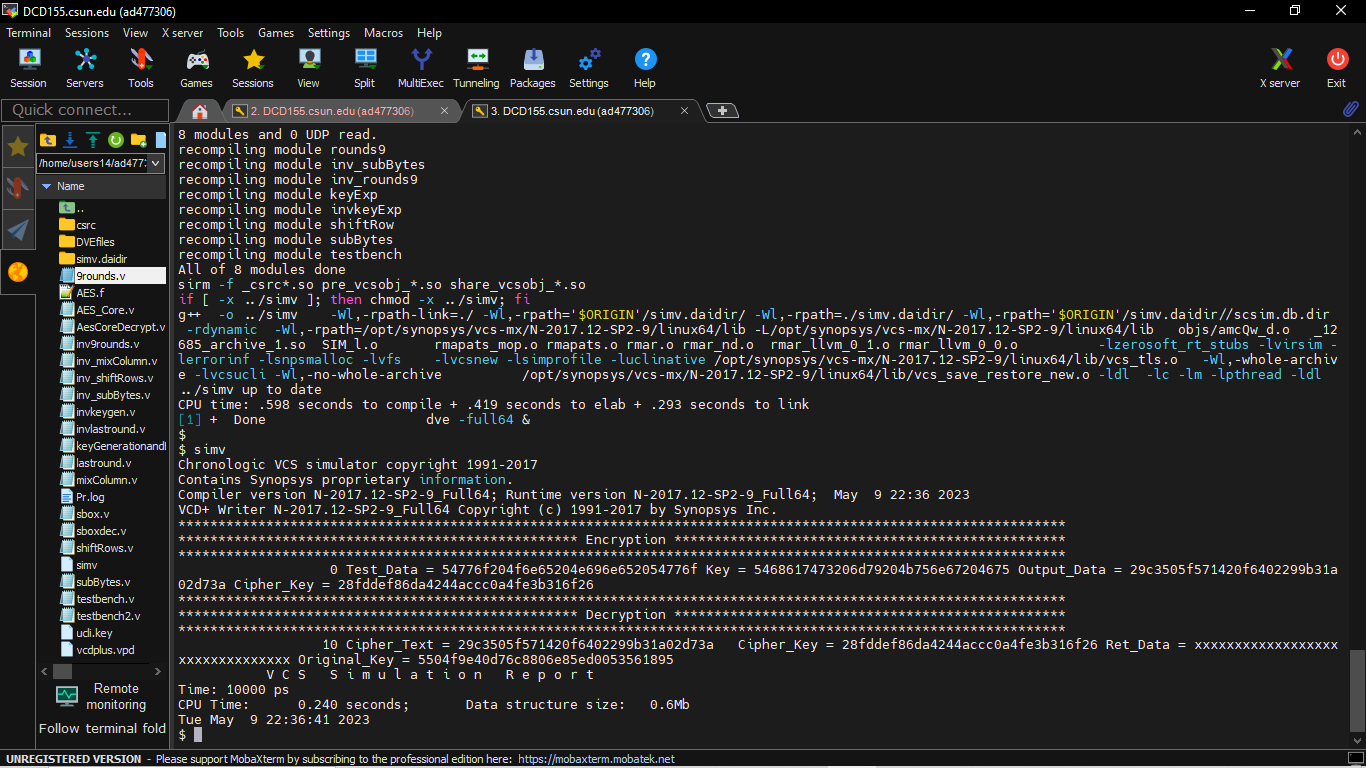
I have written the test bench for the **AES** module. We require test bench just to make sure that the module we have created is working properly. Here, in this testbench I have used some inputs Example with the secret key.

**c. Part 3: execution.**

Using “vcs -debug -full64 …..testbech.v” command I executed testbench file.

**d. Part 4: Simulation**

After an execution of all modules, I have run the command “simv” for simulation.

****

**e. Part 5: Creating Log File**

After running the simulation I created the log file for testing using the “simv -l Proj.log” command

Command: /home/users14/ad477306/Verilog/Project/./simv -l Pr.log

Chronologic VCS simulator copyright 1991-2017

Contains Synopsys proprietary information.

Compiler version N-2017.12-SP2-9\_Full64; Runtime version N-2017.12-SP2-9\_Full64; May 9 22:37 2023

VCD+ Writer N-2017.12-SP2-9\_Full64 Copyright (c) 1991-2017 by Synopsys Inc.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Encryption \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

0 Test\_Data = 54776f204f6e65204e696e652054776f Key = 5468617473206d79204b756e67204675 Output\_Data = 29c3505f571420f6402299b31a02d73a Cipher\_Key = 28fddef86da4244accc0a4fe3b316f26

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Decryption \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

10 Cipher\_Text = 29c3505f571420f6402299b31a02d73a Cipher\_Key = 28fddef86da4244accc0a4fe3b316f26 Ret\_Data = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx Original\_Key = 5504f9e40d76c8806e85ed0053561895

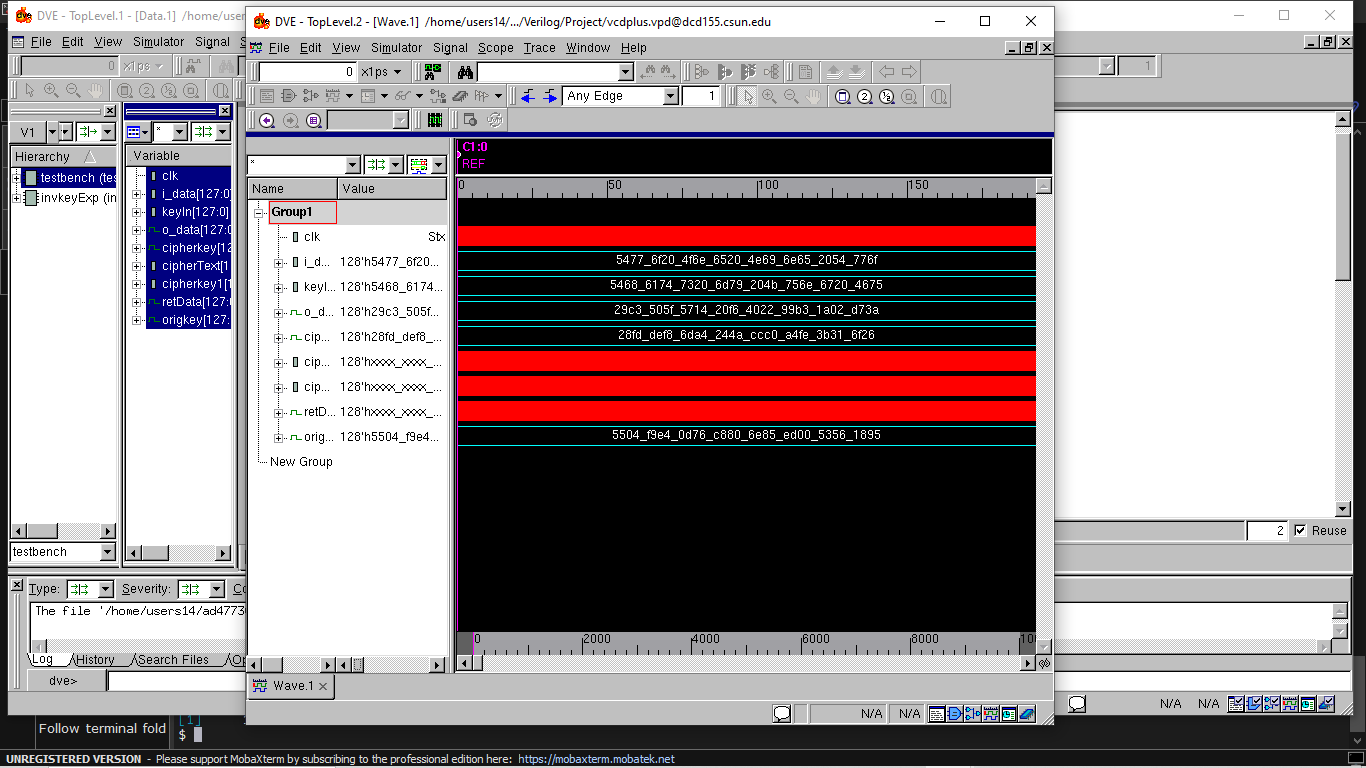
V C S S i m u l a t i o n R e p o r t

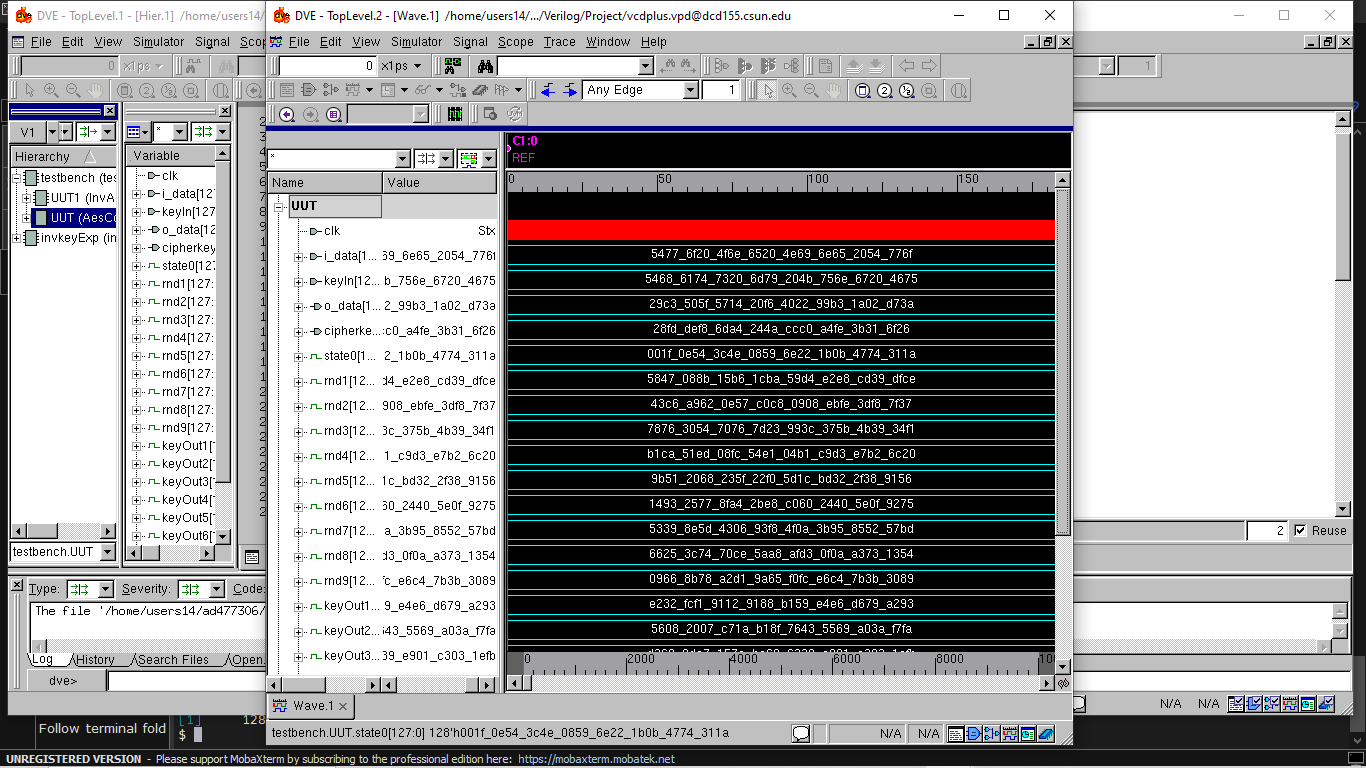
Time: 10000 ps

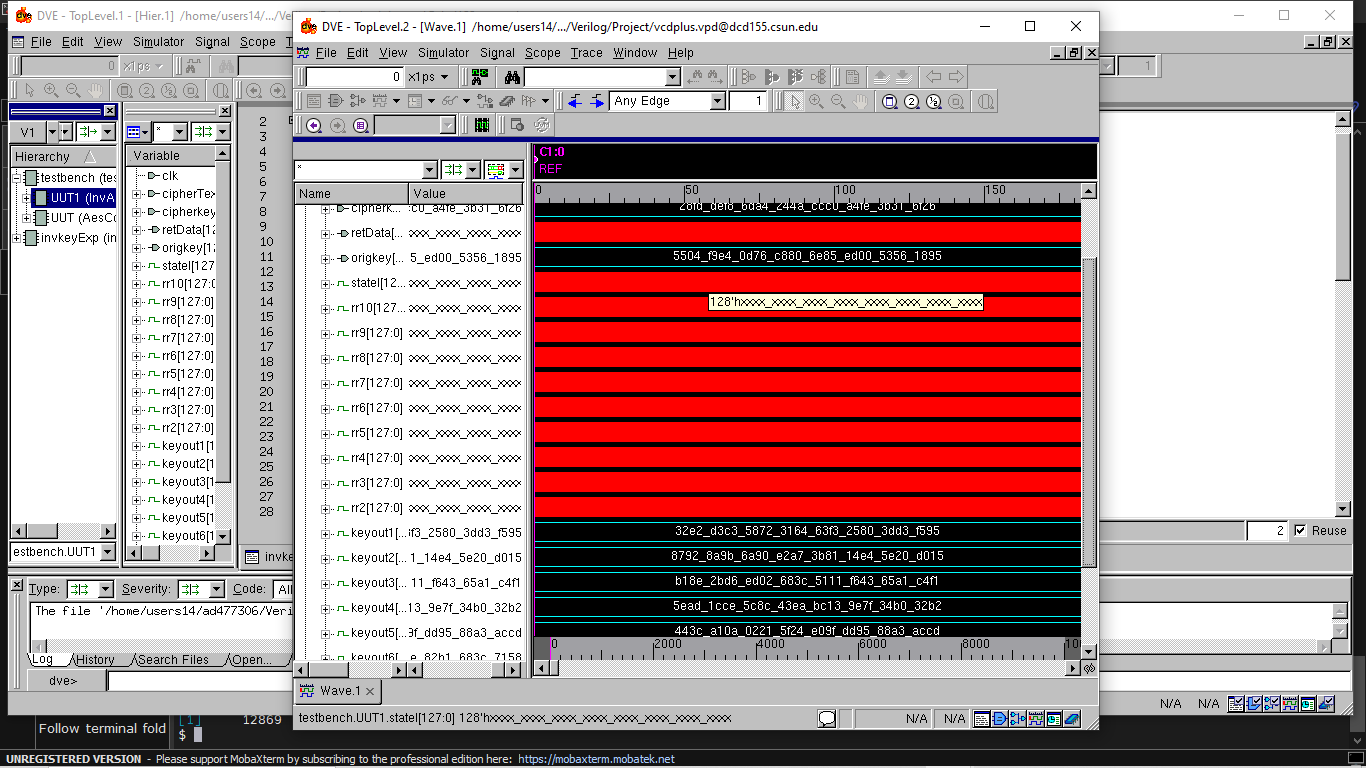
CPU Time: 0.230 seconds; Data structure size: 0.6Mb Tue May 9 22:37:02 202

**f. Part 6: Seeing the waveform.**

After creating the log file I opened the DVE using “dve -full64 &” command to see the waveforms.

****

****

****

**TESTING :**

**Message :**

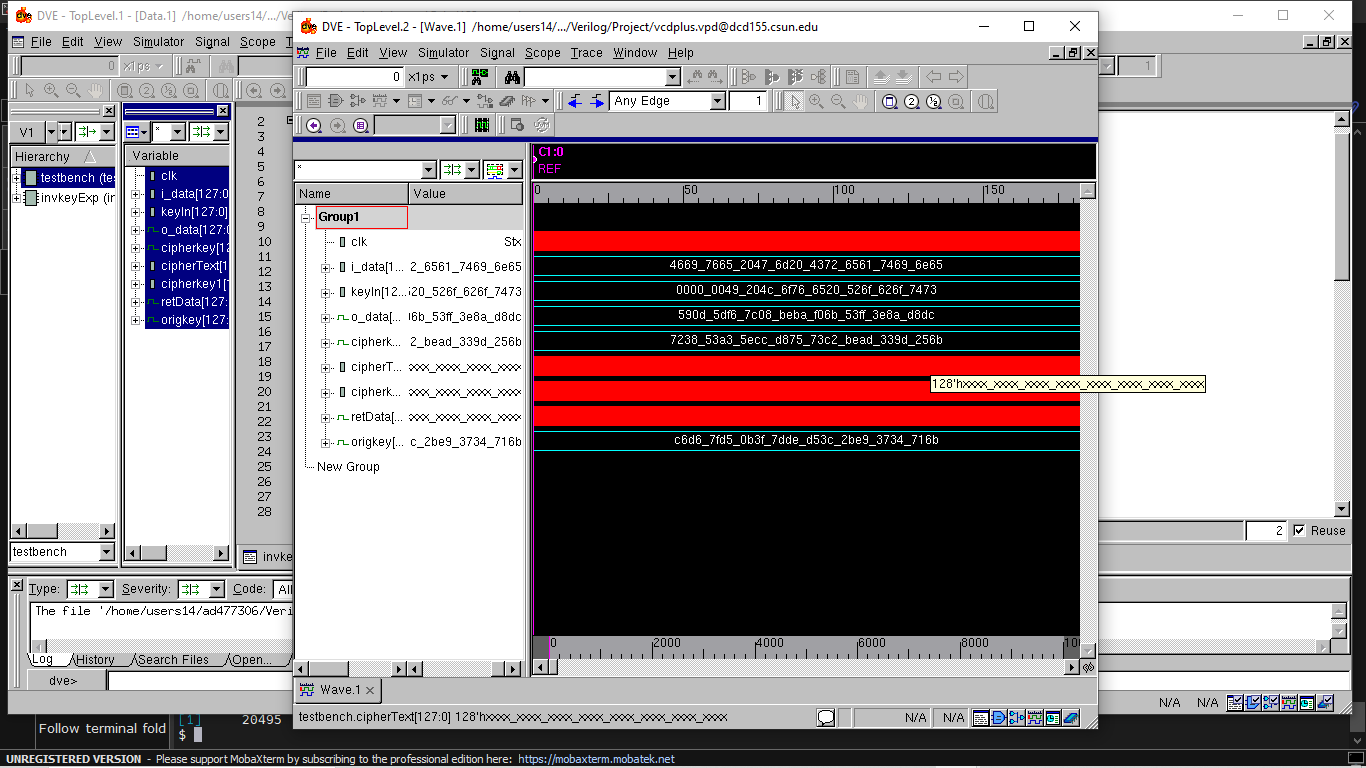
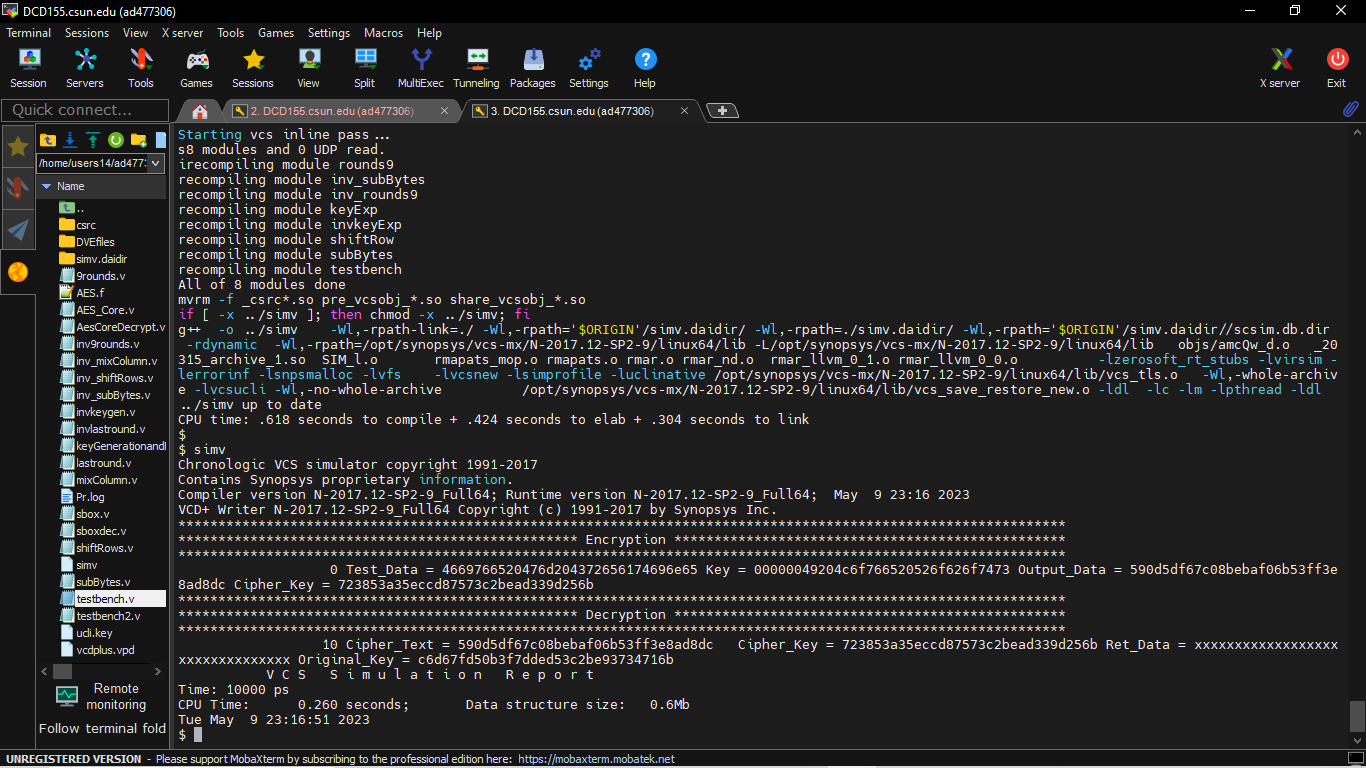
**Text:** Five G Creatine

Hex : 4669766520476d204372656174696e65

**Key :**

**Text :** I Love Robots

Hex : 49204c6f766520526f626f7473

****

**Conclusion:**

The constructed AES module in this experiment was verified of its functionalities. The Encryption of message id done successfully, although the decryption of the cipher text back to the original message had some error. But finally I concluded that the encryption algorithm works correctly.(Using solved example form internet).

I hereby attest that this lab report is entirely my own work. I have not copied either code or text from anyone, nor have I allowed or will I allow anyone to copy my work.

Name (printed) Avinash Damse

Name(signed) Date : 14-May-2023

