# FIRST REVIEW REPORT

# VLSI IMPLEMENTATION IN HARDWARE SECURITY MODULE BASED ON AES ENCRYPTION METHOD

**20ECPJ701 - PROJECT PHASE 1**

***Submitted by***


**AKASH A - 412520106007**

**JAIGANESH P - 412520106053**

**MAHIZHAN M - 412520106085**


***In Partial fulfillment for the award of the degree***

***of***

***BACHELOR OF ENGINEERING***

***in***

**ELECTRONICS AND COMMUNICATION ENGINEERING**


**SRI SAIRAM ENGINEERING COLLEGE (AUTONOMOUS),**

**SAI LEO NAGAR, CHENNAI-44**


**ANNA UNIVERSITY: CHENNAI 600 025**

**SEPTEMBER - 2023**

## ABSTRACT

Cryptography is very important now-a-days for data security and integrity as the ecommerce and internet applications has increased. But, it has least importance in many cases because of extra memory and other requirements needed for the implementation. The main aim of this work is to implement Advanced Encryption Standard (AES) Encryption using Verilog. To protect data like electronics, cryptographic algorithms are used. Each round of encryption associated with delay can be reduced by AES parallel design. This work proposes a low power and high throughput implementation of AES algorithm using key expansion approach. This minimizes the power consumption and critical path delay using the proposed high-performance architecture. The fundamental goal of the initiative is to increase data flow, although security considerations have become increasingly important over time. The use of encryption and decryption techniques inside VLSI has recently increased since cryptography can convert plaintext to cipher and vice versa. The most recent developments in cryptography technology will be applied in the hardware security module. by simultaneously writing a lot of HDL modules. The main objective is to send and receive data securely without allowing data to be hacked, as well as to improve the performance of a specific parameter. It is interesting to note that any encryption algorithm works in a digital environment and all the blocks in the system will handle digital data in security.

## JUSTIFICATION FOR SDG & SAP

**SDG No: 9**

: Industry, Innovation and Infrastructure


**SAP No: SAP090C**

9 Support domestic technology development, research and innovation in developing countries, including by ensuring a conducive policy environment for, inter alia, industrial diversification and value addition to commodities.


## 1)DESIGN:

Verilog coding is the method used in this system. We would examine the most recent version of AES encryption first, modify it to reach the algorithm's optimum efficiency, and then implement it in a Hardware Security Module. After that We would implement into a Controller for further study of the Security of the Controller System. This is our proposed methodology of our Project.

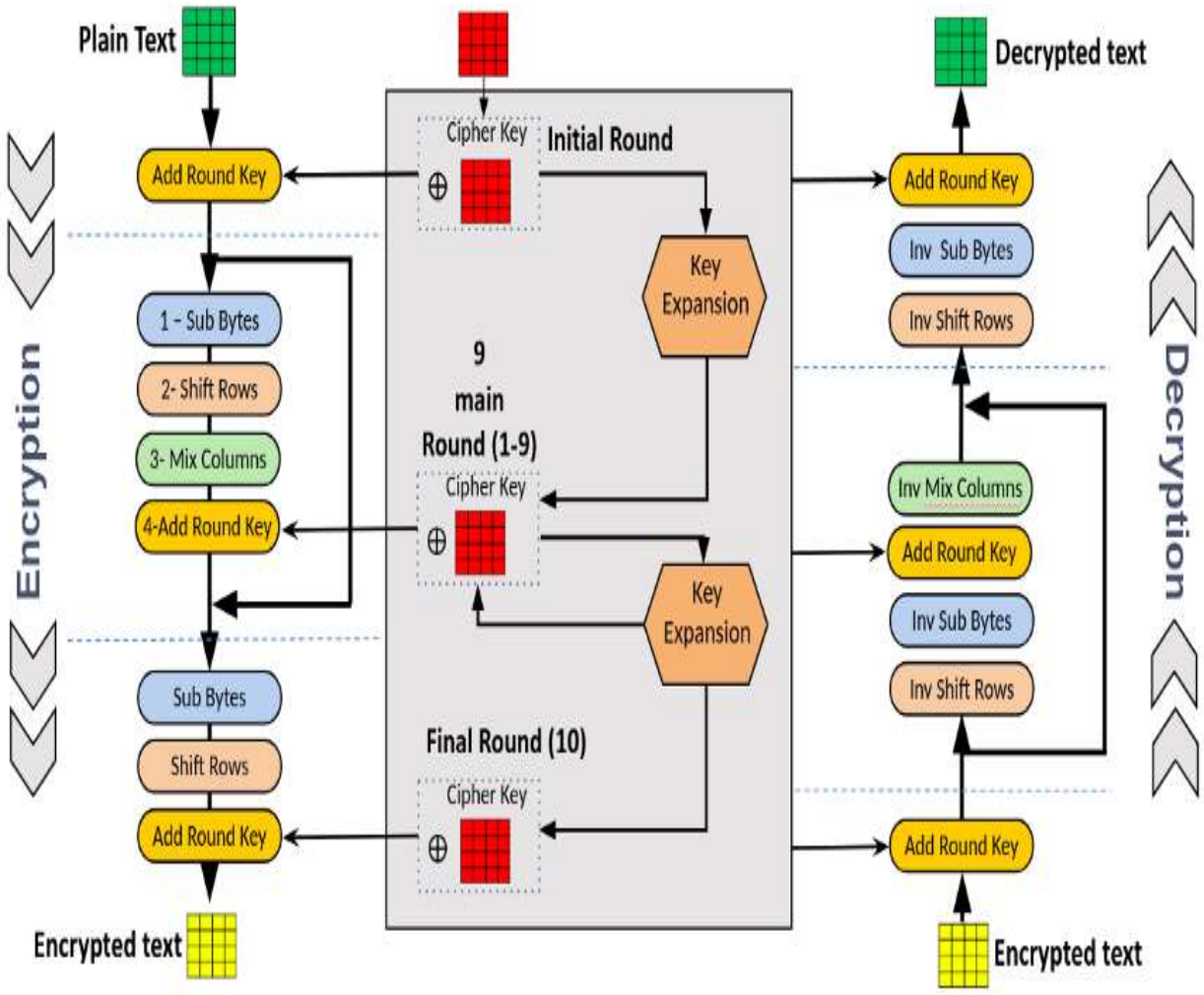



**Figure 1.1 BLOCK DIAGRAM OF AES**

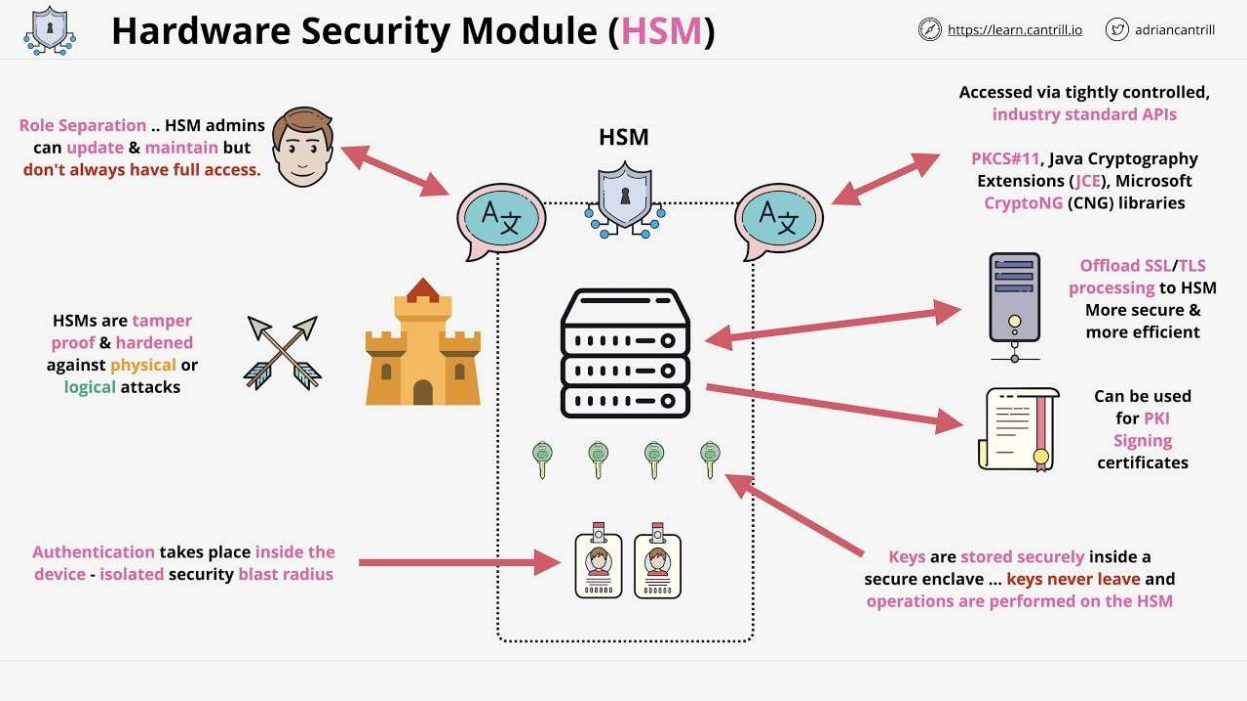



**Figure 1.2 BLOCK DIAGRAM OF HARDWARE SECURITY MODULE**

## 2)MODULAR REQUIREMENT:

### 2.1) Hardware Requirement

#### 2.1.1) Hardware security module

- A hardware security module (HSM) is a physical computing device that safeguards and manages digital keys, performs encryption and decryption functions for digital signatures, strong authentication and other cryptographic functions.
- These modules traditionally come in the form of a plug-in card or an external device that attaches directly to a computer or network server.
- A hardware security module contains one or more secure crypto processor chips

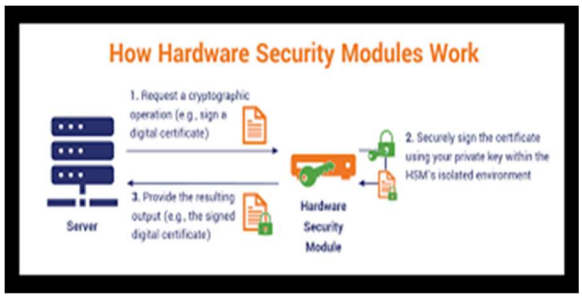



Figure 2.1.1 HSM


Figure 2.1.2 HSM

#### 2.1.2) FPGA

- Field Programmable Gate Arrays (FPGAs) are semiconductor devices that are based around a matrix of configurable logic blocks (CLBs) connected via programmable interconnects.
- FPGAs can be reprogrammed to desired application or functionality requirements after manufacturing.
- This feature distinguishes FPGAs from Application Specific Integrated

Circuits (ASICs), which are custom manufactured for specific design tasks.

- Although one-time programmable (OTP) FPGAs are available, the dominant types are SRAM based which can be reprogrammed as the design evolves

Figure 2.1.3

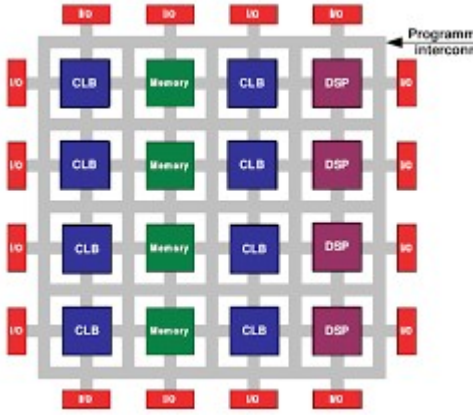



Figure 2.1.4

## 2.2) Software Requirement

### Xilinx ISE Software

**Xilinx ISE** (**I**ntegrated **S**ynthesis **E**nvironment) is a discontinued software tool from Xilinx for synthesis and analysis of HDL designs, which primarily targets development of embedded firmware for Xilinx FPGA and CPLD integrated circuit (IC) product families. It was succeeded by Xilinx Vivado. Use of the last released edition from October 2013 continues for in-system programming of legacy hardware designs containing older FPGAs and CPLDs otherwise orphaned by the replacement design tool, Vivado Design Suite.

ISE enables the developer to synthesize ("compile") their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. Other components shipped with the Xilinx ISE include the Embedded Development Kit (EDK), a Software Development Kit (SDK) and Chip Scope

Pro. The Xilinx ISE is primarily used for circuit synthesis and design, while ISIM or the Model Sim logic simulator is used for system-level testing.

## Simulation:

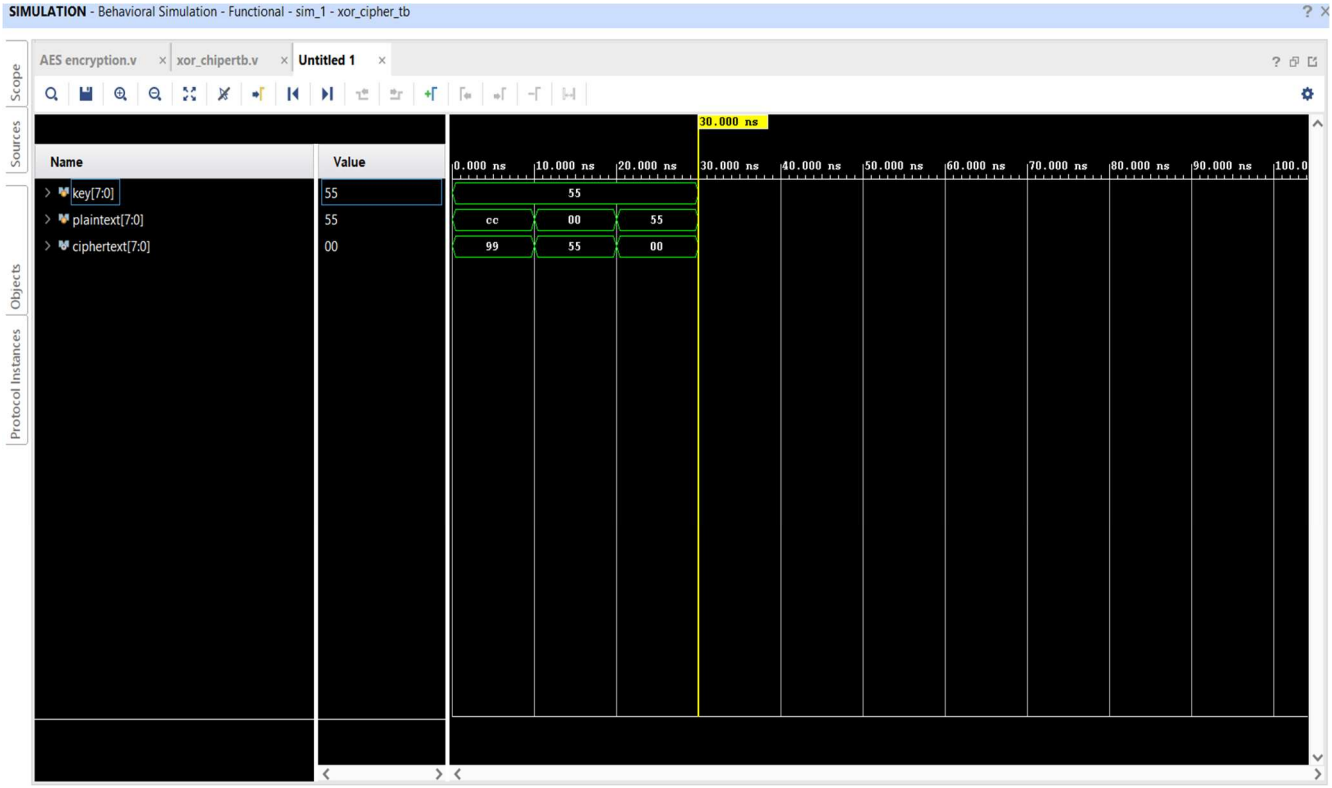
System-level testing may be performed with ISIM or the Model Sim logic simulator, and such test programs must also be written in HDL languages. Test bench programs may include simulated input signal waveforms, or monitors which observe and verify the outputs of the device under test.

Model Sim or ISIM may be used to perform the following types of simulations:

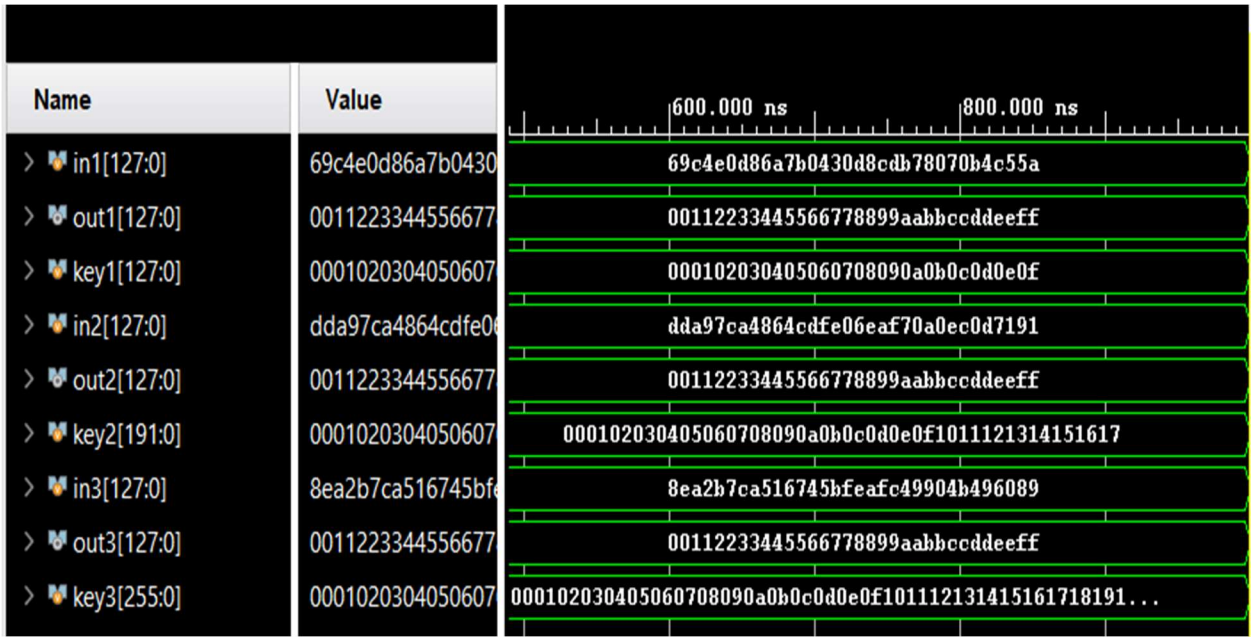
- Logical verification, to ensure the module produces expected results
- Behavioral verification, to verify logical and timing issues
- Post-place & route simulation, to verify behavior after placement of the module within the reconfigurable logic of the FPGA


Figure 2.2.1

Experimental Results:




This is the output of a programme using the basic AES encryption Algorithm with a 128-bit key.




This is the output of a programme using the  AES Encryption and Decryption Algorithm with a 128-bit key , 192-bit key and 256-bit key.