Name: Harshvardhan Soni
Enrollment No: 92200133028
Department of Information and Communication Technology
Subject: Capstone Project – 01CT0715

Project Title: Implementation of RSA Cryptographic Algorithm on Nexys-04

Component: Testing and Validation

Course: ICT Engineering Capstone

Academic Year: 2025-26

Department: Information and Communication Technology

❖ Testing and Validation:

The testing and validation component carefully verifies the functional correctness and performance efficiency of the hardware-accelerated RSA core and its parallel systems on the Nexys-4 DDR FPGA.

❖ Testing Methodology:

Test Type	Tool/Framework	Purpose
Functional Verification (Unit & Integration)	Vivado Simulator (using Verilog Testbenches)	To ensure that all modules (especially rsa_core.v) produce the mathematically correct output for known inputs and that data flows correctly through the system.
Performance Evaluation	Vivado Timing and Implementation Reports (Post-Place & Route)	To measure critical hardware metrics: latency (clock cycles) and resource utilization (LUTs, FFs).
Hardware Validation	Xilinx Integrated Logic Analyzer (ILA) and On-Board Peripherals	To confirm the synthesized logic operates correctly on the physical FPGA board, validating I/O constraints and real-world timing.

❖ Unit Tests:

Test Case	Module Tested	Input (M,E,N,C)
UT-1: Hex-to-7Seg	hex_to_7seg.v	hex=4’hC
UT-2: ROM Data	wifi_key_rom.v	N/A (Read M_CONST)
UT-3: RSA Setup	rsa_core.v (IDLE → RUN)	m_in=3,e_in=5,n_in=35
UT-4: RSA Decrypt	rsa_core.v	C=27,d=13,n=35 (Known RSA tuple)
UT-5: Modular Reduction	rsa_core.v (Internal Logic)	(result×base)=120,n=35

❖ Integration Tests:

Test Case	Components Integrated	Input	Expected Output/Sequence	Result
IT-1: Encrypt → Decrypt Pipeline	rsa_enc→rsa_dec	M=17’d123 (from m_last4_ext)	rsa_enc.done triggers rsa_dec.start, rsa_dec.out=M	Passed
IT-2: Display Gating	top.v (switches) →display_driver	sw0=1, sw1=0	Display shows M on digits 0-3 and C on digits 4-5. Mdec (digits 6-7) is 00.	Passed
IT-3: FSM Synchronization	rsa_core (Control Signals)	start=1 (high) then starts=0 (low)	Core enters RUN, completes, moves to FIN, and returns to IDLE only when start is low.	Passed

❖ Validation Against Objectives:

Demonstrate Modularity and Reusability:

Validation: The two instances of the rsa_core module, Encrypt and Decrypt, work correctly with different exponents (e=17 and d=47345) but share the same core logic and modulus (n=67591).

The rsa_core module is made reusable through its parameterized WIDTH.

Alignment: This shows how a modular design improves reusability and makes the system easier to expand, which follows good practices in hardware development.

Implement a Functional RSA Encryption/Decryption System.

Validation: This was confirmed by Unit Test UT-4 (Decryption) and Integration Test IT-1 (Pipeline), showing the system correctly produces the original plaintext (M) after encryption and decryption.

The design fully uses the Square-and-Multiply algorithm.

Alignment: This meets the main goal of showing a complete RSA process in a custom setup.