INITIAL PLANS FOR ESTIMATING THE HARDWARE PERFORMANCE OF AES SUBMISSIONS

The National Security Agency (NSA) will provide technical support to the National Institute of Standards and Technology (NIST) in the form of an analysis of the hardware performance of the Advanced Encryption Standard (AES) algorithm submissions which are under consideration in Round 2 of the review process. This analysis will consist of the design, coding, simulation and testing of the submitted algorithms using the procedure outlined below. Throughout this evaluation, NSA will take pains to assure that best design practices are used and that all algorithms receive equal treatment. No attempt will be made to optimize any particular design, but care will be taken to find the best configuration for each algorithm. Cross-validation measures will be used to try to overcome the subjective effects of the design process and to ensure that all designs receive the same calibre of attention. The results of this analysis should provide an accurate measure of the hardware performance of each algorithm relative to the others. Undoubtedly more optimized (and hence better performing) implementations of these algorithms can be designed, so the individual score of any particular algorithm is not very valuable outside the context of this test. The point of this analysis is to provide a controlled setting in which a meaningful comparison can be made. The results of this testing, which should be available six months after the start of Round 2, will be unclassified and will be made public by NIST.

Overview

Based on a mathematical description (and possible C code model) of the submitted algorithms that make it to Round 2, NSA designers will fully describe each of the algorithm submissions in a hardware modeling language. A review by a team of design engineers will follow the initial design stage to reduce the effects of coding style on performance. Using analysis tools and simulations, NSA designers will provide performance estimates based on each of the hardware models. A summary report of the performance of all the algorithm submissions will compare and contrast the results of the analysis.

Hardware Description

Very High Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL)

VHDL modeling is analogous to programming simulations in C code and follows much of the same syntax. However, unlike a behavioral description of the algorithm, VHDL specifies how the algorithm will be implemented in hardware. Using this hardware language, NSA designers will fully describe the hardware necessary to implement each of the algorithm submissions. Performance metrics, such as speed, area, etc. (see below) can be estimated from the hardware description using available analysis tools and computer aided design (CAD) tools.

Code Generation

NSA will assign one or more engineers to the design of the VHDL (IEEE 1076) for each algorithm submitted. Initial hardware designs will be straightforward implementations of the core algorithm. Following completion of each initial design, an informal group of engineers will meet to review and to provide feedback for the design. Improvements and alternatives to the initial design will be examined to determine potential benefits from differing architecture approaches (area compression, pipelining, etc.). Variants of the design which improve the performance of the algorithm may then be programmed for comparison. The VHDL code generated during this test procedure will be unclassified and publicly available in a manner consistent with export requirements.

Simulation Modeling

NSA will follow-up the design phase with a functional VHDL simulation of the designs using Vantage Simulator (Vantage Analysis Systems, Inc.) to verify the correct operation of the algorithm. The submitted test vectors will be applied to assure that the design is working as intended. Opportunities to compare the algorithm output with the C code model supplied will provide an added assurance that the algorithm is operating as expected.

Synthesis

Gate-level synthesis of the algorithm will utilize Synopsys Inc's Design Compiler to produce a functionally equivalent schematic in hardware. A MOSIS specific technology library, which uses a multi-vendor integrated circuit manufacturer's library and allows fabrication at several foundries with a single design, will be used to generate a gate-level schematic of the design and provide more accurate area/timing estimates (as if the design were to be implemented in an integrated circuit (IC)).

Hardware Performance Evaluation

Reporting

Algorithm performance in each of the evaluation categories will be documented for each algorithm submission. Specific performance areas to be addressed include the following:

1. Area
2. Throughput
3. Transistor count
4. Input/Outputs Required
5. Key setup time
6. Algorithm setup time (tables, etc.)
7. Time to encrypt one block
8. Time to decrypt one block
9. Time to switch keys

Area

As an estimate based on an available MOSIS library, the results of the synthesis area reporting will consist of pre-layout area estimates of the algorithm. Although potentially different from a post-layout estimate, the area reports generated from Synopsys will provide a relative comparison of each of the algorithm submissions.

Throughput

A variety of architecture approaches will be considered to maximize the data throughput of the algorithm implementation. Timings for all relevant architectures will be reported.

Transistor Count

Based on the synthesized netlist (from Synopsys), an additional report describing the number of transistors required to implement the algorithm will be provided.

Input/Outputs (I/O) Required

Timing considerations and algorithm structure will dictate the I/O requirements for specific implementations of each of the algorithms. Trade-offs in reducing pin count will be examined to meet performance goals (e.g., pipelining may require a large I/O count).

Key Setup Time

Key run-up times will be examined to assess the overhead of each algorithm in establishing a usable key.

Algorithm Setup Time

This area will address minimum setup times before an algorithm is ready to process data. Time to create look-up tables, etc. will fall in this category.

Time to Encrypt One Block

This area will address minimum latency times for each of the algorithm submissions. Both the time to encrypt a single block and the average time to encrypt one block when encrypting a file will be reported.

Time to Decrypt One Block

This area will address minimum latency times for each of the algorithm submissions. As above, both single block time and average time will be reported.

Time to Switch Keys

This area will address key agility and minimum times required for multiple keys.

A table summarizing the results and performance metrics will be provided for algorithm comparison. This table will be unclassified and publicly available.