Recommended Schedule of Work on the ECE 545 Project

Cryptographic Projects

Week 1: Saturday, Nov. 9 – Saturday, Nov. 16

1. Understanding of the specification and software implementation of the algorithm assigned to you.

2. Assumptions (if any)
   e.g.,
   - supported key sizes (one key size, 128 bits for majority of algorithms, is sufficient; support for other key sizes will be rewarded with bonus points)
   - limitations on the allowed sizes of the message
   - padding done in software
   - other parameters of the algorithm (if not specified clearly in the project assignment or the algorithm specification)

3. Understanding Top-Level Interface (to be provided to you by the instructor)

4. Block diagram of the Datapath

5. Interface with the division into the Datapath and Controller

6. Formulas for
   a. Execution time as a function of the message size, m
   b. Latency
   c. Throughput
   as a function of the clock period, T, and the message size, m.

Week 2: Sunday, November 17 – Saturday, November 23

1. RTL VHDL code of the Datapath

2. Generation of detailed test vectors for the Datapath based on the software implementation

3. Testbench for the Datapath

4. Debugging VHDL code of the Datapath

5. Confirming formulas for the
   a. Execution time as a function of the message size
   b. Latency
   using functional simulation
   (adjusting formulas and/or code if needed)

Week 3: Sunday, November 24 – Saturday, November 30

1. Synthesis and Implementation of the Datapath using Xilinx Spartan 6 FPGA

2. Determining resource utilization of the Datapath after placing and routing

3. Post-synthesis and timing simulation

4. Determining
   a) minimum clock period, and
   b) critical path
   of the Datapath after placing and routing

5. Determining
   a. Minimum Latency
   b. Maximum Throughput
   c. Maximum Throughput/Area
   of the Datapath based on the minimum clock period after placing and routing.

6. ASM chart of the Control Unit.

Week 4: Sunday, December 1 – Saturday, December 7

1. RTL VHDL Code of the Control Unit

2. RTL VHDL code of the Top-level Unit

3. Generation of detailed test vectors for the Top-level Unit based on the software implementation

4. Debugging Top-level Unit

5. Confirming formulas for the
   a. Execution time as a function of the message size
   b. Latency
   using functional simulation
   (adjusting formulas or/and code if needed)

6. Synthesis and Implementation of the entire circuit using Xilinx Spartan 6 FPGA

7. Determining resource utilization of the entire circuit after placing and routing

8. Post-synthesis and timing simulation

9. Determining
    a. minimum clock period, and
    b. critical path

    of the entire circuit after placing and routing

10. Determining
    a. Minimum Latency
    b. Maximum Throughput
    c. Maximum Throughput/Area

    of the entire circuit based on the minimum clock period after placing and routing.

Week 5: Sunday, December 8 – Saturday, December 14

1. Final optimizations of the VHDL code

2. Preparing final deliverables

3. Benchmarking using ATHENa targeting 3 FPGA families:
    a. Xilinx Spartan 6
    b. Xilinx Virtex 6
    c. Altera Stratix IV.