Hardware Performance Simulations of Round 2 Advanced Encryption Standard Algorithms

National Security Agency

Bryan Weeks
Mark Bean
Tom Rozylowicz
Chris Ficke

Introduction

- GOAL: Provide an unbiased comparison of the Round 2 AES candidates in hardware
- Supply hardware performance to NIST for evaluation
- Group has provided similar services within NSA.

Topics of Discussion

- Tasking
  - Deliverables
  - Target Applications
- Approach and Methods
  - Design Guidelines
  - Design Flow
  - Synthesis Analysis
- Results and Comparison
- Next Steps

Deliverables

- Hardware models for all 5 finalists
  - 128 bit key size
  - 192 bit key size (added)
  - 256 bit key size (added)
  - 3-in-1(capable of all 3 key sizes)
- Conference report
- Final report
- Design Notebooks
  - Timing/area performance curves

Target Applications (1)

- Iterated, Medium speed
  - One generalized algorithm step is implemented in hardware
  - Hardware is accessed repeatedly
  - Only 1 block of output per x number of clocks
    - Small area
    - Lower throughput
  - Virtual Private Network (VPN) applications
    - ~100 Mbps
    - e.g., Business-to-business, web applications

Target Applications (2)

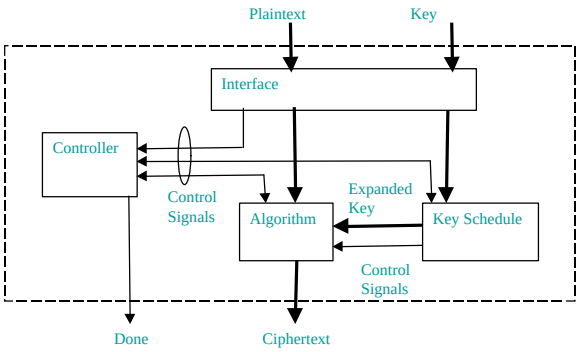
- Pipelined, High speed
  - Algorithm is “unraveled” such that each step of the algorithm is cascaded with the previous stage
  - Multiple blocks can be started successively without overrun
  - 1 block of output every clock
    - Large area
    - Higher throughput
  - Future high speed encryptors (VPNs) at 1-10 Gbps
    - e.g., uncompressed HDTV (1.5 Gbps), video delivery

Design Guidelines

- Assumptions
  - Simplified user interface
  - ROM implemented as combinational logic
  - Round by Round pipelining
- Technology
  - Used 0.5um CMOS library
  - Get 2-3x improvement with each generation (0.35um, 0.18um)
- Intention is impartiality of method

Design Guidelines

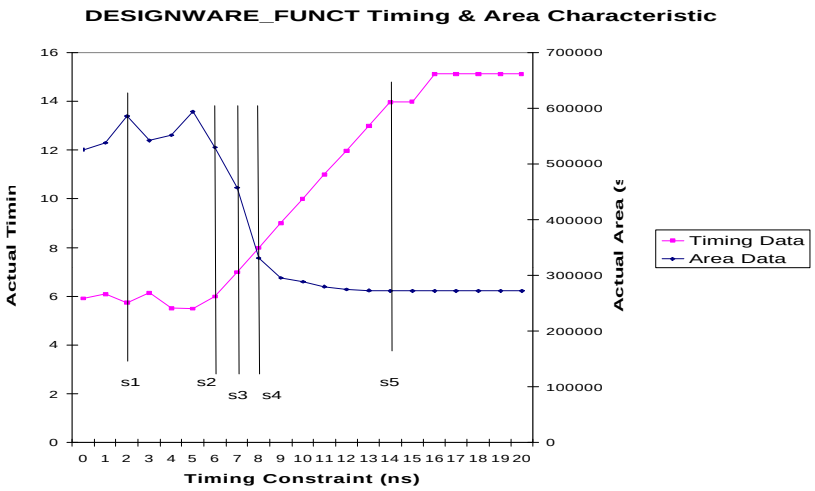
- Common Architecture
- Both encrypt and decrypt implemented



Design Flow Steps

- VHDL modeling
- Code Review
- Simulation and verification
  - Variable key and variable text testing
- Synthesis (mechanical approach through scripting)
  - Synopsys version 1999.10
- Documentation and data collection

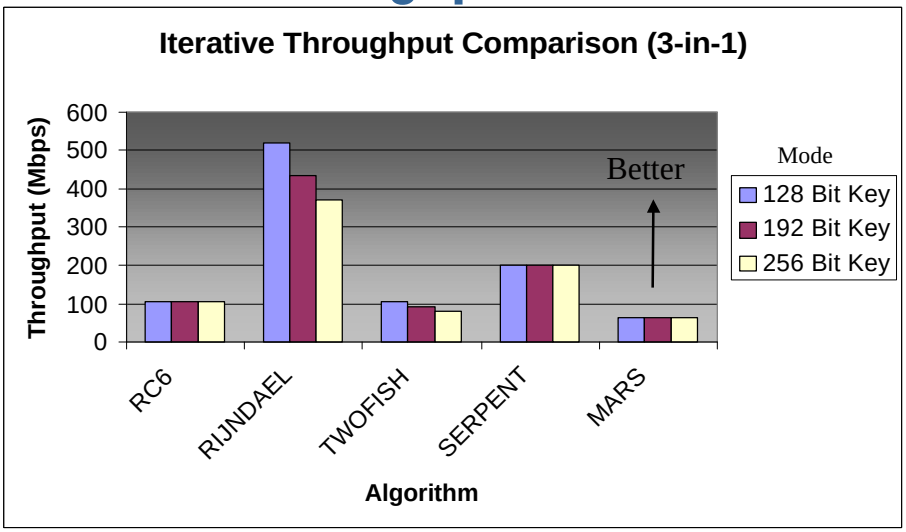
Synthesis Analysis - Example

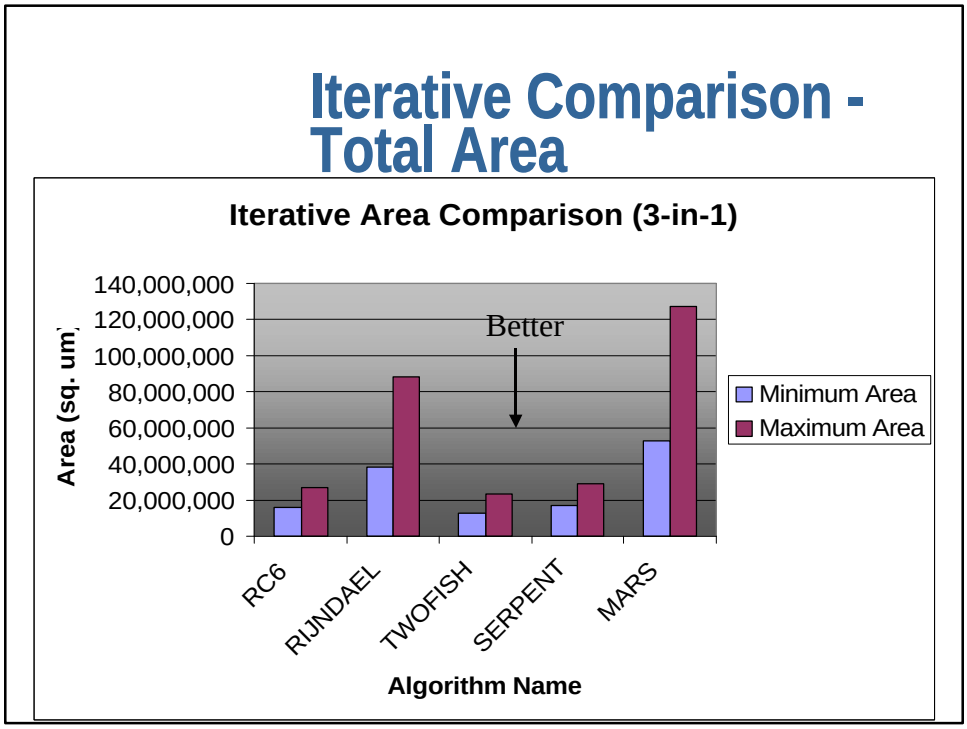


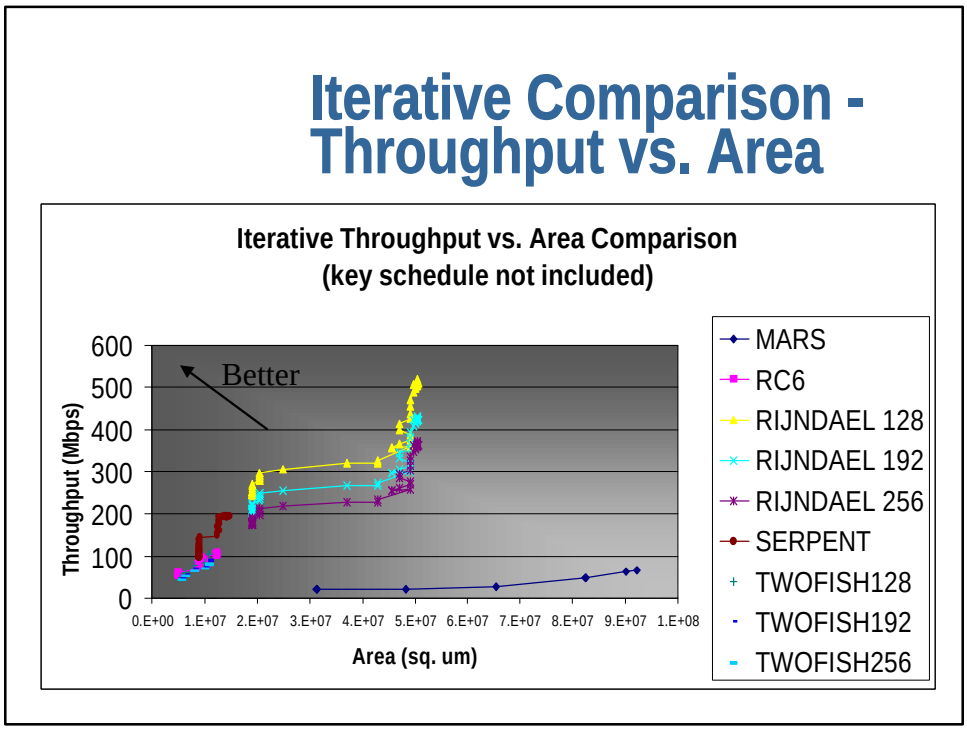
Documentation

- Automated data collection
- Area and timing data assembled into design notebook for each algorithm (Excel)
- Graphs and actual data provided

Iterative Comparison - Throughput




Iterative Comparison - Total Area
Iterative Area Comparison (3-in-1)
Area (sq. um)
140,000,000
120,000,000
100,000,000
80,000,000
60,000,000
40,000,000
20,000,000
0
Better
Minimum Area
Maximum Area
RC6
RIJNDAEL
TWOFISH
SERPENT
MARS
Algorithm Name

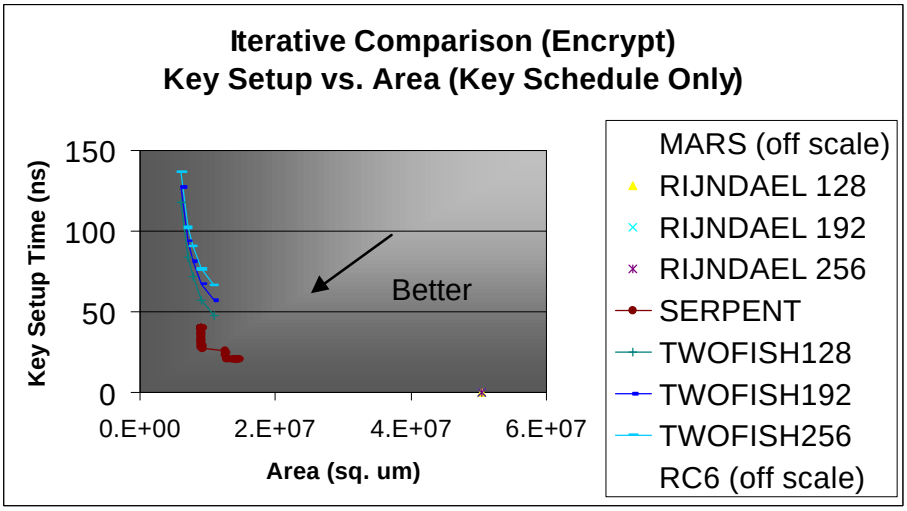

Iterative Comparison - Throughput vs. Area
Iterative Throughput vs. Area Comparison
(key schedule not included)
Throughput (Mbps)
600
500
400
300
200
100
0
Better
0.E+00 1.E+07 2.E+07 3.E+07 4.E+07 5.E+07 6.E+07 7.E+07 8.E+07 9.E+07 1.E+08
Area (sq. um)
MARS
RC6
RIJNDAEL 128
RIJNDAEL 192
RIJNDAEL 256
SERPENT
TWOFISH128
TWOFISH192
TWOFISH256

Iterative Comparison - Key Setup

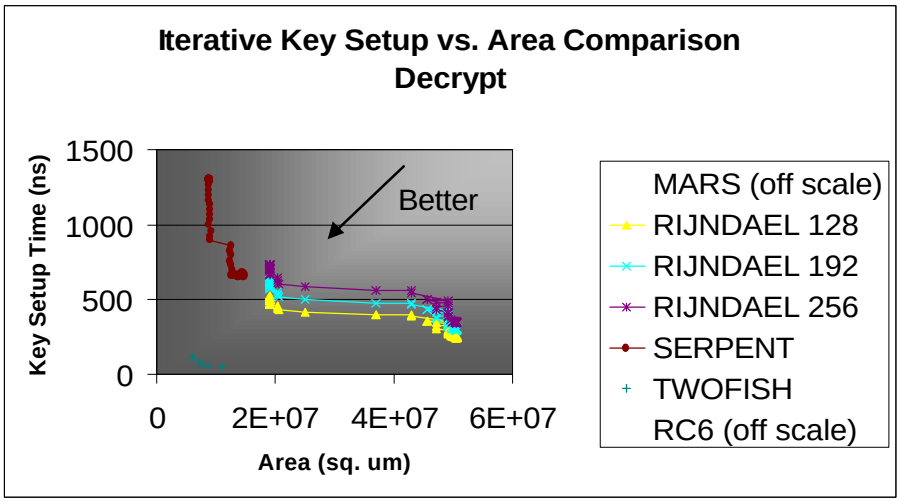
	Encrypt			Decrypt		
	128	192	256	128	192	256
MARS	9553	9553	9553	9553	9553	9553
RC6	7920	7920	7920	7920	7920	7920
RIJNDAEL	0	0	0	246.4	295.68	344.96
SERPENT	19.77	19.77	19.77	672.18	672.18	672.18
TWOFISH	42.48	61.28	79.49	42.48	61.28	79.49

Note: All times in ns

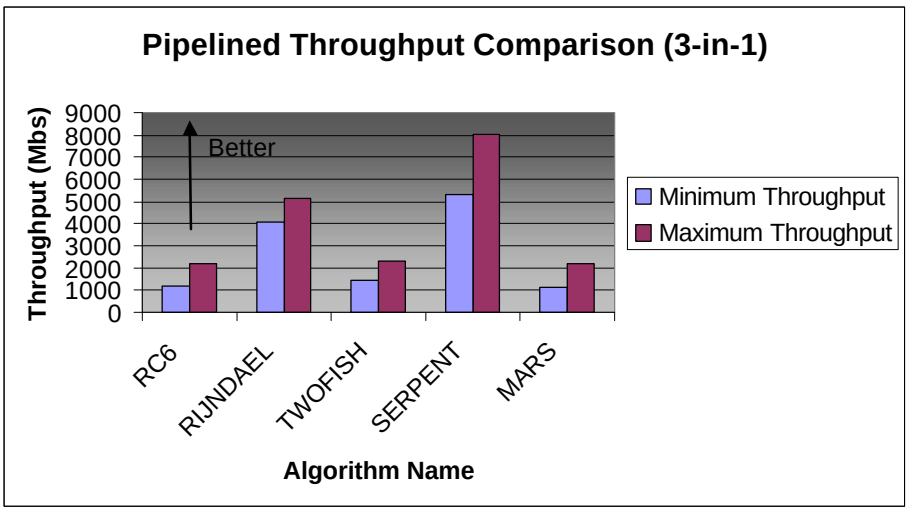
Iterative Comparison - Key Setup vs. Area (Encrypt)

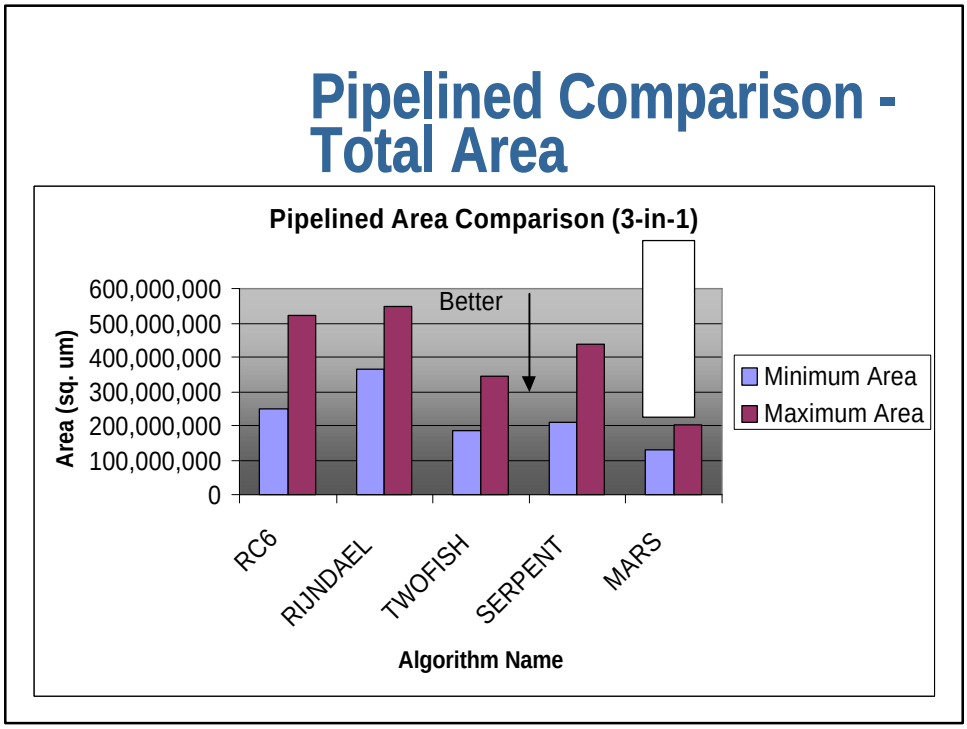


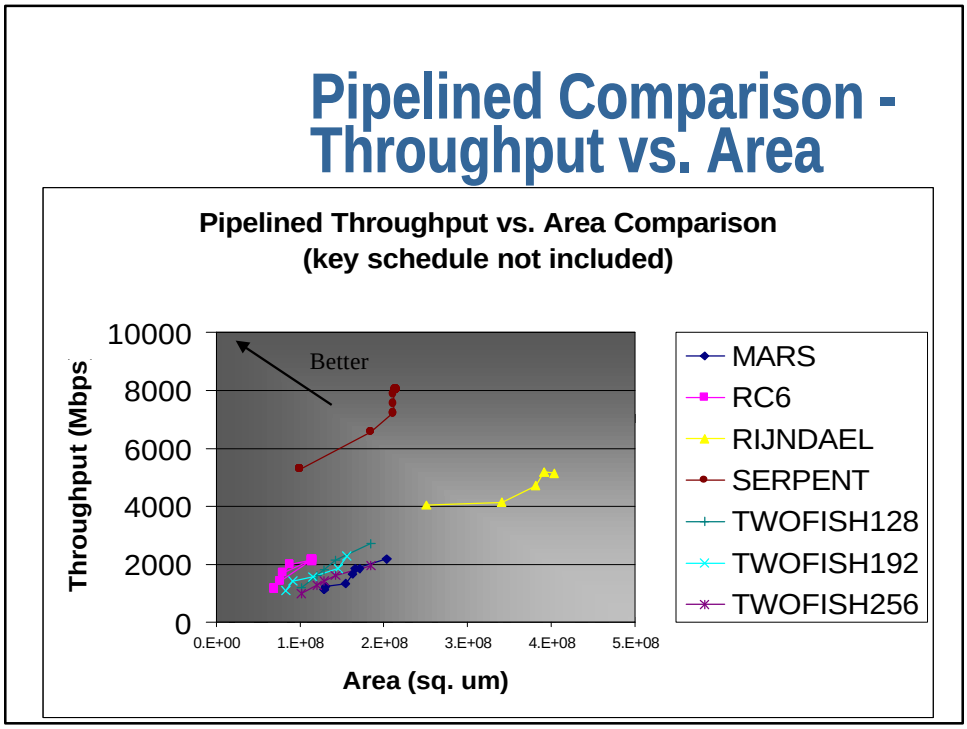
Iterative Comparison - Key Setup vs. Area (Decrypt)



Pipelined Comparison - Throughput




Pipelined Comparison - Total Area
Pipelined Area Comparison (3-in-1)
Area (sq. um)
600,000,000
500,000,000
400,000,000
300,000,000
200,000,000
100,000,000
0
Better
Minimum Area
Maximum Area
RC6
RIJNDAEL
TWOFISH
SERPENT
MARS
Algorithm Name

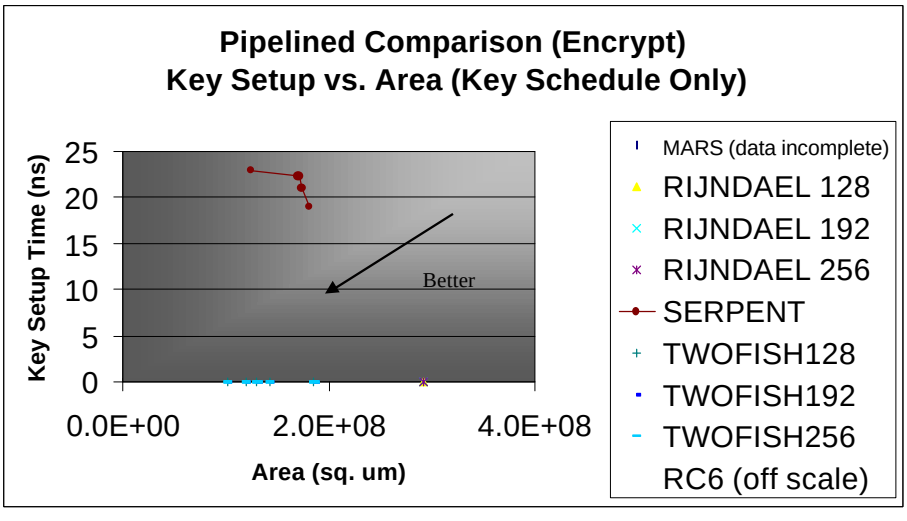

Pipelined Comparison - Throughput vs. Area
Pipelined Throughput vs. Area Comparison
(key schedule not included)
Throughput (Mbps)
10000
8000
6000
4000
2000
0
Better
0.E+00
1.E+08
2.E+08
3.E+08
4.E+08
5.E+08
Area (sq. um)
MARS
RC6
RIJNDAEL
SERPENT
TWOFISH128
TWOFISH192
TWOFISH256

Pipelined Comparison - Key Setup

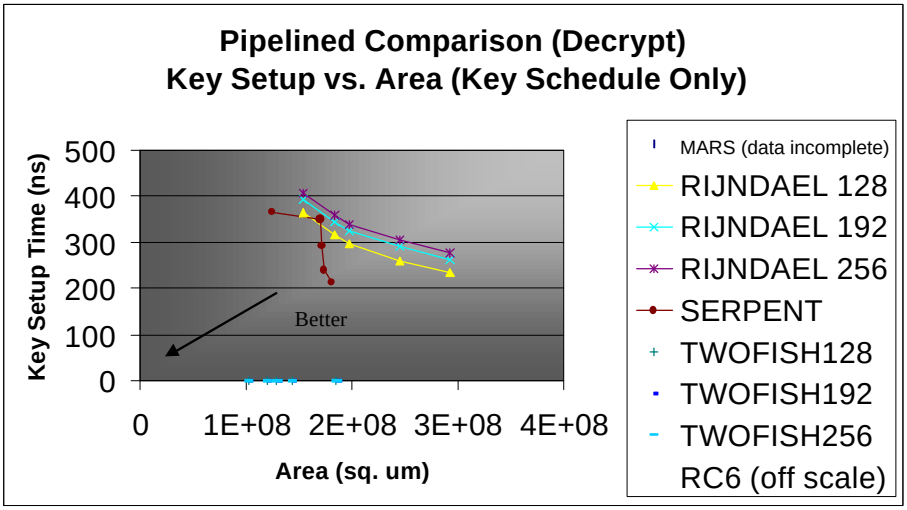
	Encrypt			Decrypt		
	128	192	256	128	192	256
MARS	Unavail.	Unavail.	Unavail.	Unavail.	Unavail.	Unavail.
RC6	3659.51	3659.51	3659.51	3659.51	3659.51	3659.51
RIJNDAEL	0	0	0	246.4	295.68	344.96
SERPENT	18.98	18.98	18.98	212.55	212.55	212.55
TWOFISH	0	0	0	0	0	0

Note: All times in ns

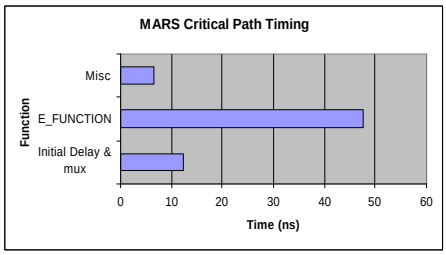
Pipelined Comparison - Key Setup vs. Area (Encrypt)

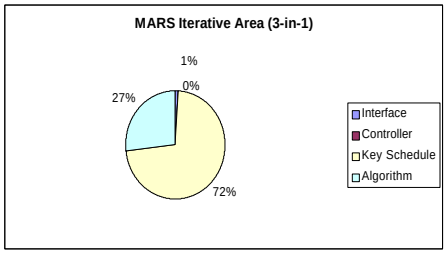


Pipelined Comparison - Key Setup vs. Area (Decrypt)



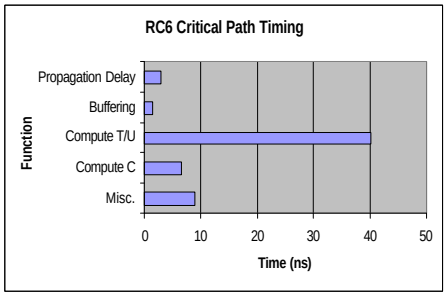
MARS

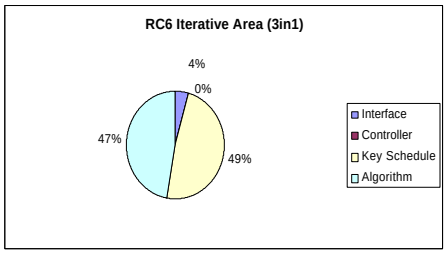




- Significant compute time in STIR portion
- Cascaded S-Boxes in key setup
- Pipelined key schedule incomplete

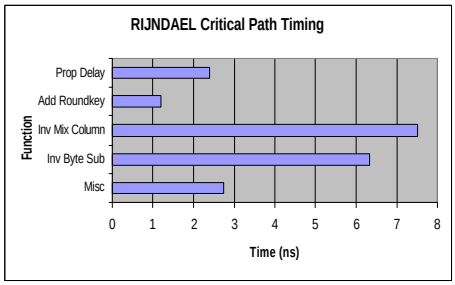
RC6

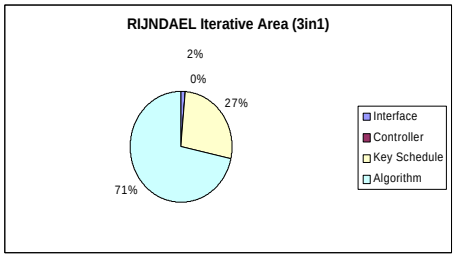




- Majority of compute time spent in multiplication
- Most of the total area is found in the key sched. portion (multiple adds)
- Iterative runup times are significant but save in area (1 copy of logic)
- Pipelined key setup and expansion large
  - (82% of area)

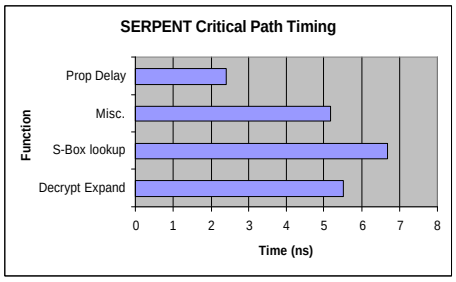
RIJNDAEL

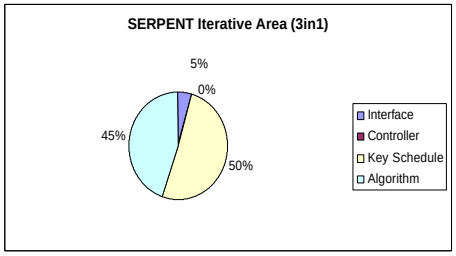




- Key scheduling is critical path block and limiting factor for processing
- Significant portion of area consumed by algorithm (S-box,mix) in iterative and pipeline
- 33% of alg. Round area is from inverse mix column

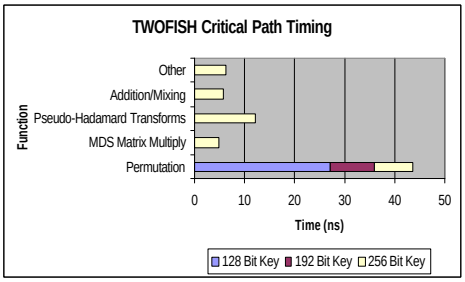
SERPENT

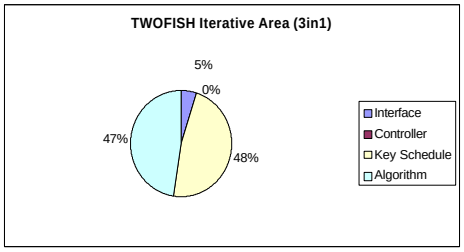




- Majority of compute time spent in S-Box
- Key schedule area reduced through expansion block re-use
- Pipelined algorithm & key schedule comparable in area

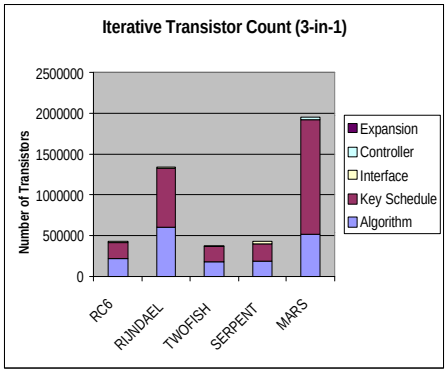
TWOFISH

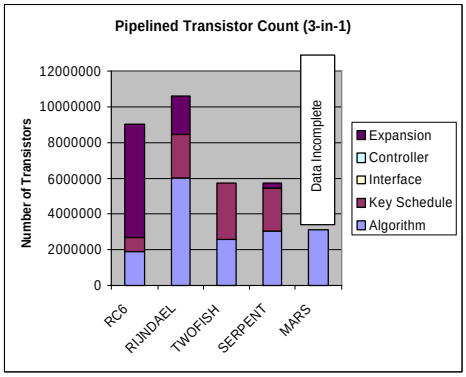




- Algorithm is limiting factor
- Significant time in S-box stacks (~60% in S-box)
- Areas comparable from function re-use
- Throughput reduced from re-use of key sched. round (20 clks iterative)
- No key setup in pipelined from parallel functions

Transistor Count





Summary

- GOAL: Provide an unbiased comparison of the algorithms
  - Same methodology applied to all
- Algorithm performance varies across parameters
- Preferences depend on how NIST and community will weight the parameters

Next Steps

- Continued work on the remaining key sizes
  - NIST originally requested 3-in-1 and 128 bit key sizes
  - Data will be provided for 192 and 256 bit keys as well
- Documentation and final report