

# CryptoManiac: Application Specific Architectures for Cryptography

**Advanced Computer Architecture Lab  
University of Michigan**

Lisa Wu, Chris Weaver, Todd Austin  
{wul,chriswea,taustin}@eecs.umich.edu



## Overview

- ❖ Goal - fast programmable cryptographic processing
  - ♦ Fast : efficient execution of computationally intensive cryptographic workloads
  - ♦ Programmable: support for algorithms within existing protocols, support for new algorithms
- ❖ Motivation
  - ♦ Cipher kernel analyses and characterizations
- ❖ Solution - hardware/software co-design
  - ♦ Software: crypto-specific ISA
  - ♦ Hardware: efficient co-processor implementation
- ❖ Results
  - ♦ More than 2 times faster than a high-end general purpose processor and orders of magnitude less area and power

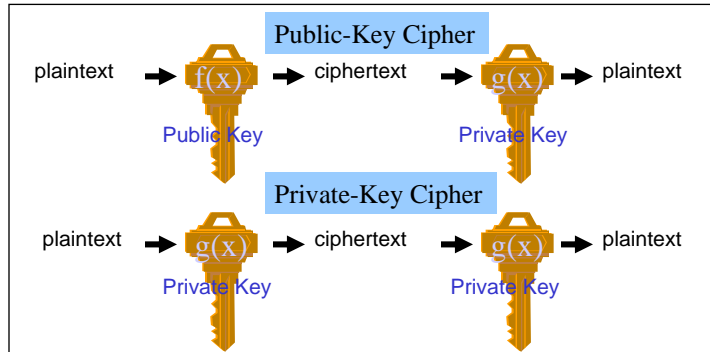


Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Cryptography

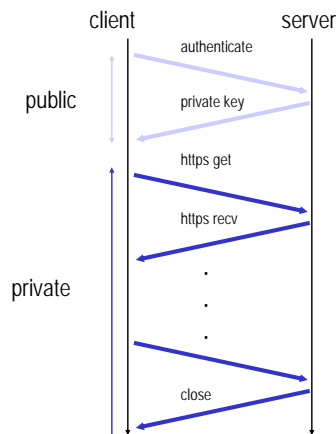
- ❖ Definitions:
  - ♦ encryption vs. decryption
  - ♦ public-key cipher vs. private-key cipher
- ❖ Public-secret key ciphers used in most protocol standards



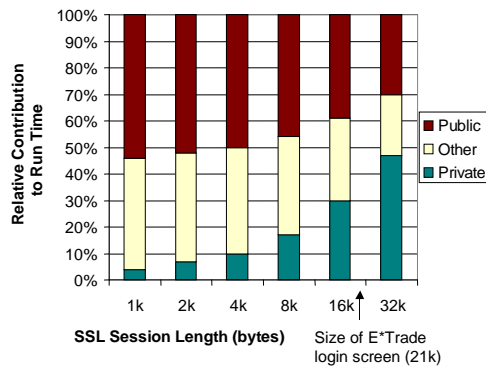
Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# SSL Session Breakdown Focus: Private-Key Ciphers



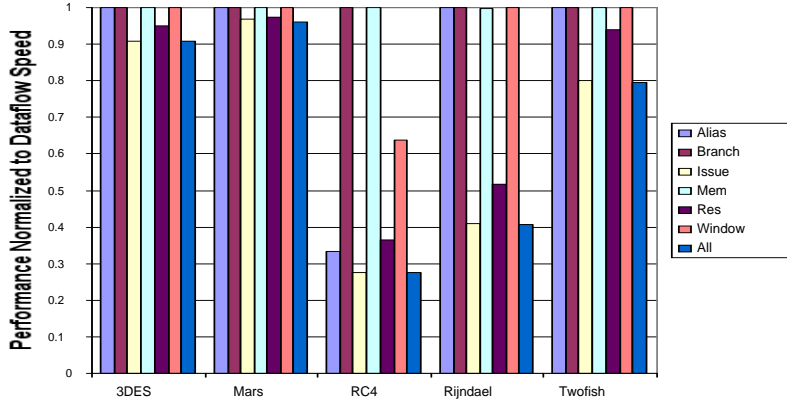
SSL Characterization by Session Length



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Cipher Bottleneck Analysis



- ❖ Likely bottlenecks - issue width, resources
- ❖ Possible bottlenecks - memory aliases, window size
- ❖ Not a bottleneck - branch mispredictions, memory latency

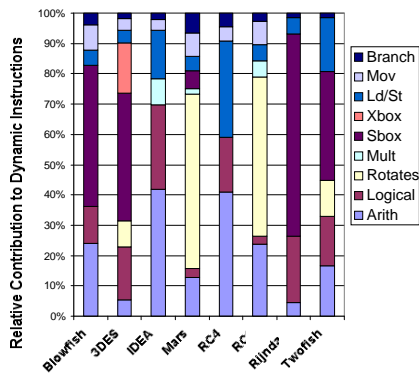


Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Cipher Kernel Characterization

Characterization of Cipher Kernel Operations



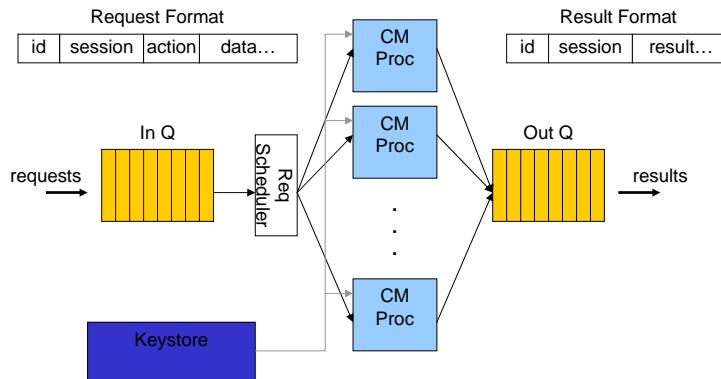
- ❖ SBOX - substitutions
- ❖ XBOX - permutations
- ❖ IDEA, Mars, RC4, and RC6 rely on arithmetic computations; benefit from more resources (multiplies) and faster operations (rotates)
- ❖ Blowfish, 3DES, Rijndael and Twofish rely on substitutions; benefit from increased memory bandwidth



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

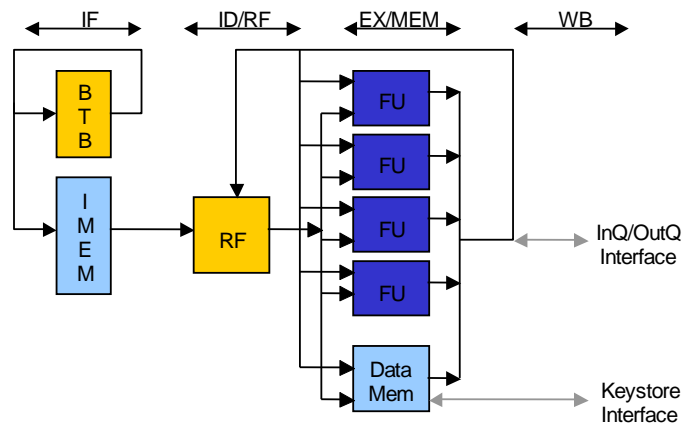
# CryptoManiac System Architecture



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

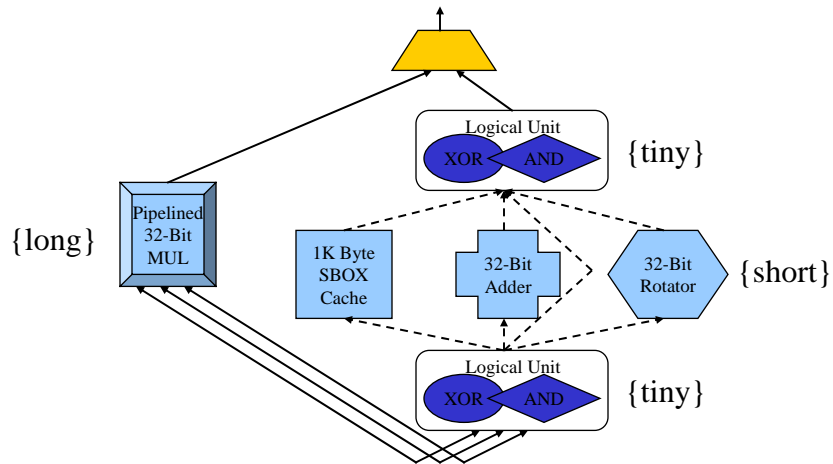
# CryptoManiac Processing Element



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## CryptoManiac Combining Functional Unit



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## CryptoManiac ISA

**bundle** := <inst><inst><inst><inst>

**inst** := <operation pair><dest><operand 1><operand 2><operand 3>

**operation pair** := <short><tiny>|<tiny><short>|<tiny><tiny>|<long><nop>

**tiny** := <xor> | <and> | <signext> | <nop>

**short** := <add> | <addinc> | <sub> | <rot> | <sbox> | <nop>

**long** := <mul> | <mulmod>

Examples:

Instruction

Add-Xor R4, R1, R2, R3

And-Rot R4, R1, R2, R3

And-Xor R4, R1, R2, R3

Expression

$R4 \leftarrow (R1+R2) \otimes R3$

$R4 \leftarrow (R1 \& \& R2) \ll R3$

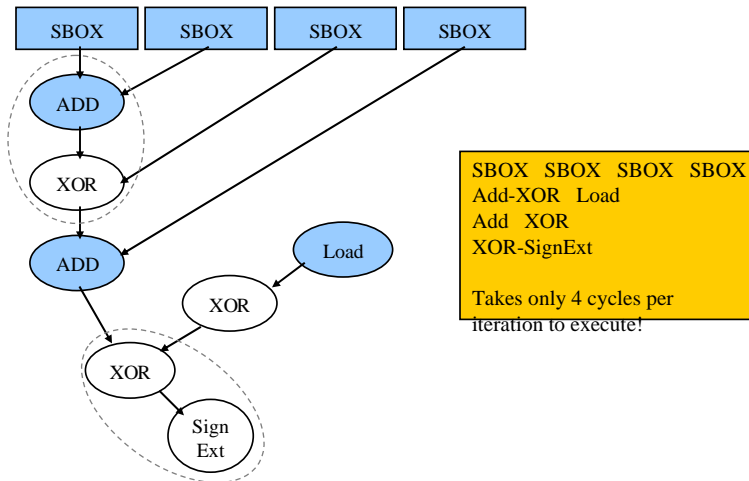
$R4 \leftarrow (R1 \& \& R2) \otimes R3$



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## Scheduling Example: Blowfish



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## Design Methodology

- ❖ Kernels were hand-scheduled and then validated with the super optimizer for accuracy
- ❖ Physical design estimates generated using Synopsys synthesis tools and Cacti 2 cache compiler for 0.25um technology
- ❖ Timing estimates are based on
  - ♦ EX synthesis estimate + bypass latency
- ❖ Area estimates are based on
  - ♦ EX synthesis estimate + array sizes + 10% control/datapath overhead
- ❖ Power estimates are based on
  - ♦ EX synthesis estimate + (array access energy \* array access frequency)



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## Timing and Area Estimates

Timing and Area Estimates for Various CryptoManiac Configurations

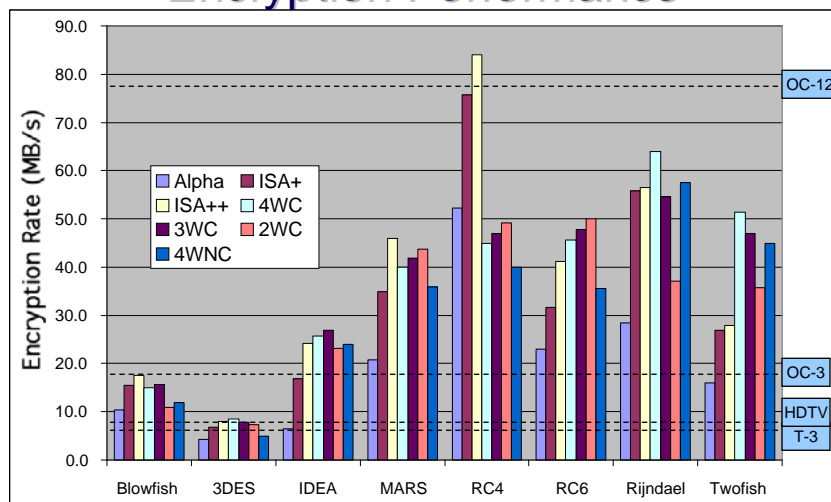
	4W Comb	3W Comb	2W Comb	4W NoComb
Timing Estimate	2.78 ns	2.66 ns	2.54 ns	2.76 ns
Area Estimate	1.39mm x 1.39mm	1.33mm x 1.33mm	1.26mm x 1.26mm	1.3mm x 1.3mm
Power Estimate	606.37 mW	593.51 mW	568.50 mW	586.86 mW
Critical Path	byps-lgc-add-lgc	byps-lgc-add-lgc	byps-lgc-add-lgc	byps-rotate



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

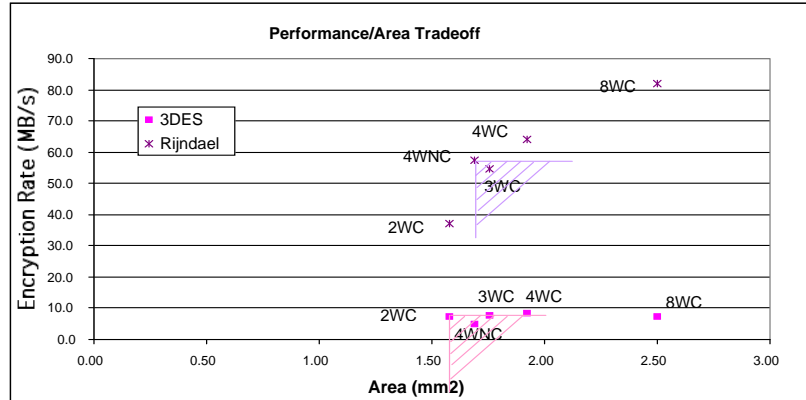
## Encryption Performance



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## Special Case Studies: 3DES and Rijndael



- ❖ System-level performance studies (in paper) show that CryptoManiac can service high bandwidth network and disk I/O traffic with one half as many processing elements (compared to Alpha 21264)



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## Conclusion and Future Work

- ❖ An efficient 4-wide VLIW cryptographic co-processor design called the CryptoManiac
  - ♦ Instruction combining - efficient utilization of clock cycle
  - ♦ Rijndael runs 2.25 times faster with 1/100th area and power of a 600MHz Alpha processor
- ❖ Assess the cost of programmability in the CryptoManiac by comparing design and performance of
  - ♦ A dedicated hardware Rijndael implementation (no programmability)
  - ♦ A FPGA Rijndael implementation (hardware programmability)
  - ♦ CryptoManiac (software programmability)
- ❖ These results make a very strong case for application-specific architecture optimization; we are exploring this approach in other program domains



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# BACK-UP



## Related Work

- ❖ Hardware only designs specific to a particular algorithm, both in the public and secret-key space
  - ◆ Examples are the DES and 3DES implemented by Shiva, IBM, and Hi-Fn.
- ❖ Research of programmable hardware FPGAs, both in the public and secret-key space
- ❖ Architectural extension has been added for the PowerPC instruction set by Shi & Lee for general permutation

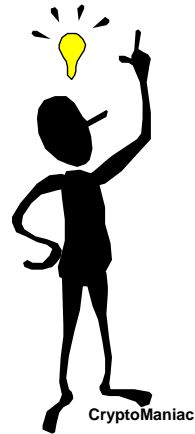


Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## My Research Contribution

- ❖ Design and implementation of the CryptoManiac co-processor
  - ♦ Hardware models of CryptoManiac
    - ♦ 8WC, 4WC, 3WC, 2WC, and 4WNC
  - ♦ ISA and scheduling of kernels
- ❖ Timing, area, power, and performance analyses of the CryptoManiac co-processor
- ❖ Design and implementation of the super optimizer
  - ♦ Instruction combination study
  - ♦ Automatic generation of varied width schedules
- ❖ Publication - ISCA 2001



Advanced Computer Architecture Lab  
University of Michigan

## Benchmark Suite

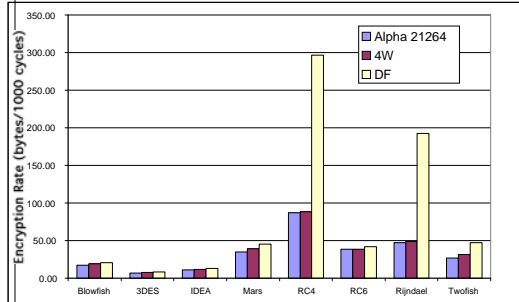
Cipher	Key Size	Blk Size	Rnds/Blk	Author	Application
3DES	112	64	48	CryptSoft	SSL, SSH
Blowfish	128	64	16	CryptSoft	Norton Utilities
IDEA	128	64	8	Ascom	PGP, SSH
Mars	128	128	16	IBM	AES Candidate
RC4	128	8	1	CryptSoft	SSL
RC6	128	128	18	RSA Security	AES Candidate
Rijndael	128	128	10	Rijmen	AES Standard
Twofish	128	128	16	Counterpane	AES Candidate



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Cipher Throughput Analysis



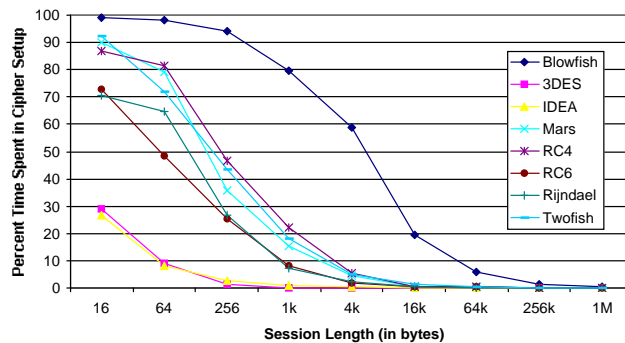
- ❖ Alpha 21264 vs. 4W
  - ♦ All except Mars and Twofish were within 10% of the actual machine tests
  - ♦ Mars 11%, Twofish 15%
- ❖ Alpha 21264 vs. DF
  - ♦ Blowfish, IDEA, and RC6 are running within 20% of DF performance
  - ♦ Mars 29%, Twofish 76%
  - ♦ RC4 and Rijndael are outliers



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Cipher Relative Run Time Cost Focus: Kernel Loop



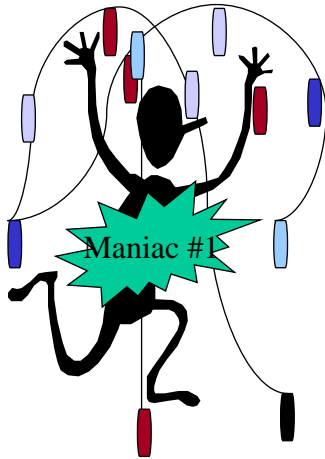
- ❖ 3DES and IDEA are small even for 16 byte sessions
- ❖ Mars, RC4, RC6, Rijndael, and Twofish drop well below 10% for 4k+ byte sessions
- ❖ Blowfish is outlier, drops below 10% only for 64k+ byte sessions



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## The CryptoManiac Processor



- ❖ A 4-wide 32-bit VLIW machine with no cache and a simple branch predictor
- ❖ Supports a triadic (three input operands) ISA that permits combining of most cryptographic operation pairs for better clock cycle utilization
- ❖ Can be combined into chip multiprocessor configurations for improved performance on workloads with inter-session and inter-packet parallelism



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## The Super Optimizer



- ❖ Validate hand-scheduled kernel results
- ❖ Automate generation of optimized kernels for the various CryptoManiac architecture studied
- ❖ Instruction combination studies give insight as to possibly eliminate unnecessary hardware

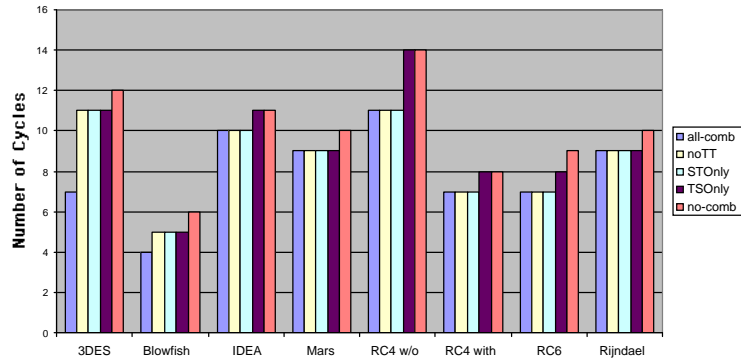


Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Instruction Combination Study

Scheduling with Various FU Configurations

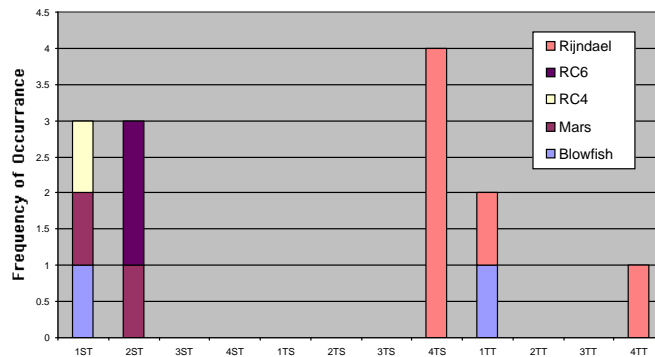


Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

# Instruction Combining Characteristics

Combinations Breakdown by Kernel

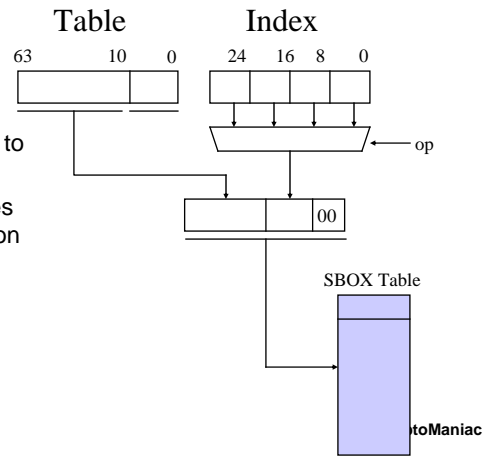


Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## SBOX Instruction Semantics

- ❖ SBOX instruction eliminates address generation
  - ♦ All SBOX tables are aligned to a 1k byte boundary
  - ♦ Address generation becomes zero-latency bit concatenation
- ❖ Stores to SBOX storage are not visible by later SBOX's until an SBOXSYNC is executed



Advanced Computer Architecture Lab  
University of Michigan

## Architectural Extensions

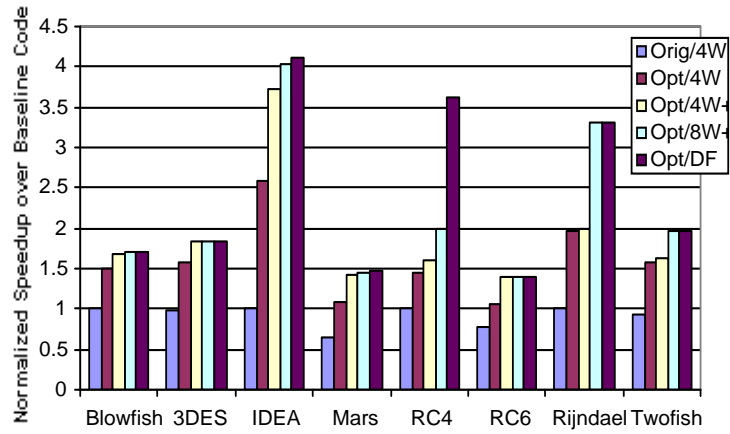
- ❖ All instructions are limited to two register input operands and one register output
  - ♦ ROL and ROR (rotates) for 64 and 32-bit data types
  - ♦ ROLX and RORX support a constant rotate of a register input, followed by an XOR with another register input
  - ♦ MULMOD computes the modular multiplication of two register values modulo the value 0x10001
  - ♦ SBOX speeds the accessing of substitution tables with 256-entry tables and 32-bit contents
  - ♦ XBOX implements a portion of a full 64-bit permutation



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac

## Performance of ISA Extensions



Advanced Computer Architecture Lab  
University of Michigan

CryptoManiac