Efficient Hardware Accelerator for IPSec based on Partial Reconfiguration on Xilinx FPGAs

Ahmad Salman Marcin Rogawski Jens-Peter Kaps

Cryptographic Engineering Research Group (CERG)
`http://cryptography.gmu.edu`
Department of ECE, Volgenau School of Engineering,
George Mason University, Fairfax, VA, USA

Int. Conference on ReConFigurable Computing and FPGAs
ReConFig 2011

Outline

1. Introduction
2. Previous Work
3. Methodology
4. Results
5. Summary

Introduction

- Internet Protocol Security (IPSec) provides security against attacks on data transmitted over the Internet.
- Provides
  - Authentication → Information Source
  - Confidentiality → Encryption
  - Data Integrity → Data alteration

Supported Protocols

- IPSec uses a series of protocols to provide security services
  - The Encapsulating Security Payload (ESP) Protocol.
  - The Authentication Header (AH) protocol.
  - The Internet Key Exchange (IKEv2) protocol in version two.
- These protocols make use of various cryptographic algorithms.

Supported Protocols

Protocol	Security Service Provided	Supported Algorithm
ESP	confidentiality through encryption and optional data integrity	AES in CBC or CTR mode and AES-XCBC-MAC-96
AH	connectionless integrity and data origin authentication	HMAC-SHA1-96, AES-XCBC-MAC-96, HMAC-SHA-256
IKE	negotiates connection parameters	Diffie-Hellman scheme in 1024 or 2048 bits groups and AES in PRNG mode

Table: IPSec Supported Protocols and Algorithms

IPSec Implementations

Software

Hardware

IPSec Implementations

Software

← Flexible

Fast →

Hardware

FPGA Platforms

- Among popular implementations of IPSec in hardware are those that target FPGAs

Problem

- Resource limited devices.
- More resources = more money.

FPGA Platforms

- Among popular implementations of IPSec in hardware are those that target FPGAs

Problem

- Resource limited devices.
- More resources = more money.

Solution

- Hardware/Software co-design.
- Partial Reconfiguration.

Implementations using Partial Reconfiguration

Authors	Embedded Processor	Hardware	Software	Implementation
G. Gogniat et al.	No	AES in Different modes	No	IPSec
I. Gonzales et al.	Microblaze	AES, RC4, IDEA	PR Initiation	VOIP SSL
I. Gonzales et al.	Microblaze	3DES, AES	MD5	SSH
K. Anjo, T. Awashima	DPR-1	AES, DES, CAST128,256, MD5	HMAC	IPSec

Other Implementations

Author	Implementation	Hardware	Software	Application
A. Dandalis, V. Prasanna	AES Finalists	MARS, RC6, Rijndael, Serpent, Twofish	No	IPSec
KAME Project	IPSEC Supported Algorithms	No	Racoon	IPSec
J. Lu, J. Lockwood	AES, HMAC-MD5, HMAC-SHA1	AES, HMAC-MD5, HMAC-SHA1	Key Negotiation	IPSec
Commercial Products	FortiGate, Helion Crypto Accelerator 4000			IPSec SSH HTTPS

Proposed Design

Table: Hardware-Software co-design implementation details of proposed IPSec system

Implementation		Application
In Hardware	In Software	
AES	CBC, CTR modes	ESP
	MAC-XCBC-96	AH
	XCBC-PRF-128	IKEv2
SHA-256	HMAC	AH
MODEXP	Pre-Calculations	IKEv2
-	Round Robin scheduling algorithm	PR trigger

Queues

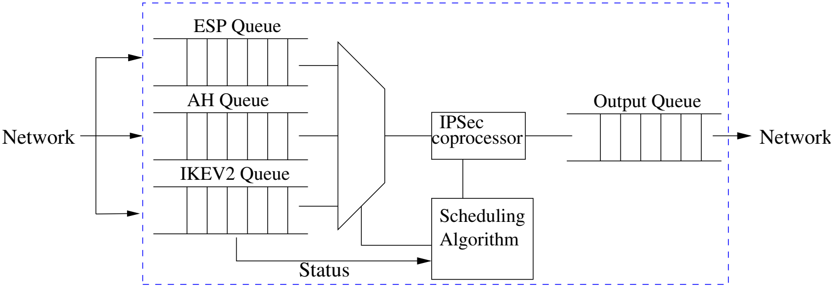


Figure: Synchronization Circuit Between Hardware and Software

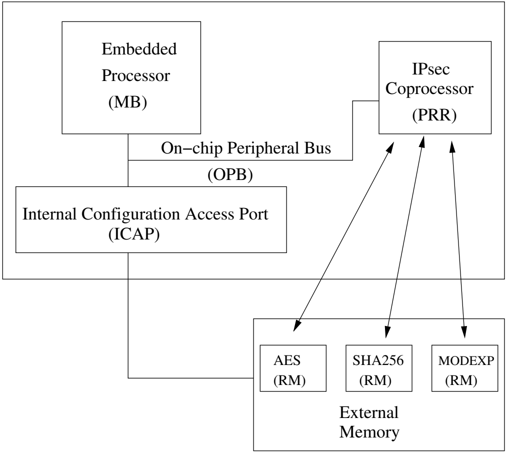
Partial Reconfiguration

- Partial Reconfiguration (PR) is a process of configuring a portion of the FPGA while the other part is still running.

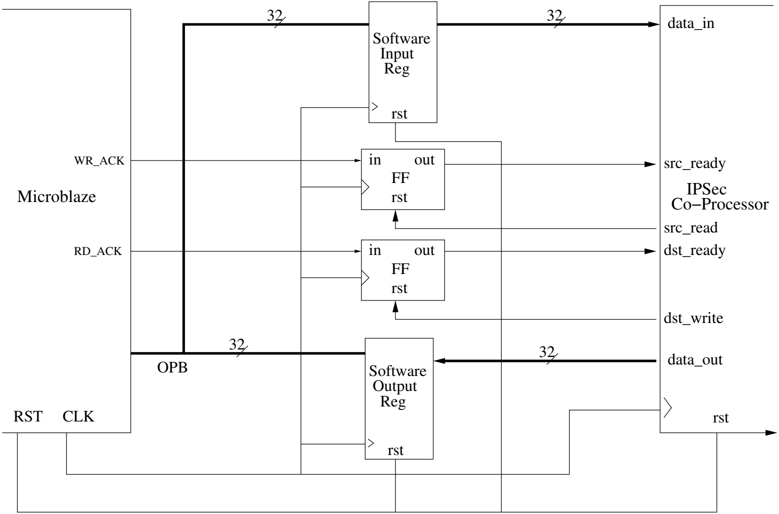
- A relatively new technique
  - Altera Stratix V.

- A PR system is divided into
  - Static region known as Base Region (BR).
  - Dynamic regions known as Partial Reconfigurable Regions (PRR).
  - Reconfigurable Modules (RMs)

Partial Reconfigurable Hardware in the System



Hardware/Software Synchronization Circuit



System Software

- Drivers for the hardware peripherals.
- Internal Control Access Port (ICAP) API initialization.
- Modes of Operations.
  - Cipher Block Chaining (CBC) mode.
  - Counter (CTR) mode.
- Hashed Message Authentication Code (HMAC) Calculation
- Pre-Computations.

Experiment Methodology

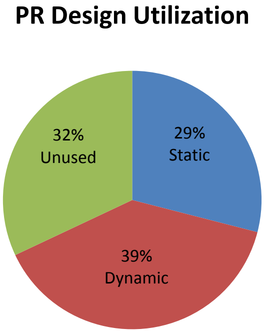
- Implementation of individual cryptographic algorithms in non-PR designs.
- Creation of the PR design with all three algorithms.
- Assign Tasks to the system processor through the scheduler.

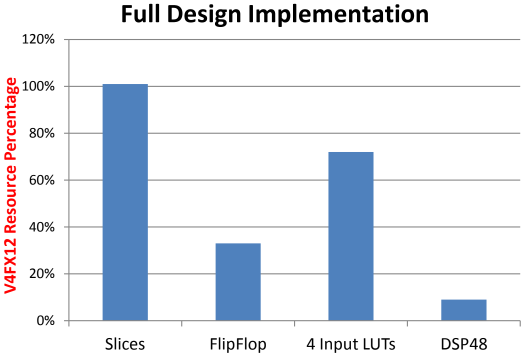
PR Implementation Results

Table: Summary for Implementations on XC4VFX12 Virtex-4 FPGA

Device Utilization Summary	PR Design		Non-PR Design
	Static	Dynamic	
Resource Logic	Used	Used	Used
Number of Slices	1588	2148	5506
Number of Slice Flip Flops	1566	1008	3906
Number of 4 input LUTs	2059	3600	8140
Number of DSP48	0	3	3
Number of FIFO16/RAMB16s	33	0	0

Utilization Percentage



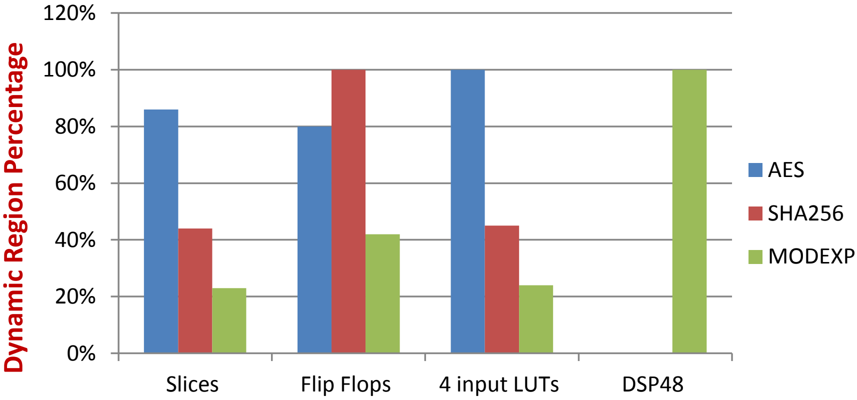


Independent Cores Implementations

Table: Summary for Implementations on XC4VFX12 Virtex-4 FPGA

Device Utilization Summary	Implementations for each core independently		
	AES	SHA-256	MODEXP
Resource Logic	Used	Used	Used
Number of Slices	1862	952	499
Number of Slice Flip Flops	807	1008	421
Number of 4 input LUTs	3600	1632	861
Number of DSP48	0	0	3
Number of FIFO16/RAMB16s	0	0	0

Dynamic Region Utilization

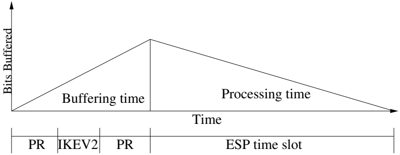


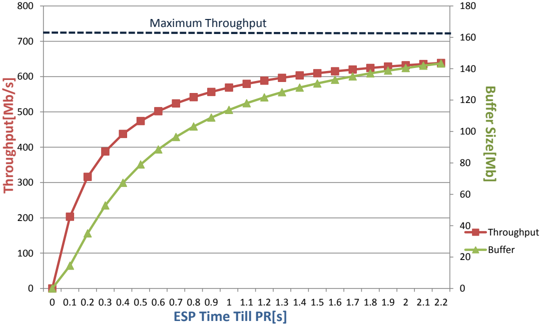
AES Throughput in a PR design

Work	AES Throughput [MB/S]
A. Dandalis and V. Prasanna	353
Y. Hasegawa et al.	363
G. Gogniat et al.	422
Our Design	711

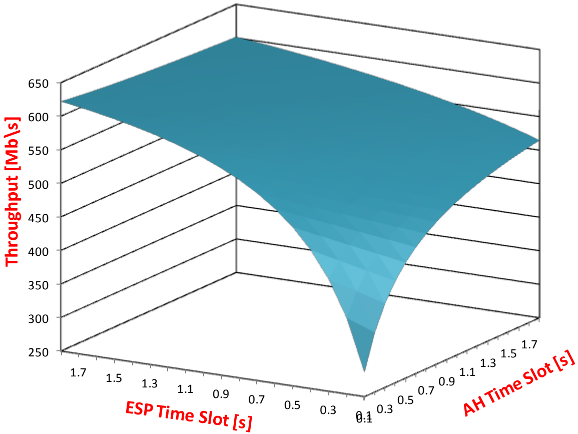
Table: Comparing our AES throughput to other implementations

ESP Time-Slot





System Latency Vs Throughput



Conclusion

- The proposed design provides a low cost solution for IPSec in Hardware.
- A scheduling algorithm was used to handle task assignments.
- Benefits of implementing IKEV2 as RM.
- Results show that the design performs well with high traffic networks.

Future Work

- Implementing the design on Faster FPGA families.
- Use new tools.
- Extending the number of supported algorithms.
- Implementing the AES core as part of the static region.

Thanks for your attention.