FPGA Implementations of the Hummingbird Cryptographic Algorithm

Xinxin Fan and Guang Gong
E&CE Department, University of Waterloo, CANADA

Ken Lauffenburger
Aava Technology LLC, USA

Troy Hicks
Revere Security Corporation, USA

Introduction

The widespread deployment of various wireless networks such as mobile ad-hoc networks, sensor networks, mesh networks, personal area networks and radio frequency identification (RFID) systems is making possible a world of **pervasive computing** a reality. While the wireless communication technology and devices under development are enabling our march toward the era of pervasive computing, the **security** and **privacy** concerns in pervasive computing remains a serious impediment to widespread adoption of emerging technologies. Employing **lightweight cryptographic primitives** that can perform strong **authentication** and **encryption** on **resource-constrained** smart devices is a promising solution to overcome those concerns in the era of pervasive computing.

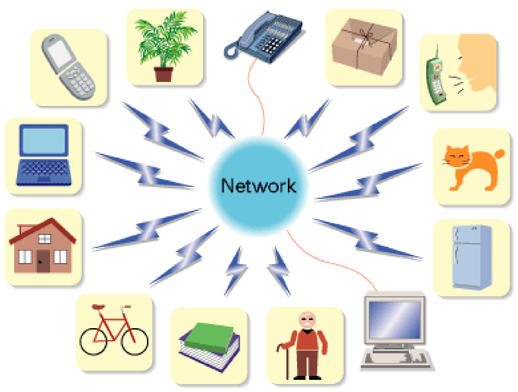


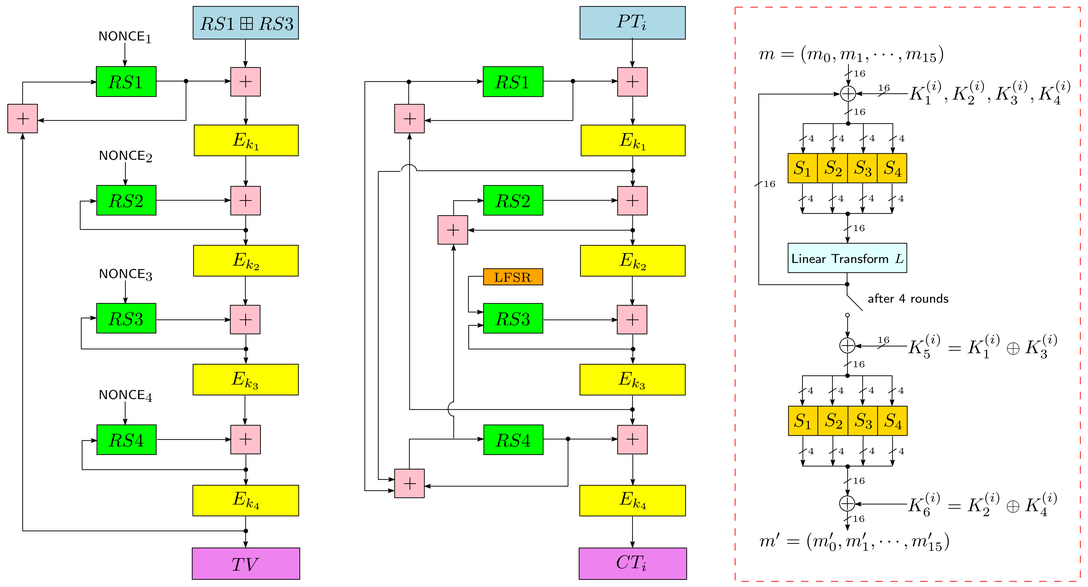
Figure 1: A World of Pervasive Computing.

Hummingbird in a Nutshell

Hummingbird is a rotor machine and has a hybrid structure of block cipher and stream cipher with 16-bit block size, 256-bit key size as well as 80-bit internal state.

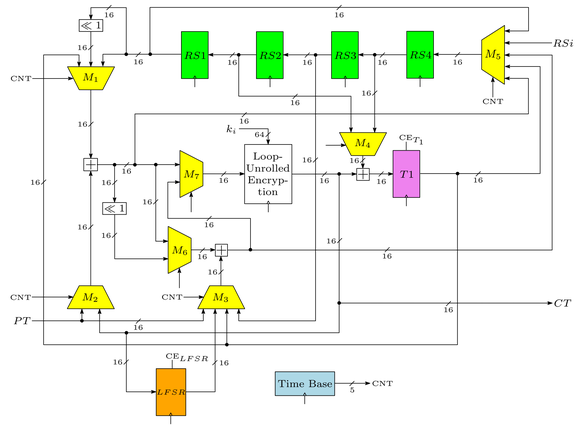
- 4 identical block ciphers with 16-bit input and 16-bit output
- 4 16-bit registers acting as 4 rotors
- A 16-bit linear feedback shift register (LFSR)
- Simple arithmetic and logic operations ($\oplus, \boxplus, \boxminus, \ll$)

Description of Hummingbird Cryptographic Algorithm



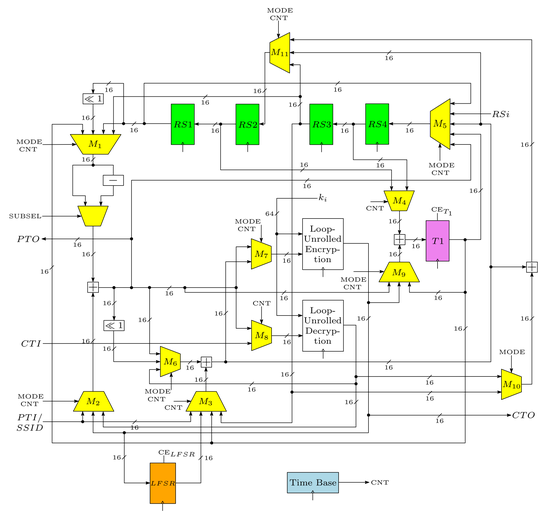
(a) Initialization Process (b) Encryption Process (c) 16-bit Block Cipher

Speed Optimized Hummingbird Encryption Core



- 16-bit block cipher is implemented using a loop-unrolled architecture
- 119 I/O pins and 273 slices on the Spartan-3 FPGA
- 20 clock cycles for the initialization process
- 4 clock cycles for encrypting one 16-bit plaintext block

Speed Optimized Hummingbird Encryption/Decryption Core



- 143 I/O pins and 558 slices on the Spartan-3 FPGA
- 20 clock cycles for the initialization process
- 4 clock cycles for encrypting/decrypting one 16-bit plaintext block

Performance Comparisons

Cipher	Key Size	Block Size	Total Occupied Slices	Max. Freq. (MHz)	Throughput (Mbps)	Efficiency (Mbps/# Slices)
Hummingbird	**256**	**16**	**273**	**40.1**	**160.4**	**0.59**
PRESENT [Poschmann'09]	80	64	176	258	516	2.93
	128	64	202	254	508	2.51
PRESENT [Guo et al.'08]	80	64	271	–	–	–
XTEA [Kaps'08]	128	64	254	62.6	36	0.14
			9,647	332.2	20,645	2.14
ICEBERG [Standaert et al.'08]	128	64	631	–	1,016	1.61
SEA [Mace et al.'08]	126	126	424	145	156	0.368
AES [Chodowiec & Gaj'03]	128	128	522	60	166	0.32
AES [Good & Benaissa'05]			17,425	196.1	25,107	1.44
			264	67	2.2	0.01
AES [Rouvroy et al.'04]			1,214	123	358	0.29
AES [Bulens et al.'08]			1,800	150	1700	0.9

Conclusions and Outlook

- The first efficient FPGA implementations of the ultra-lightweight cipher Hummingbird
- 20 clock cycles for initialization and 4 clock cycles for encryption/decryption
- Hummingbird can achieve larger throughput with smaller area requirement, when compared to other lightweight FPGA implementations of block ciphers in the literature
- Hummingbird is an ideal cryptographic primitive for resource-constrained environments
- Future work: low power ASIC implementations for low-cost RFID tags

References

[1] D. Engels, X. Fan, G. Gong, H. Hu, and E. M. Smith, "Hummingbird: Ultra-Lightweight Cryptography for Resource-Constrained Devices", to appear in the proceedings of *The 14th International Conference on Financial Cryptography and Data Security - FC 2010*, Berlin, Germany: Springer-Verlag, 2010.

[2] X. Fan, H. Hu, G. Gong, E. M. Smith and D. Engels, "Lightweight Implementation of Hummingbird Cryptographic Algorithm on 4-Bit Microcontrollers", *The 1st International Workshop on RFID Security and Cryptography 2009 (RISC'09)*, pp. 838-844, 2009.

Acknowledgements

This work is supported by an NSERC Discovery Grant and an NSERC Strategic Project Grant.