Hardware-Software Codesign of RSA for Optimal Performance vs. Flexibility Trade-off

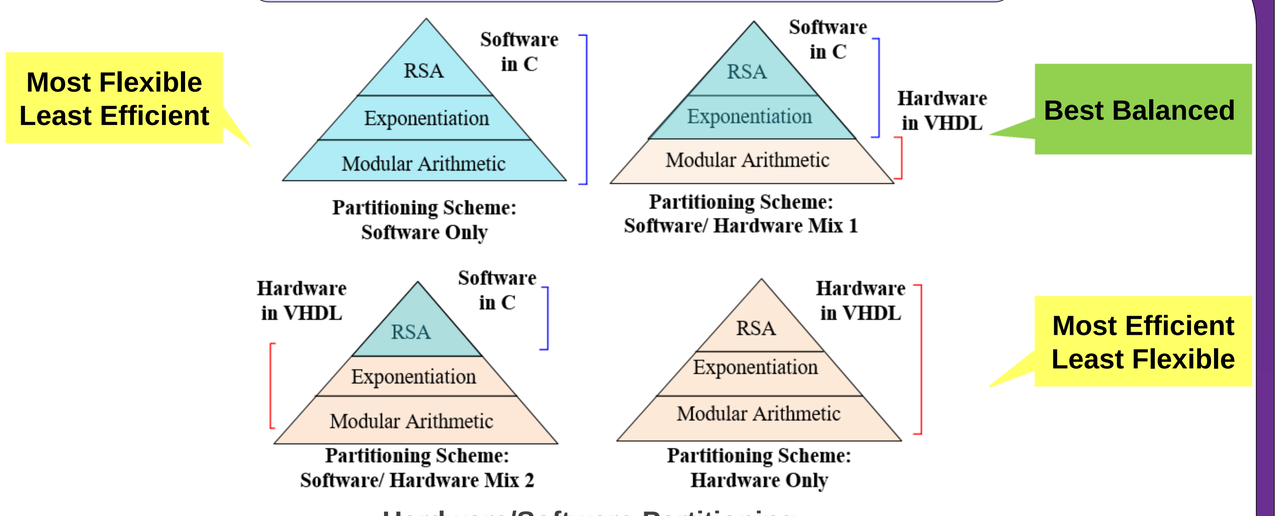
Malik Umar Sharif, Rabia Shahid, Marcin Rogawski and Kris Gaj

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

Motivation

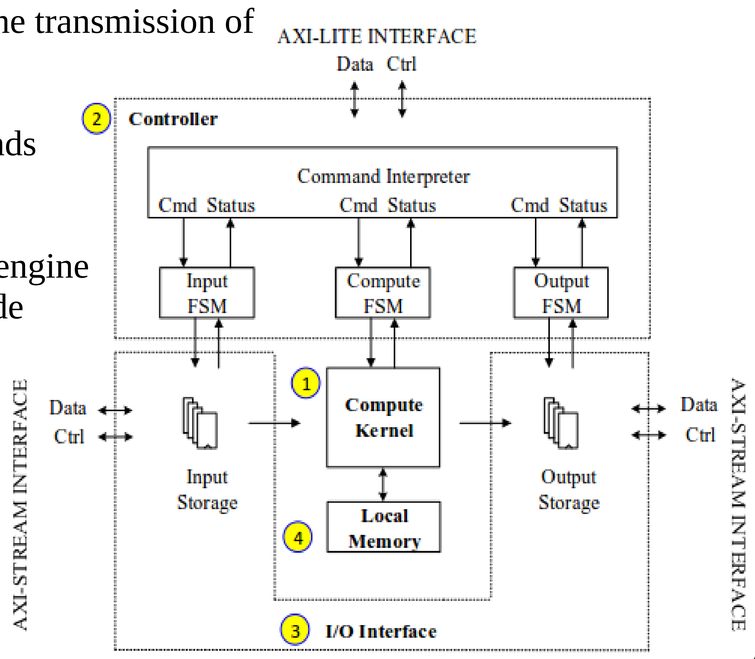
- New SoC device (Zynq-7000) and new software (Vivado) facilitating hardware/software codesign
- Combining flexibility and scalability of software with performance and low power/energy consumption of hardware
- Significant speed-up over purely software implementation
- Minimizing the communication overhead between the microprocessor and hardware accelerator

Design Methodology



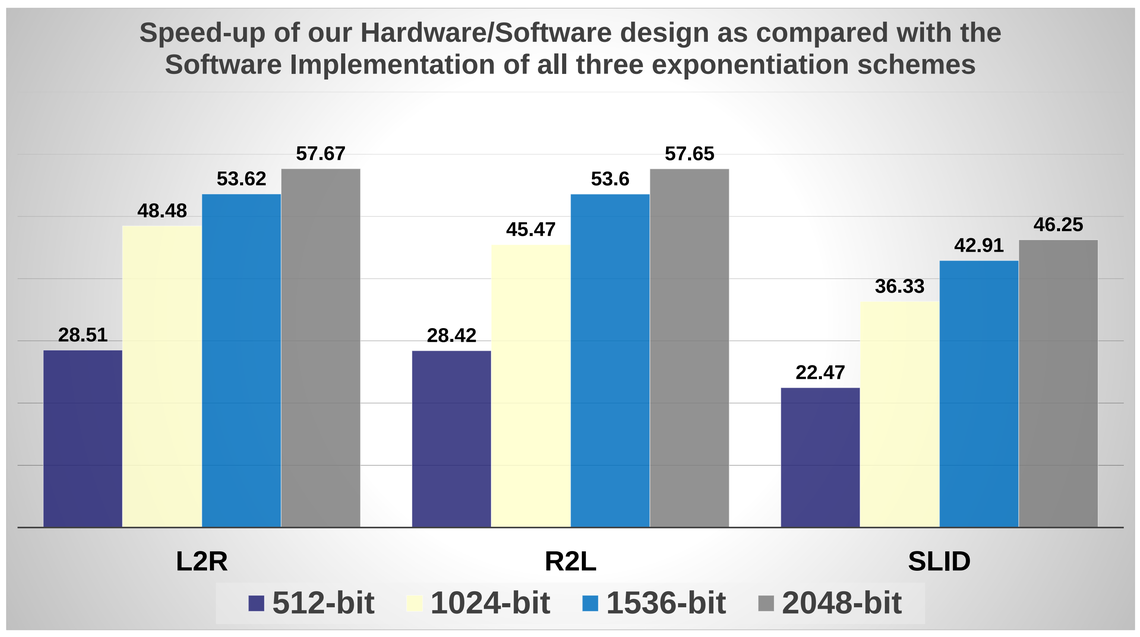
Hardware/Software Partitioning

- **Device - Xilinx Zynq-7000 SoC**
  - Processing System (PS) – ARM based Microprocessor System
  - Programmable Logic (PL) – a 28nm Xilinx reconfigurable logic equivalent to Artix-7 FPGA
- **Platform - ZedBoard Development Board**
  - Xilinx Zynq-7000 All Programmable SoC XC7Z020-CLG484-1
  - Bare Metal Mode
- **Software – C program on ARM in PS**
  - Based on RELIC library by D.F. Aranha and C.P.L. Gouvêa
  - Three modular exponentiation (ME) schemes: Left-to-Right (L2R), Right-to-Left (R2L) and Sliding Window (SLID), selected at run time
  - Four operand sizes: 512, 1024, 1536, 2048 bits, selected at run time
- **Hardware Accelerator - RTL VHDL design in PL**
  1. ***Compute Kernel:*** Coprocessor unit to perform compute-intensive tasks
  2. ***Controller:*** Interprets commands sent by PS
  3. ***I/O Interface:*** Interface with the bus that includes argument and result storage
  4. ***Local Memory:*** To store the intermediate results in hardware
- **Communication Interface**
  - AMBA Advanced Extensible Interface 4 (AXI4)
    - AXI4-Stream used for the transmission of arguments and results
    - AXI4-Lite used for the transmission of commands
  - Processing system (PS) is connected to the hardware accelerator through DMA engine to stream data in burst mode
  - Simple DMA Transfer using AXI DMA Core operating at 100 MHz
- **Tools**
  - Xilinx Vivado Design Suite 2015.4
  - Xilinx SDK 2015.4



Generic Hardware Coprocessor for RSA

Results



Comparison of our work with existing implementations

Reference	ME Schemes (Flexibility)/ Operand Sizes (Scalability)	Coprocessor Performs	Device	Freq MHz	Area (LUTs, BRAMs, DSP48)	Time (ms) 1024-bit Operands
HW/ SW Codesign-based Implementations						
This work	L2R, R2L, SLID* / 512, 1024, 1536, 2048-bit	MM	Zynq SoC	100/ 200	10385, 26, 17	6.25
San et al. (2014)	R2L / 512, 1024, 2048-bit	ME	Zynq SoC	100	6224, 0, 62	3.04
Issad et al. (2014)	R2L / 1024-bit	MM	Virtex-5	63	1848, 11, 22	22.25
Uhsadel et al. (2012)	R2L, L2R*, MPL, BFR / 1024-bit	ME	Virtex-4	111	27467, 0, 0	29.37
HW-Only Implementations						
Suzuki et al. (2007)	SLID / 512, 1024, 1536, 2048-bit	ME	Virtex-4	200/ 400	4190, 7, 17	1.71
Song et al. (2011)	R2L / 1024-bit	ME	Virtex-5	447	180, 1, 1	36.37
Wang et al. (2012)	L2R / 1024-bit	ME	Virtex-5	200	5730, 0, 0	679

* denotes ME scheme used for time measurement
MM – Modular Multiplication, ME – Modular Exponentiation,
MPL – Montgomery Powering Ladder, BFR – Blinded Fault Resistant

Conclusions

- **3 exponentiation schemes** and **4 operand sizes**, controlled at run time
- Up to **57x speedup** as compared to the software implementation based on RELIC, running on the same platform
- 2x reduction in speed vs. less-flexible design by San et al. with the entire ME (R2L only) in hardware
- Less than **1% communication overhead**
- A novel approach to support **algorithmic and implementation level flexibility**
- **Balanced hardware/software partitioning**
- **Generalizable** to other public-key cryptosystems

Acknowledgements

This poster is based upon work supported by the National Institute of Standards and Technology, U.S. Department of Commerce under Grant No. 60NANB15D058