Looting the LUTs

FPGA Optimization of AES and AES-like Ciphers for Authenticated Encryption

Mustafa Khairallah[1] Anupam Chattopadhyay[1]
Thomas Peyrin[1]

[1]Nanyang Technological University, Singapore

mustafam001@e.ntu.edu.sg

11 December, 2017

Outline

1. Motivation
   - FPGA
   - ASIC Optimization
   - Authenticated Encryption
2. Architecture
3. Zero Overhead Pipeline
4. Linear Functions: Technology Mapping
5. Results

Plan

1. Motivation
   - FPGA
   - ASIC Optimization
   - Authenticated Encryption
2. Architecture
3. Zero Overhead Pipeline
4. Linear Functions: Technology Mapping
5. Results

Outline

1 Motivation
- FPGA
- ASIC Optimization
- Authenticated Encryption

2 Architecture

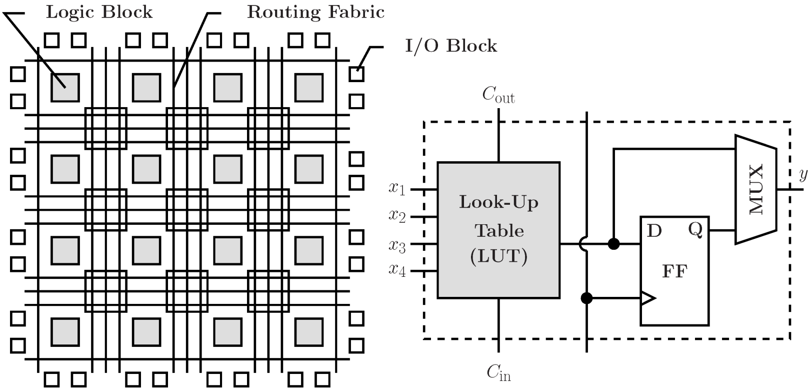
3 Zero Overhead Pipeline

4 Linear Functions: Technology Mapping

5 Results

FPGA Architecture

Motivation



(a) (b) [1]

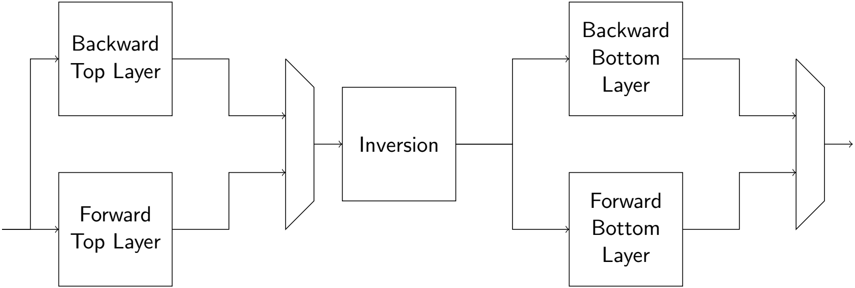
[1] http://www.mdpi.com

Outline

1. Motivation
   - FPGA
   - ASIC Optimization
   - Authenticated Encryption
2. Architecture
3. Zero Overhead Pipeline
4. Linear Functions: Technology Mapping
5. Results

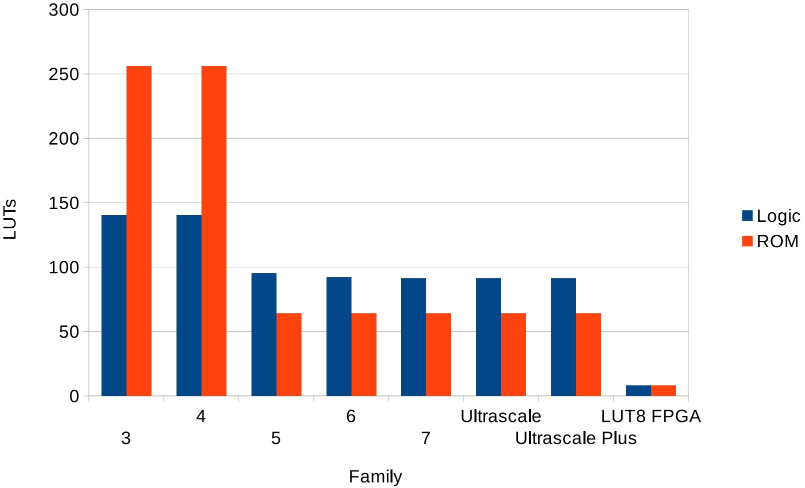
AES Boyar's Sbox: Boolean Optimization [BMP13]

Motivation



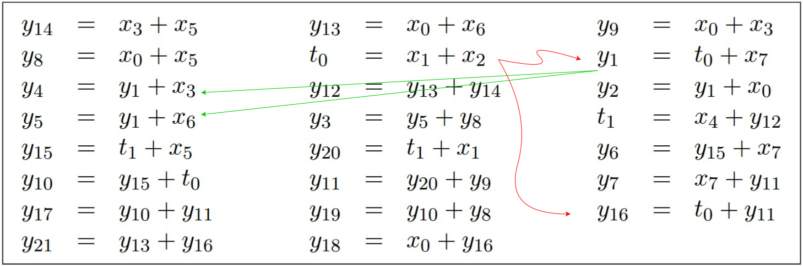
Sbox Evolution:Xilinx FPGA

Motivation



AES Boyar's Sbox:Forward Top Layer [BMP13]

Motivation



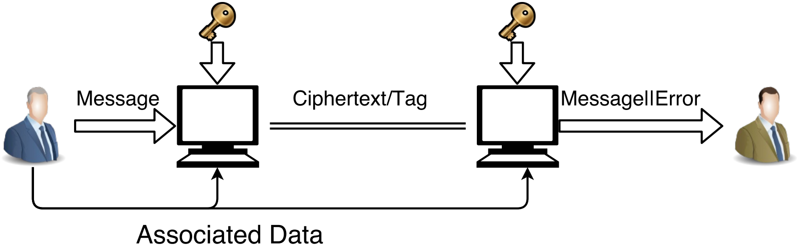
$$
\begin{aligned}
y_{14} &= x_3 + x_5 & y_{13} &= x_0 + x_6 & y_9 &= x_0 + x_3 \\
y_8 &= x_0 + x_5 & t_0 &= x_1 + x_2 & y_1 &= t_0 + x_7 \\
y_4 &= y_1 + x_3 & y_{12} &= y_{13} + y_{14} & y_2 &= y_1 + x_0 \\
y_5 &= y_1 + x_6 & y_3 &= y_5 + y_8 & t_1 &= x_4 + y_{12} \\
y_{15} &= t_1 + x_5 & y_{20} &= t_1 + x_1 & y_6 &= y_{15} + x_7 \\
y_{10} &= y_{15} + t_0 & y_{11} &= y_{20} + y_9 & y_7 &= x_7 + y_{11} \\
y_{17} &= y_{10} + y_{11} & y_{19} &= y_{10} + y_8 & y_{16} &= t_0 + y_{11} \\
y_{21} &= y_{13} + y_{16} & y_{18} &= x_0 + y_{16} & &
\end{aligned}
$$

Outline

1. Motivation
   - FPGA
   - ASIC Optimization
   - Authenticated Encryption
2. Architecture
3. Zero Overhead Pipeline
4. Linear Functions: Technology Mapping
5. Results

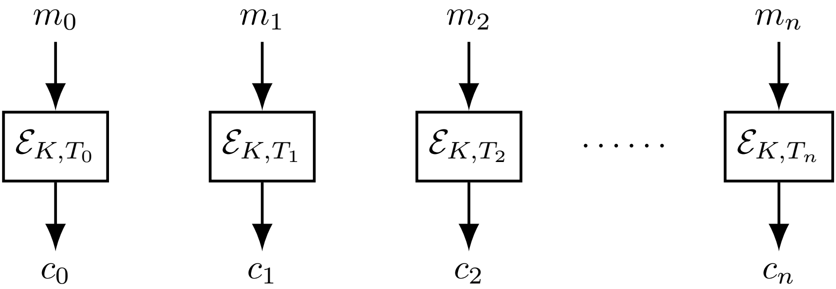
AEAD

Motivation



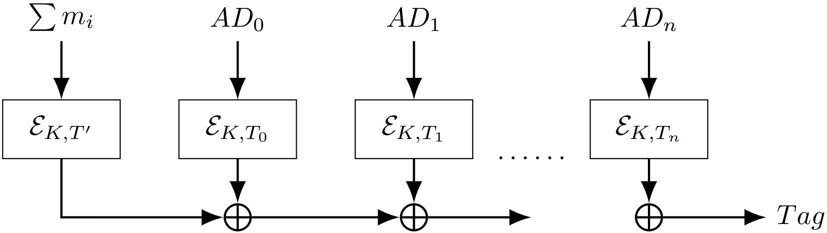
Parallelisable Modes: ECB

Motivation



Parallelisable Modes: PMAC

Motivation



Problem

Motivation

We usually do not take the potential for parallel execution into account in hardware evaluation, as opposed to software.

Plan

1 Motivation
- FPGA
- ASIC Optimization
- Authenticated Encryption

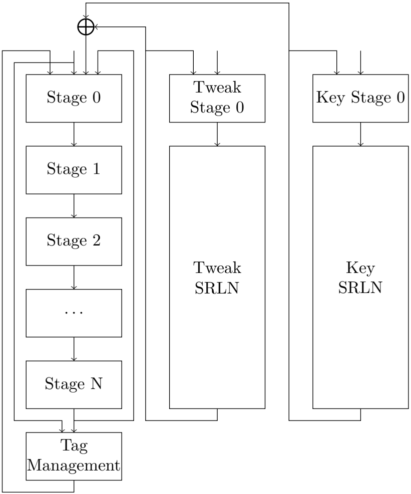
2 Architecture

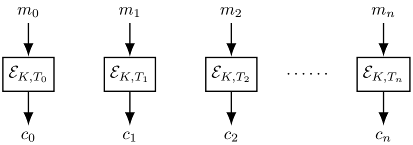
3 Zero Overhead Pipeline

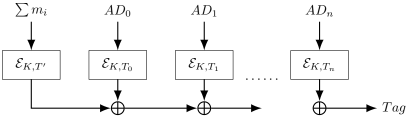
4 Linear Functions: Technology Mapping

5 Results

Parallel AEAD Round Based Architecture







Key Scheduling

In order to minimize the key scheduling overhead, it is performed in only one pipeline stage and then shifted N cycles using SRL.

Pipeline Registers

The pipeline registers can add a huge overhead over the simple round implementation.

Plan

1 Motivation
- FPGA
- ASIC Optimization
- Authenticated Encryption

2 Architecture

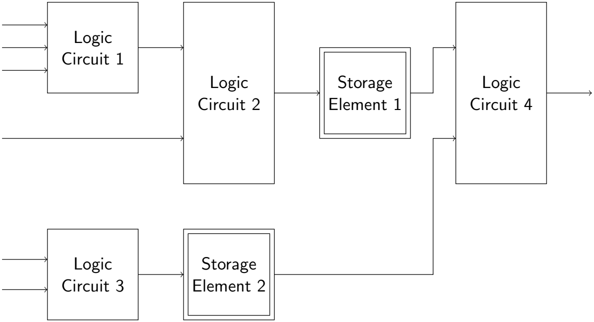
3 Zero Overhead Pipeline

4 Linear Functions: Technology Mapping

5 Results

Zero Overhead Pipeline

Sub-optimal



Design Flow

1. Synthesize the combinational circuit without pipelining.

Design Flow

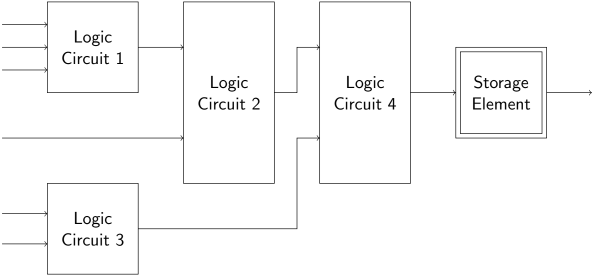
1. Synthesize the combinational circuit without pipelining.
2. Determine the LUT structure and the best locations to insert pipeline registers.

Design Flow

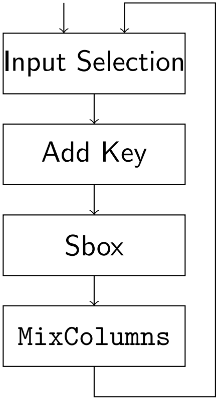
1. Synthesize the combinational circuit without pipelining.
2. Determine the LUT structure and the best locations to insert pipeline registers.
3. Resynthesize and compare the number of slices (the differences should be near 0).

Zero Overhead Pipeline

Optimal



AES



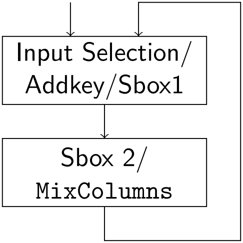
Input Selection

Add Key

Sbox

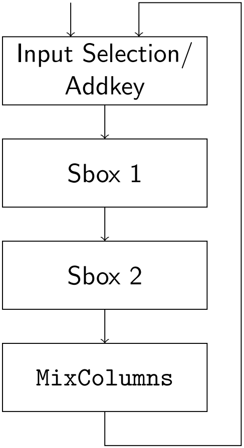
MixColumns

Figure: The AES encryption data path from BSQ+08



Input Selection/ Addkey/Sbox1

Sbox 2/ MixColumns



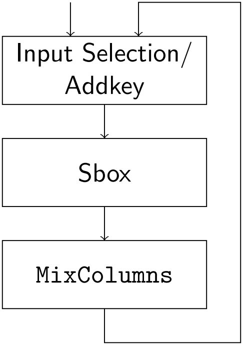
Input Selection/ Addkey

Sbox 1

Sbox 2

MixColumns

LED



Plan

1 Motivation

- FPGA
- ASIC Optimization
- Authenticated Encryption

2 Architecture

3 Zero Overhead Pipeline

4 Linear Functions: Technology Mapping

5 Results

Linear Functions: Technology Mapping

Example: `AES MixColumns`

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$

$$p = 2 \cdot a \oplus 3 \cdot b \oplus c \oplus d$$

Linear Functions: Technology Mapping

Example: AES MixColumns Bit Decomposition

$$\begin{bmatrix} a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & a_0 & 0 \\ 0 & 0 & 0 & a_7 & a_7 & 0 & a_7 & a_7 \\ b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & b_0 & 0 \\ 0 & 0 & 0 & b_7 & b_7 & 0 & b_7 & b_7 \\ b_7 & b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & b_0 \\ c_7 & c_6 & c_5 & c_4 & c_3 & c_2 & c_1 & c_0 \\ d_7 & d_6 & d_5 & d_4 & d_3 & d_2 & d_1 & d_0 \end{bmatrix}$$

Linear Functions: Technology Mapping

Example: `AES` `MixColumns` Bit Decomposition

$$\begin{bmatrix} a_6 & a_5 & a_4 & a_3 & a_2 & a_1 & a_0 & a_7 \\ b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & b_0 & b_7 \\ 0 & 0 & 0 & x & x & 0 & x & 0 \\ b_7 & b_6 & b_5 & b_4 & b_3 & b_2 & b_1 & b_0 \\ c_7 & c_6 & c_5 & c_4 & c_3 & c_2 & c_1 & c_0 \\ d_7 & d_6 & d_5 & d_4 & d_3 & d_2 & d_1 & d_0 \end{bmatrix}$$

$$x = a_7 \oplus b_7$$

Linear Functions: Technology Mapping

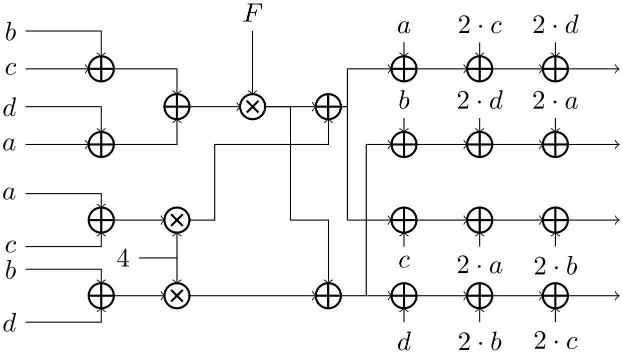
Example: Inverse AES `MixColumns`

$$\begin{bmatrix} E & B & D & 9 \\ 9 & E & B & D \\ D & 9 & E & B \\ B & D & 9 & E \end{bmatrix} = \begin{bmatrix} F & F & F & F \\ F & F & F & F \\ F & F & F & F \\ F & F & F & F \end{bmatrix} \oplus \begin{bmatrix} 1 & 0 & 2 & 2 \\ 2 & 1 & 0 & 2 \\ 2 & 2 & 1 & 0 \\ 0 & 2 & 2 & 1 \end{bmatrix} \oplus \begin{bmatrix} 0 & 4 & 0 & 4 \\ 4 & 0 & 4 & 0 \\ 0 & 4 & 0 & 4 \\ 4 & 0 & 4 & 0 \end{bmatrix}$$

$$p = F \cdot (a \oplus b \oplus c \oplus d) \oplus (a \oplus 2 \cdot c \oplus 2 \cdot d) \oplus 4 \cdot (b \oplus d)$$

Linear Functions: Technology Mapping

Example: Inverse `AES` `MixColumns`



Linear Functions: Technology Mapping

Example: Inverse `AES MixColumns`

Implementation[2]	M^3	$M \cdot N$	Ours
LUTs/output bit	3.375	2.25	1.875

[2] M is the MDS matrix used in `AES MixColumns`

Plan

1. Motivation
   - FPGA
   - ASIC Optimization
   - Authenticated Encryption
2. Architecture
3. Zero Overhead Pipeline
4. Linear Functions: Technology Mapping
5. Results

Results

Algorithm	Family	Impl.	Throughput (Gbps)	Slices	Efficiency (Mbps/slice)
AES Encryption	Virtex 5	Ours	8.0	347	23.00
		BSQ+08	4.5	400	11.20
		LXY13	46.0	3,579	12.88
	Virtex 6	Ours	9.5	247	38.46
		LXY13	64.1	3.121	20.55
AES Decryption	Virtex 5	Ours	6.1	294	20.7
		BSQ+08	4.5	550	7.6
Deoxys-I-128	Virtex 6	Ours	3.8	861	4.5
		CERG	2.2	946	2.57
Deoxys-I-128 Encryption Only	Virtex 6	Ours	3.5	566	6.2
		Axel Poschmann & Marc Stöttinger	1	920	1.12
LED	Spartan 3	Ours	0.51	204	2.5
		APP14	0.19	204	0.97

Future Work

- Automate the pipeline selection flow.
- Design an algorithm for FPGA mapping of other primitives.
- Lightweight implementations of the CAESAR Competition finalists.

Thank you!

Any Questions?!