

AES as a Stream Cipher



Bin ZHOU

Instructor: Dr. Kris. Gaj

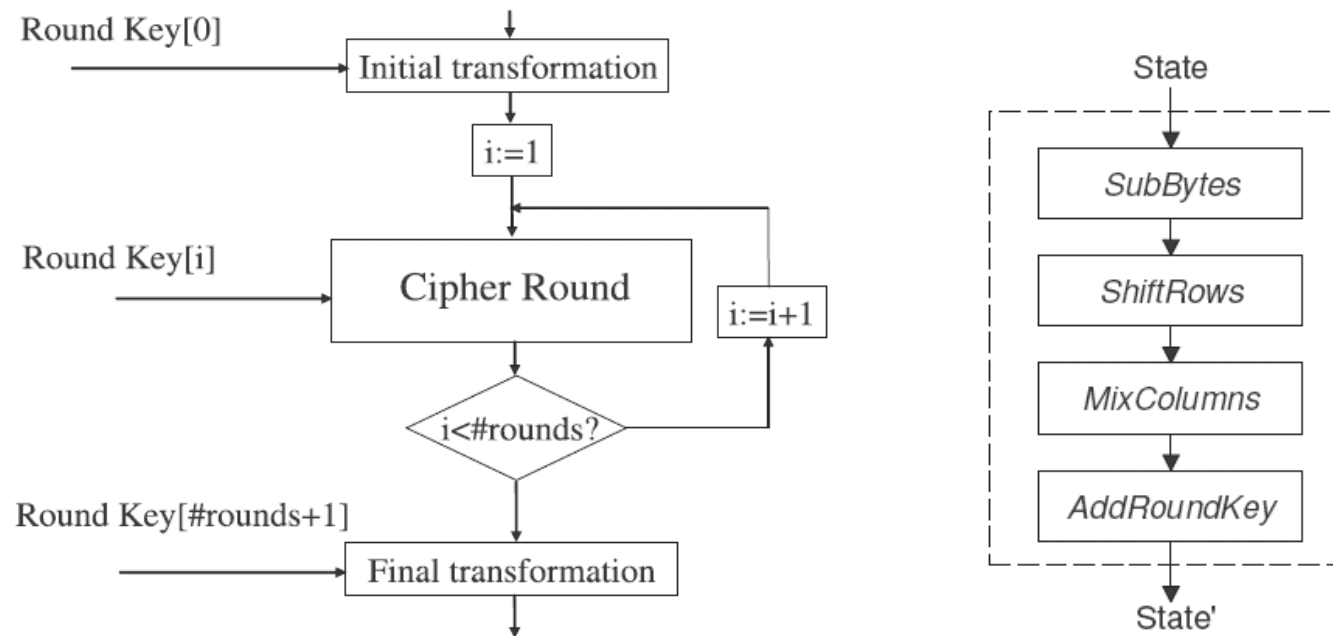
ECE GMU

Outline

- Introduction & Background
- AES & Mode Selection
- On-the-fly Key Scheduling
- Compact Mode Design
- Pure Logic S-Box
- Implementation
- Results
- Q&A

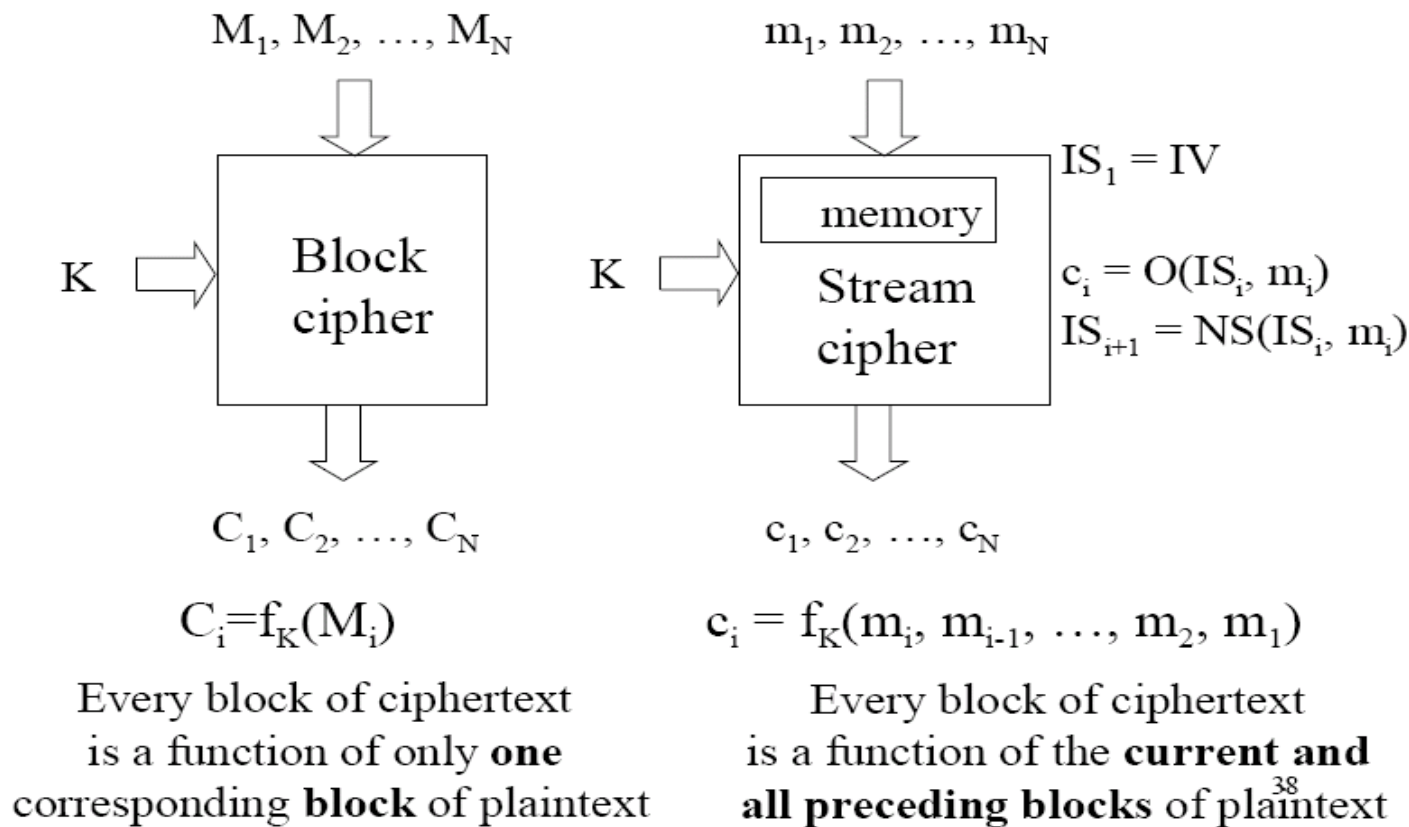
Introduction & Background

□ Advanced Encryption Standard (AES)



Block vs. Stream Ciphers

Block vs. stream ciphers

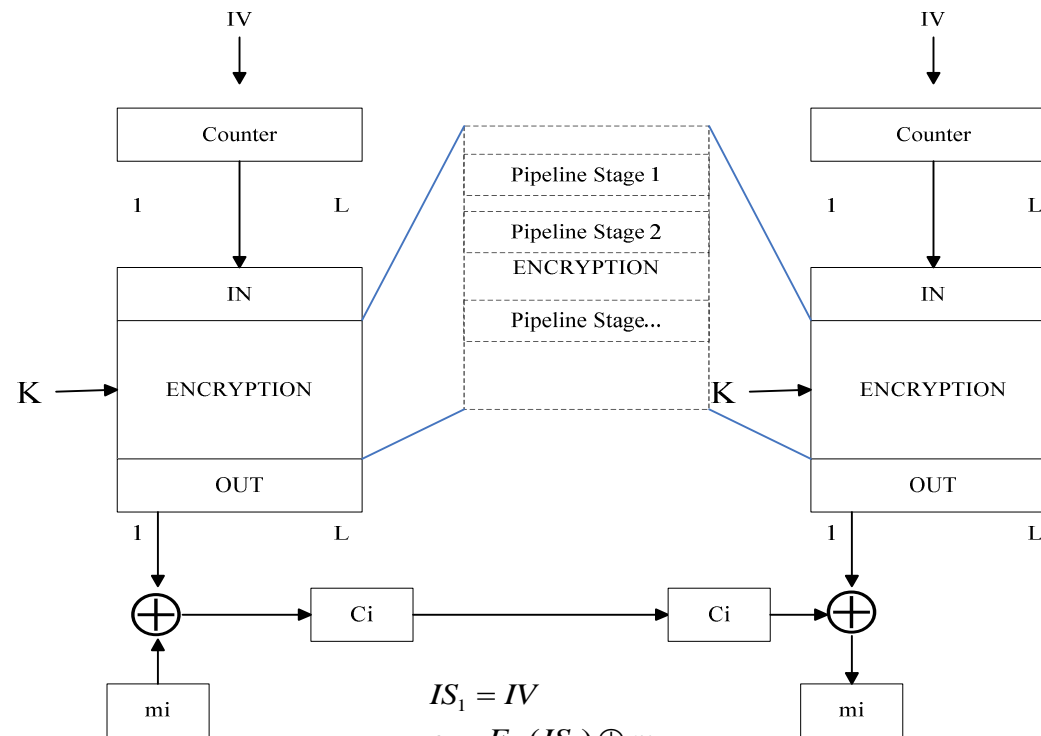


Objective

- AES for Resource Limited Environments:
 - cell phones
 - network stream media
 - wireless networks
 - mobile devices
- Compact Architectures of AES as a Stream cipher
 - Compare Different Compact Architectures
 - Consider Different Data Path Widths: 128, 64, 32, and 8 bits
- Comparison with Dedicated Hardware-Oriented Stream Ciphers Participating in the eSTREAM Contest

Mode Selection

Counter mode



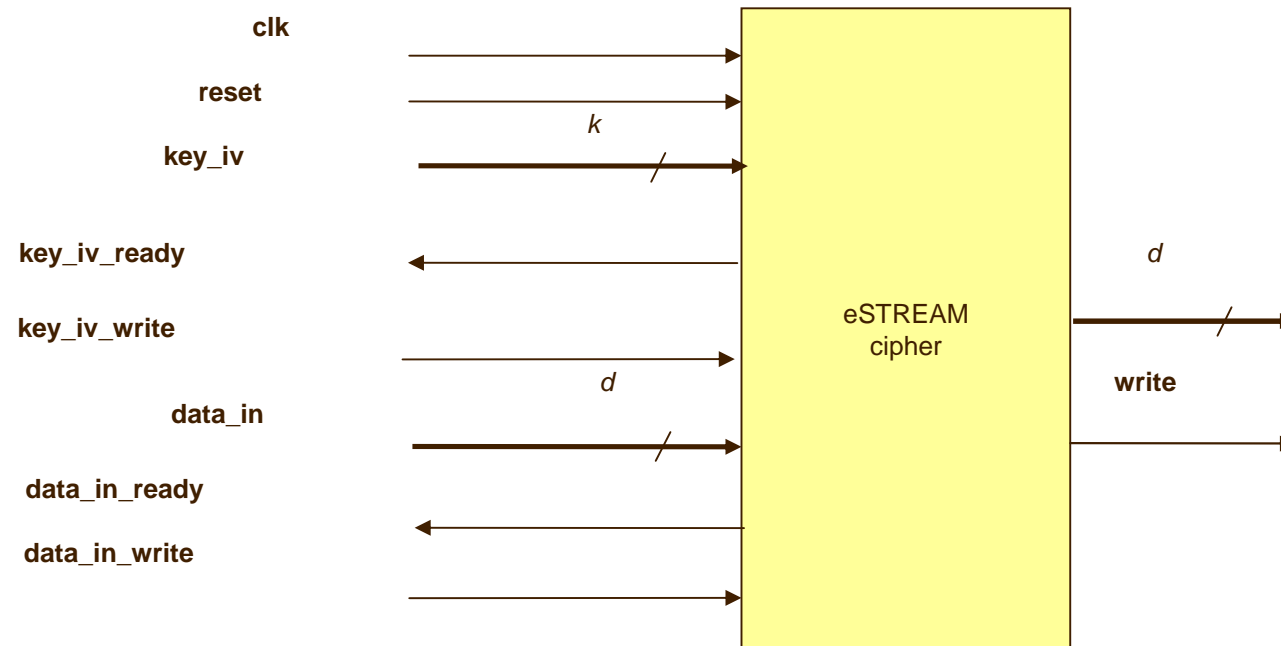
$$\begin{aligned} IS_1 &= IV \\ c_i &= E_K(IS_i) \oplus m_i \\ IS_{i+1} &= IS_i + 1 \\ m_i &= E_K(IS_i) \oplus c_i \end{aligned}$$

Why Stream Cipher?

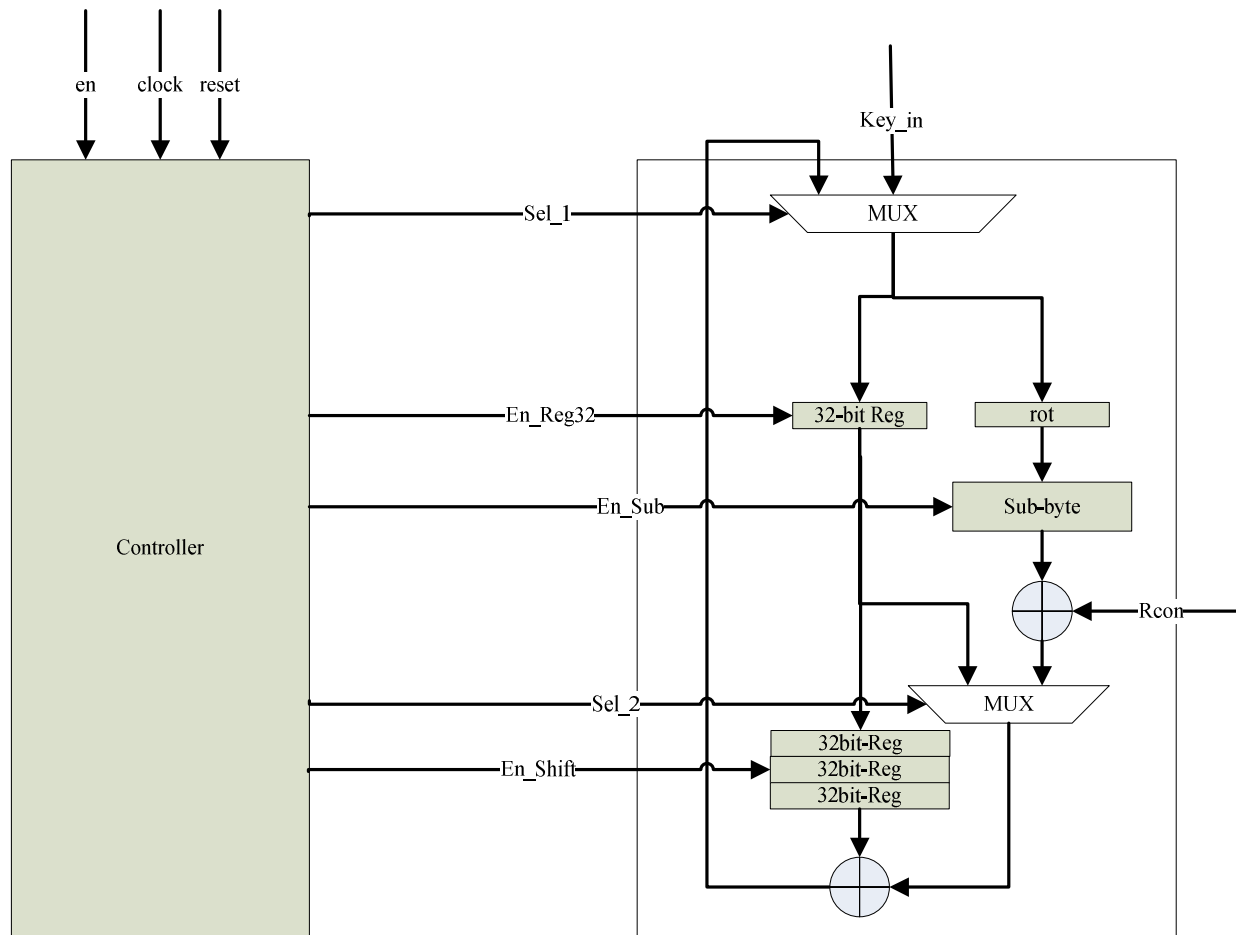
The Next State Depends on the Previous One

$$S_n = S_{n-1} + 1$$

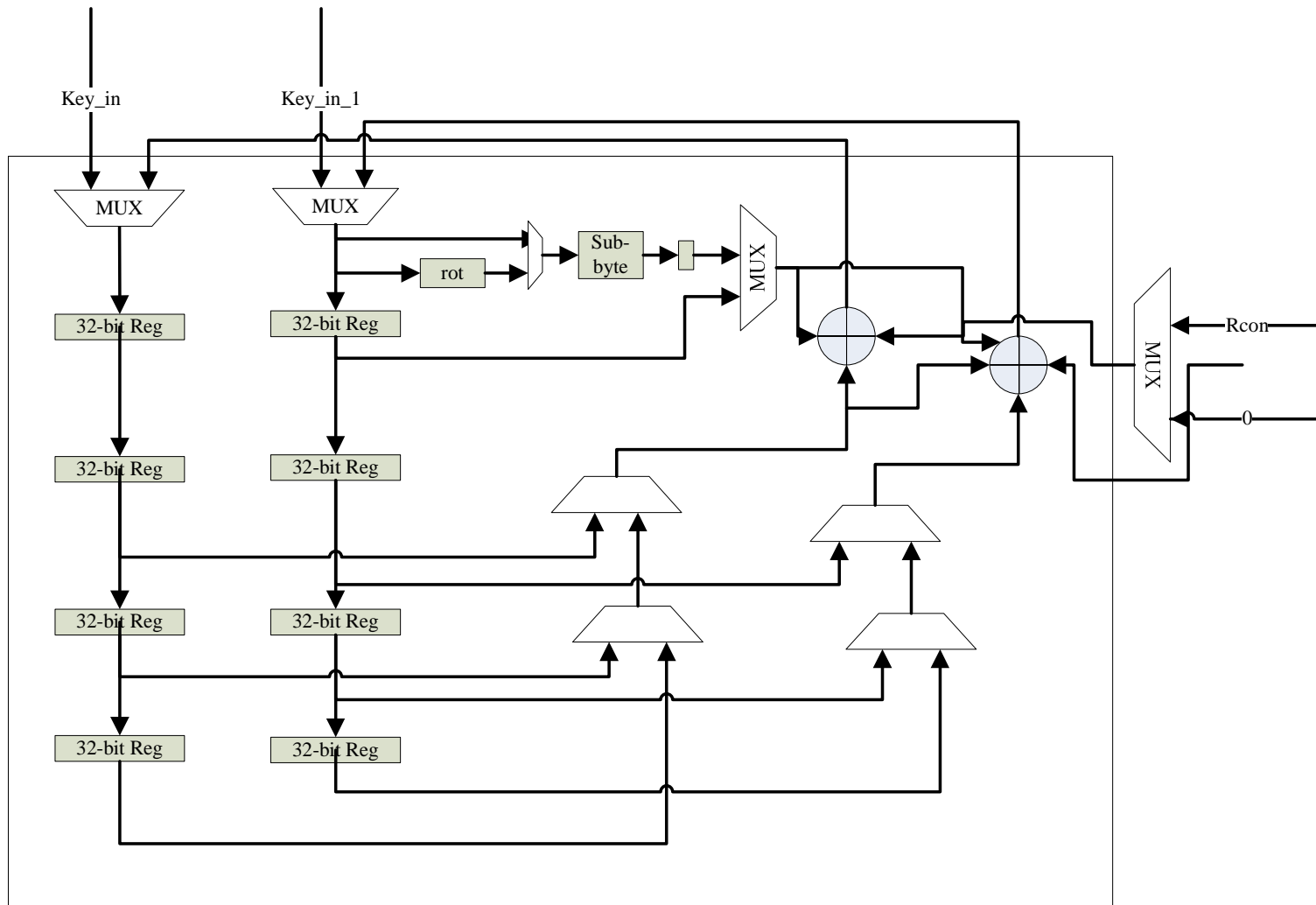
Interfaces



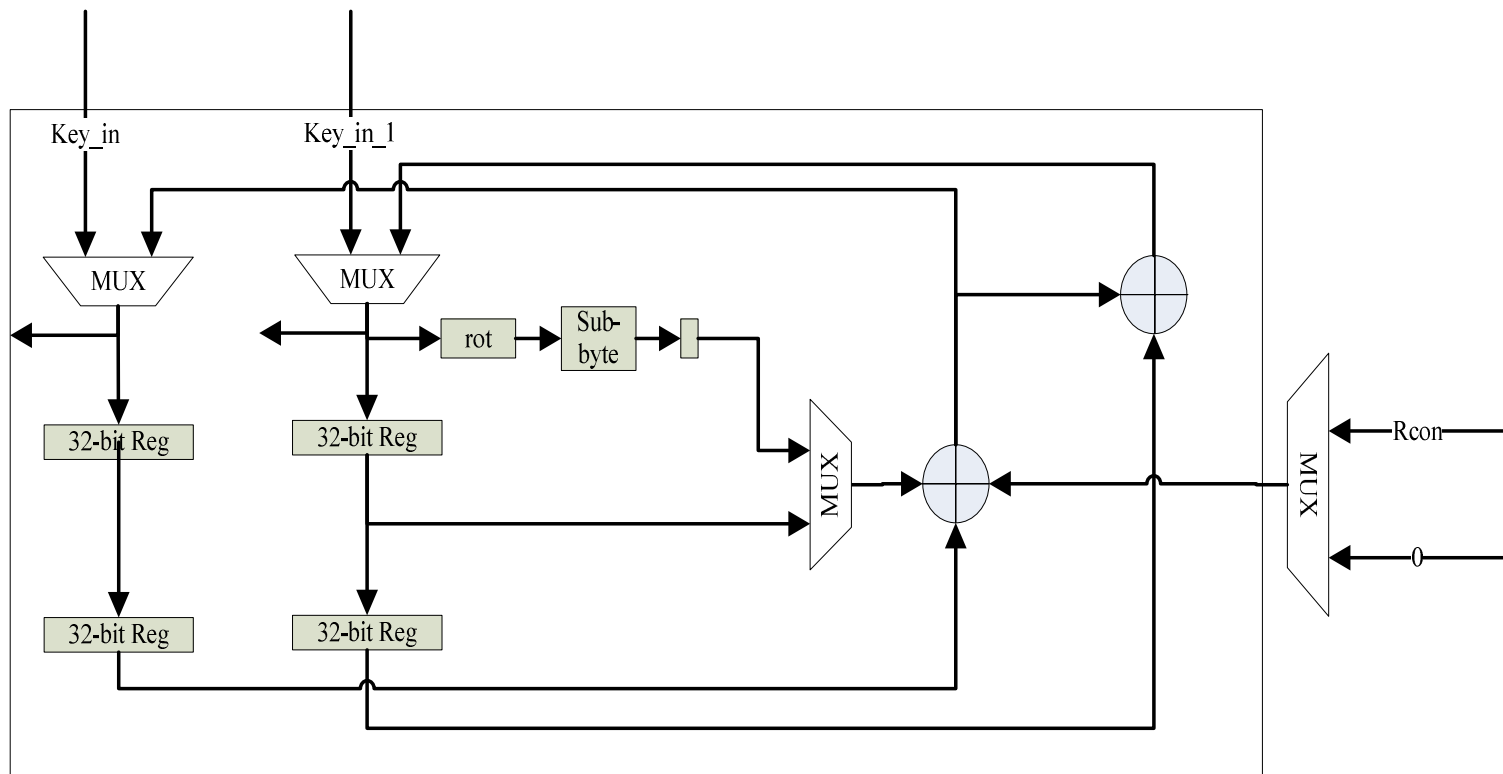
On-the-fly Key Scheduling



64-bit 3-in-1 on-the-fly key scheduling

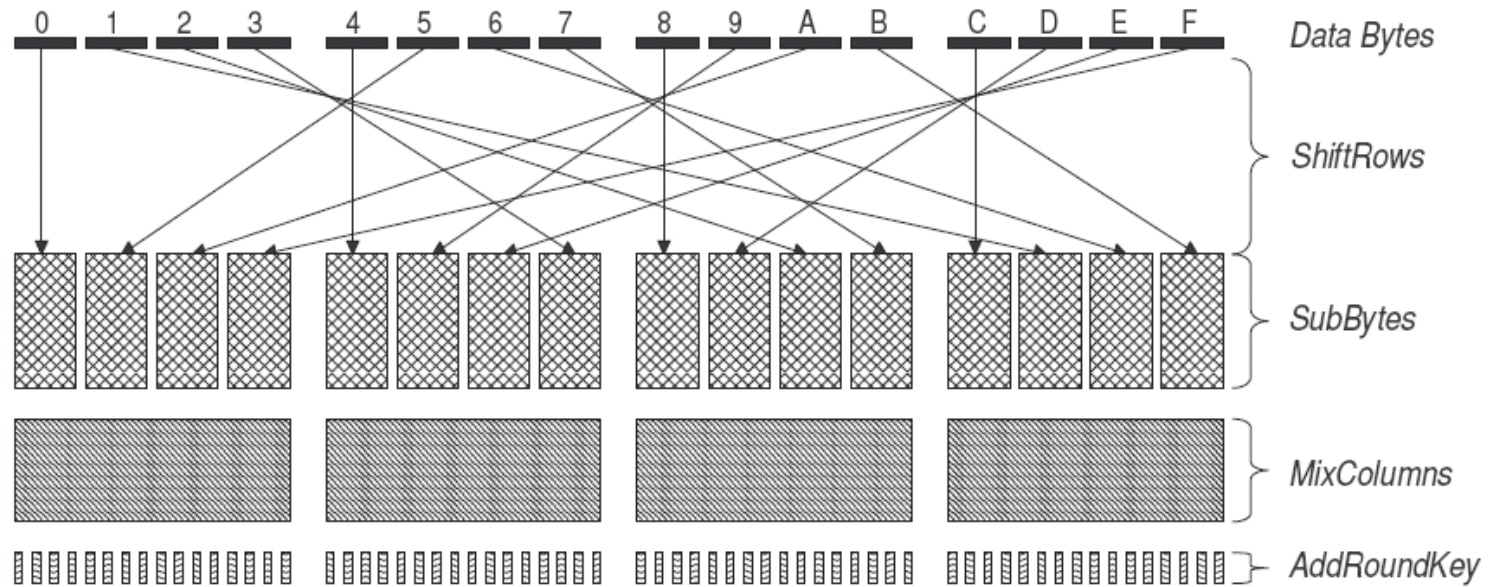


Simplified 64-bit

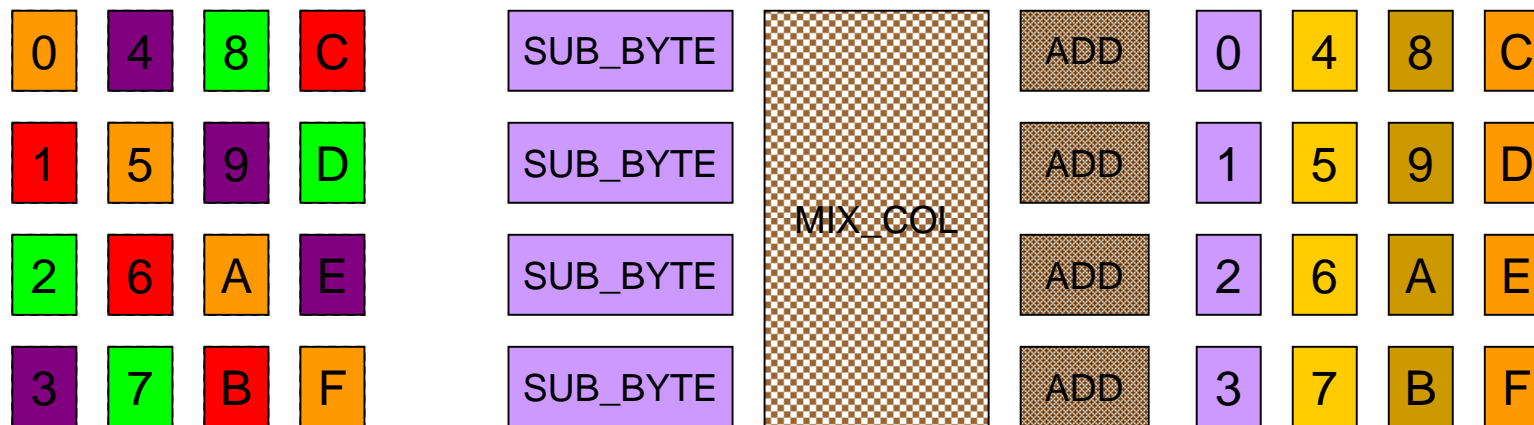


Compact Mode

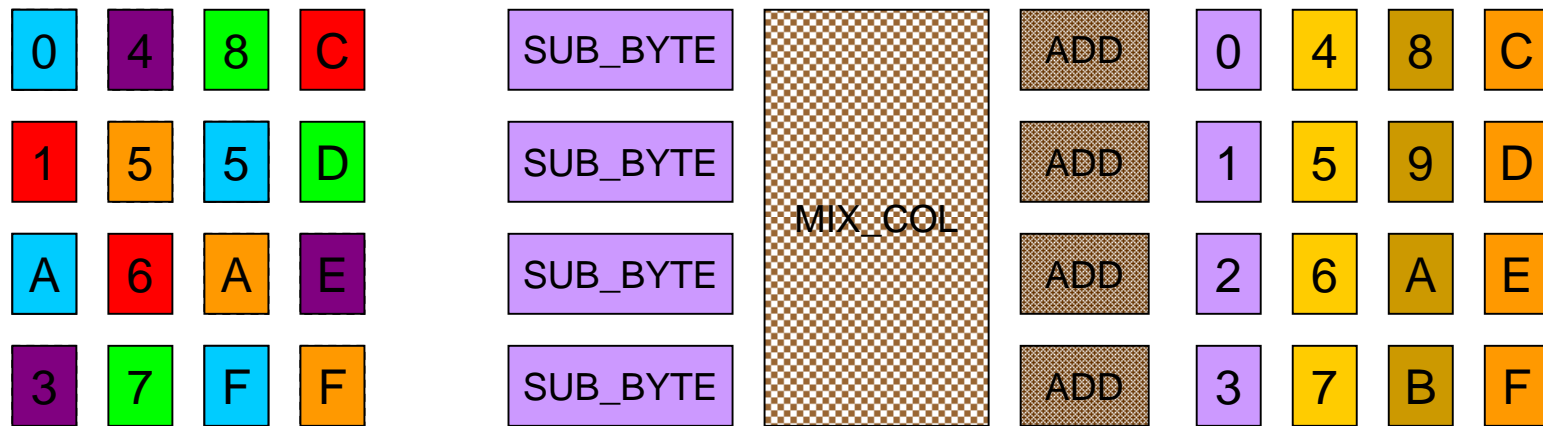
FULL MODE



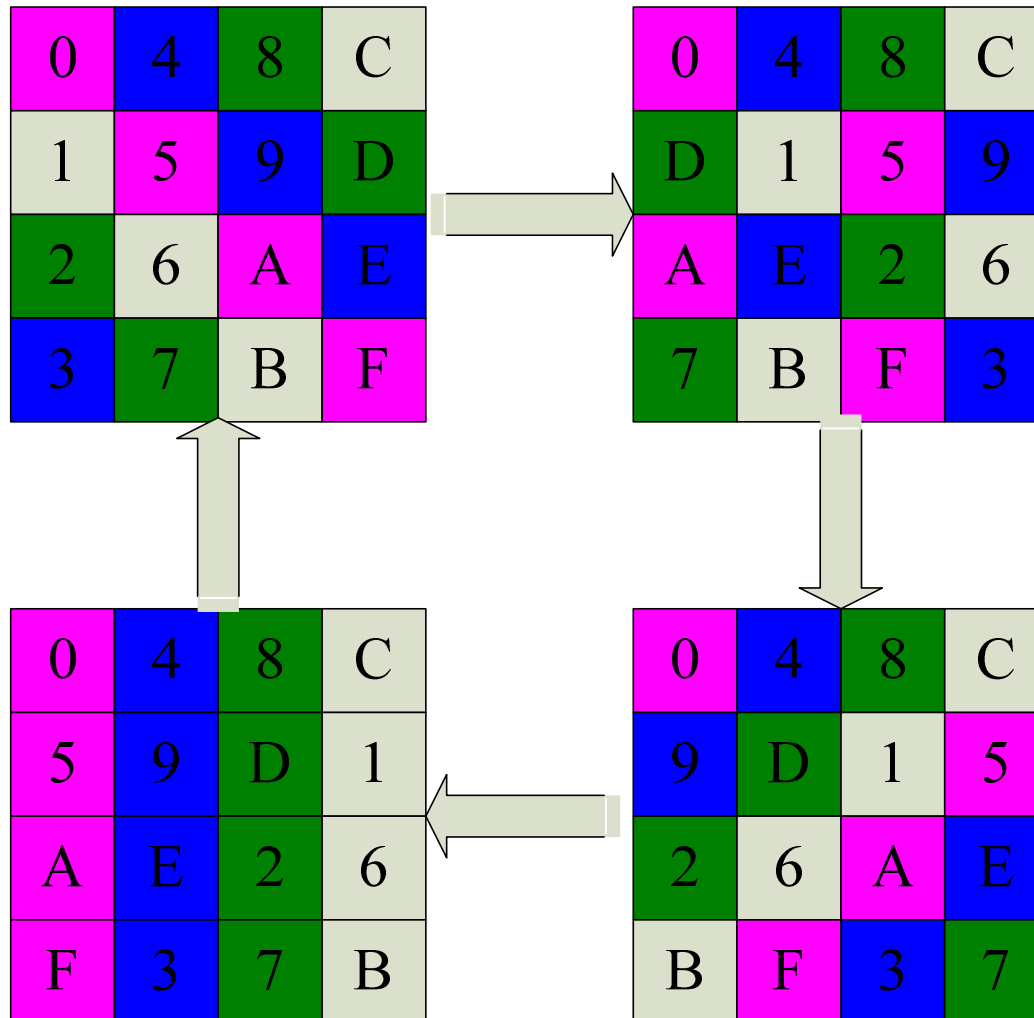
32-bit Compact Mode



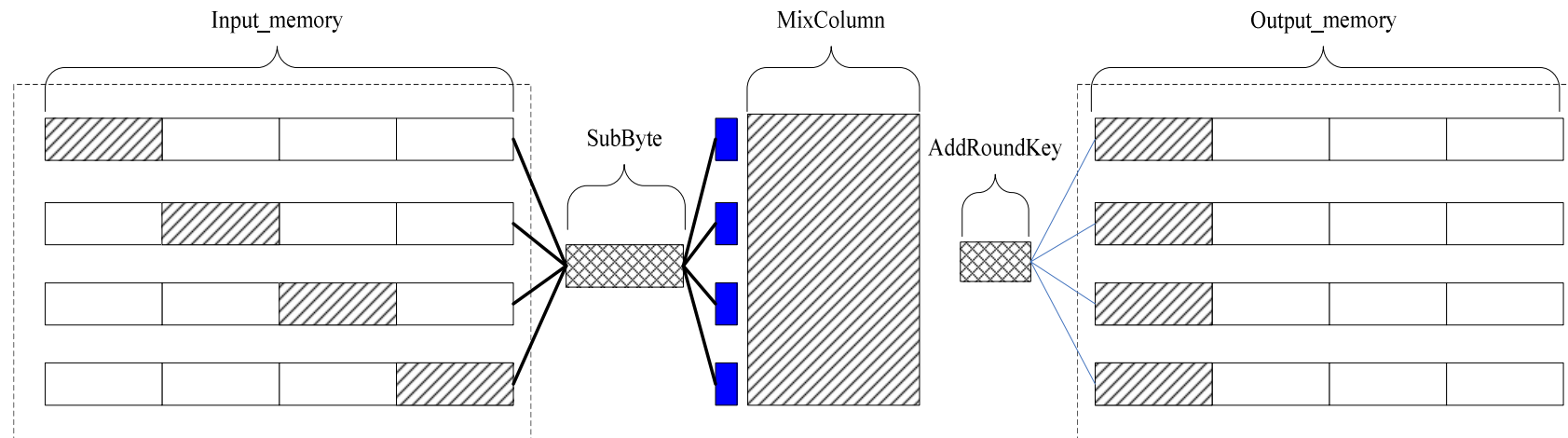
32-bit Share Memory Compact



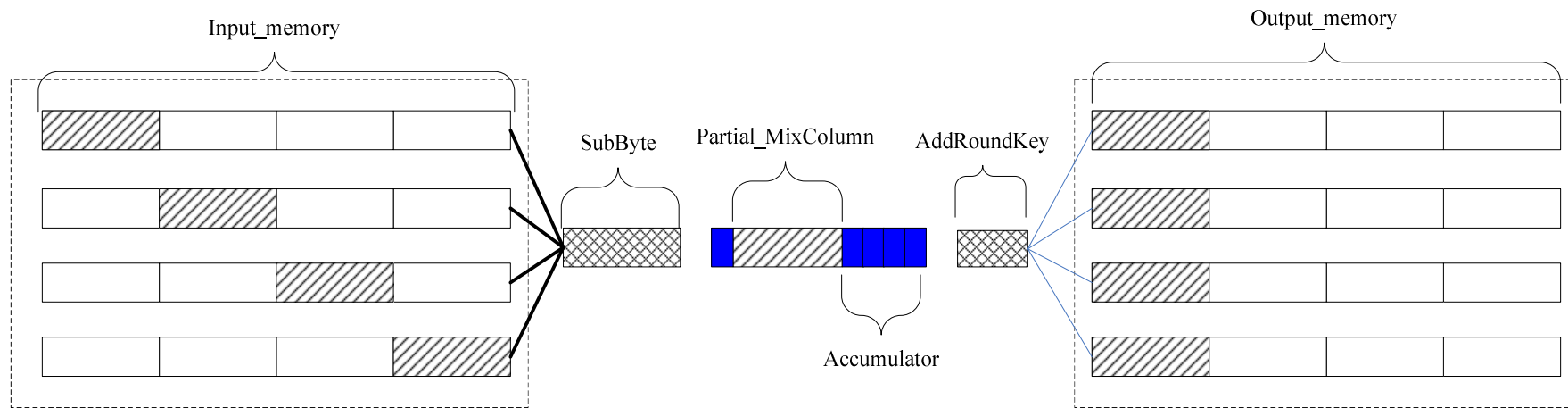
Memory Address



8-bit Compact

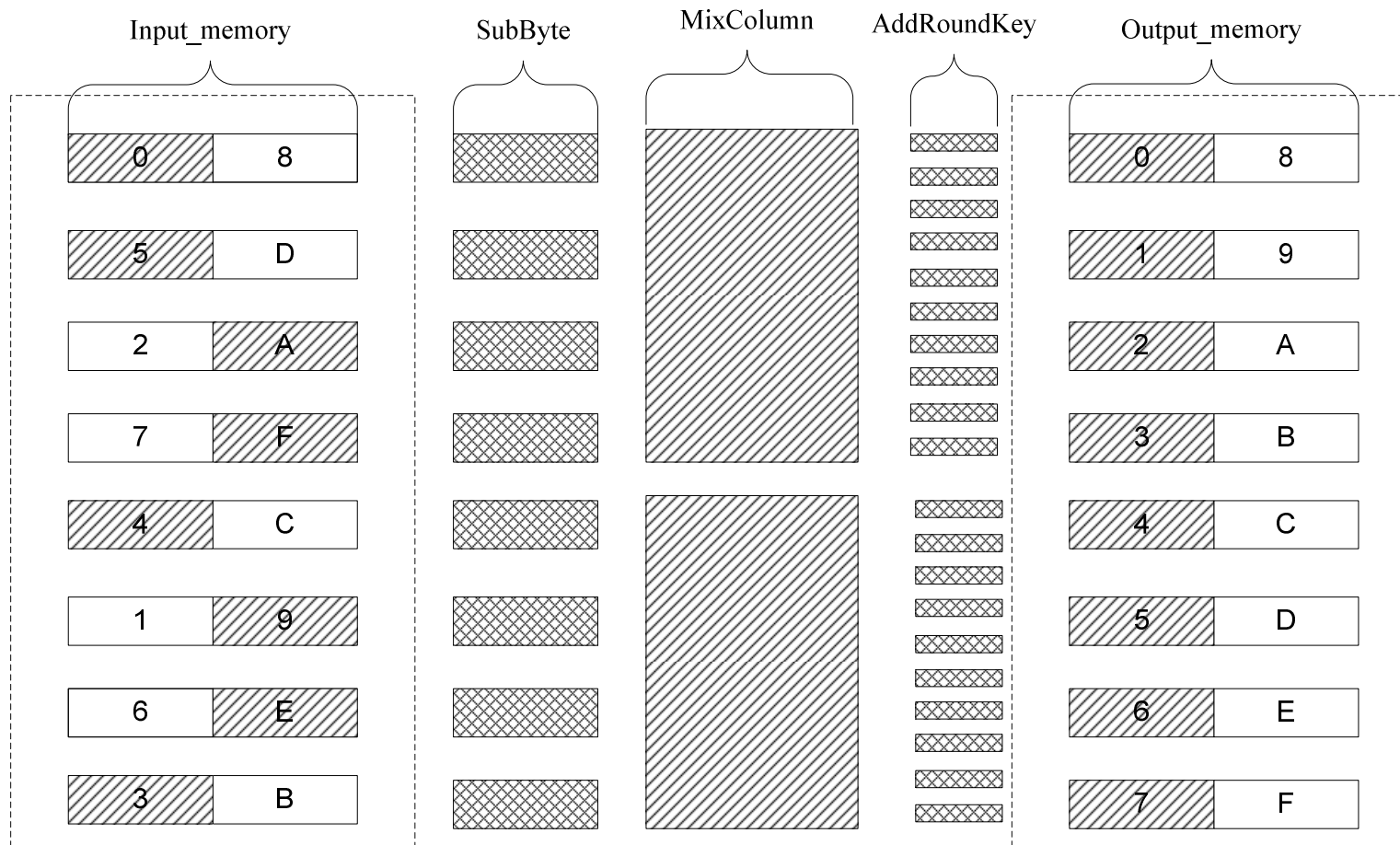


Can We Compress the MixColumn?



$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} c_0 & c_3 & c_2 & c_1 \\ c_1 & c_0 & c_3 & c_2 \\ c_2 & c_1 & c_0 & c_3 \\ c_3 & c_2 & c_1 & c_0 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

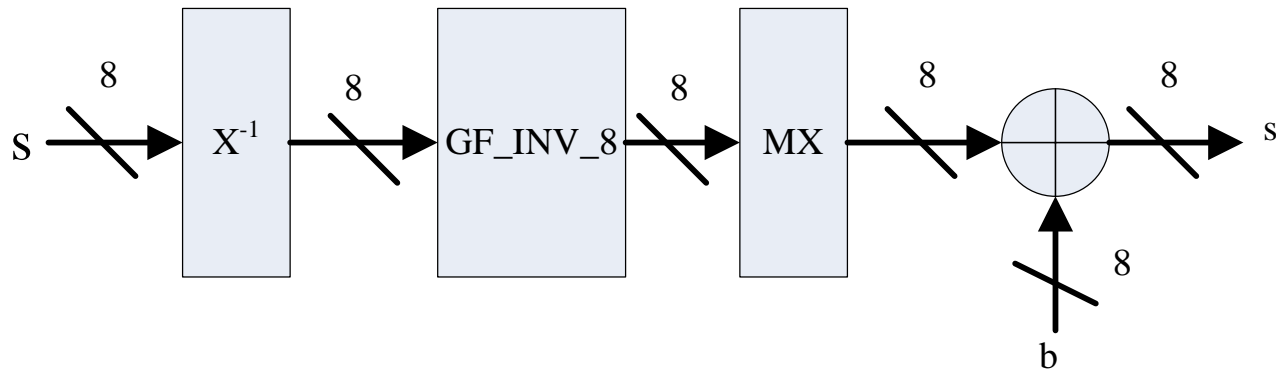
64-bit Compact Mode



Resource Usage

Arch	Memory	SubByte	MixColumn	AddRoundKey (XOR)
64-bit	256*8	8	2	64
32-bit	256*8	4	1	32
8-bit	256*8+32	1	1	8
32-bit (ShareMem)	128*8	4	1	32
64-bit (shareMem)	128*8	8	2	64
8-bit (shareMem)	128*8+32	1	1	8

S-Box Logic



- ❑ An inversion in $GF(2^8)$, GF_INV_8 , can be decomposed into a sequence of operations in $GF(2^4)$, including addition, multiplication, and inversion.
- ❑ Operations in $GF(2^4)$ can be expressed in terms of operations in $GF(2^2)$.
- ❑ Operations in $GF(2^2)$ can be easily expressed by the operations in $GF(2)$, which only consist of simple XOR gate (addition) and AND gate (multiplication). Inverse of 1 in $GF(2)$ is 1, and the inverse of 0 does not exist.
- ❑ Thus, the entire inversion in $GF(2^8)$ can be decomposed into a logic circuit composed of XOR and AND gates only.

Results

S-Box	Data path	Area (slices)	Max Clock Frequency [MHz]	Data rate (bits/clock cycle)	Max Throughput [Mbit/s]	Throughput /Area (kbit/s·slices)
logic	8	594	75.6	0.73	55.2	92.9
logic	32	728	62.5	2.91	181.2	248.9
logic	64	958	62.7	5.82	364.7	380.6
table	8	689	139.6	0.73	101.7	147.6
table	32	1347	122.5	2.91	356.4	264.6
table	64	1350	117.6	5.82	684.4	506.7

Comparison

S-Box	Data path	Area (slices)	Max Clock Frequency [MHz]	Data rate	Max Throughput [Mbit/s]
logic	8	594	75.6	0.73	55.2
logic	32	728	62.5	2.91	181.2
logic	64	958	62.7	5.82	364.7
table	8	689	139.6	0.73	101.7
table	32	1347	122.5	2.91	356.4
table	64	1350	117.6	5.82	684.4
DECIM V2		80	185	0.25	46.25
DECIM 128		89	174	0.25	43.5
Grain 128		50	196	1	196
Trivium		50	240	1	240
Moustique		278	225	1	225

Conclusions

- ❑ AES has a comparable throughput but significantly larger area than the dedicated stream ciphers
- ❑ Compact architectures are suitable for the resource limited environments, but their area is adversely affected by the controller complexity
- ❑ There are still things to improve
 - Simplify the Controller
 - Compress the Mix_Column
 - Carefully design the whole architecture....

References

1. Tim Good and Mohammed Benaissa, AES as Stream Cipher on a Small FPGA,
2. David Huang, Mark Chaney, Shashi Karanam, Nich Tom and Kris Gaj, *Comparison of FPGA-Targeted Hardware Implementations of eSTREAM Stream Cipher Candidates*, CA: Wadsworth, 2008, pp. 123–135.
3. eSTREAM Project, could be found at <http://www.ecrypt.eu.org/stream/>.
4. Kris Gaj and Pawel Chodowiec, “FPGA and ASIC Implementations of AES”, book chapter,
5. NIST, “Announcing the ADVANCED ENCRYPTION STANDARD (AES)”, Federal Information Processing Standards Publication 197, November 26, 2001
6. Network Working Group, R. Housley, “RFC 3686 - Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP)”, Internet RFC/STD/FYI/BCP Archives, January 2004.
7. Paweł Chodowiec and Kris Gaj, “Very Compact FPGA Implementation of the AES Algorithm”, Cryptographic Hardware and Embedded Systems -- CHES 2003: 5th.
8. H. Lipmaa, P. Rogaway, and D. Wagner, “CTR-Mode Encryption, Comments to NIST concerning AES Modes of Operations”, Symmetric Key Block Cipher Modes of Operation Workshop, 2000.
9. Kris. Gaj, “AES implementations in software and hardware”, ECE746 Lecture, GMU.

Thanks

- Thanks Dr. David Hwang and Dr. Kris Gaj
- Q&A?