SCIENCE
PASSION
TECHNOLOGY

Digital System Design

Design case study for AES

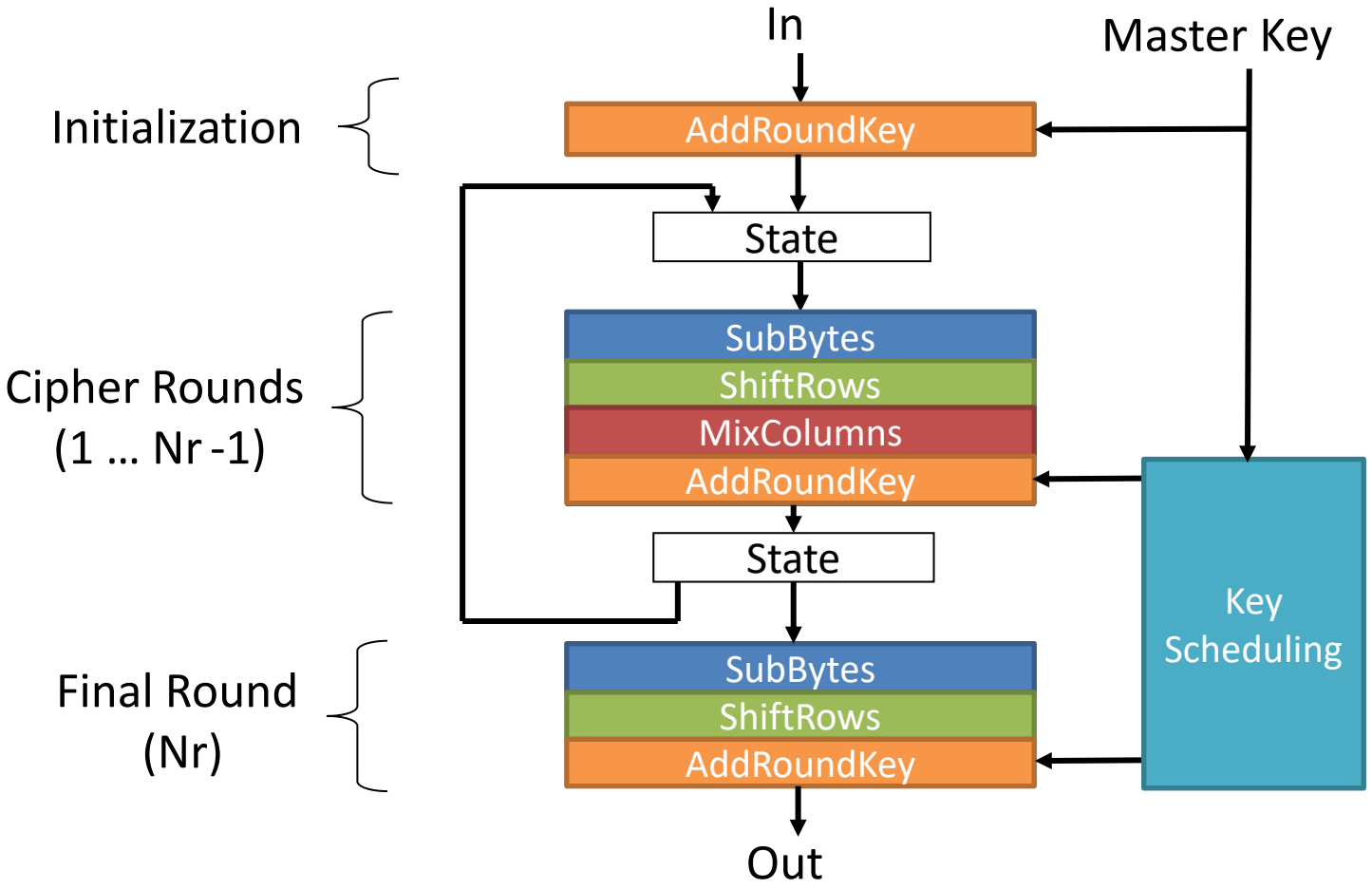
March, 2025

Sujoy Sinha Roy
sujoy.sinharoy@tugraz.at
Graz University of Technology

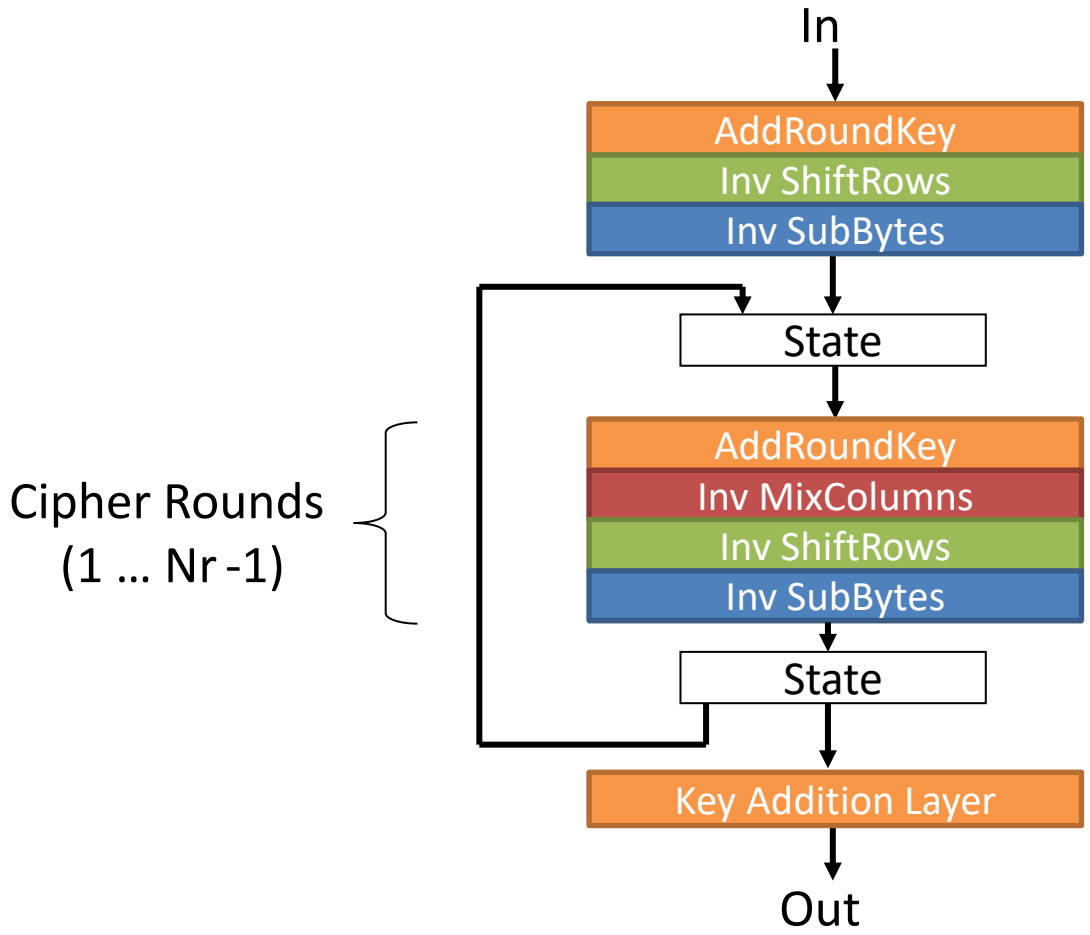
https://www.isec.tugraz.at/course/digital-system-design-705044-ss-2025/

Acknowledgement:
The slides are borrowed from an early version of
“Cryptography on Hardware Platforms” lecture by
Ahmet Can Mert

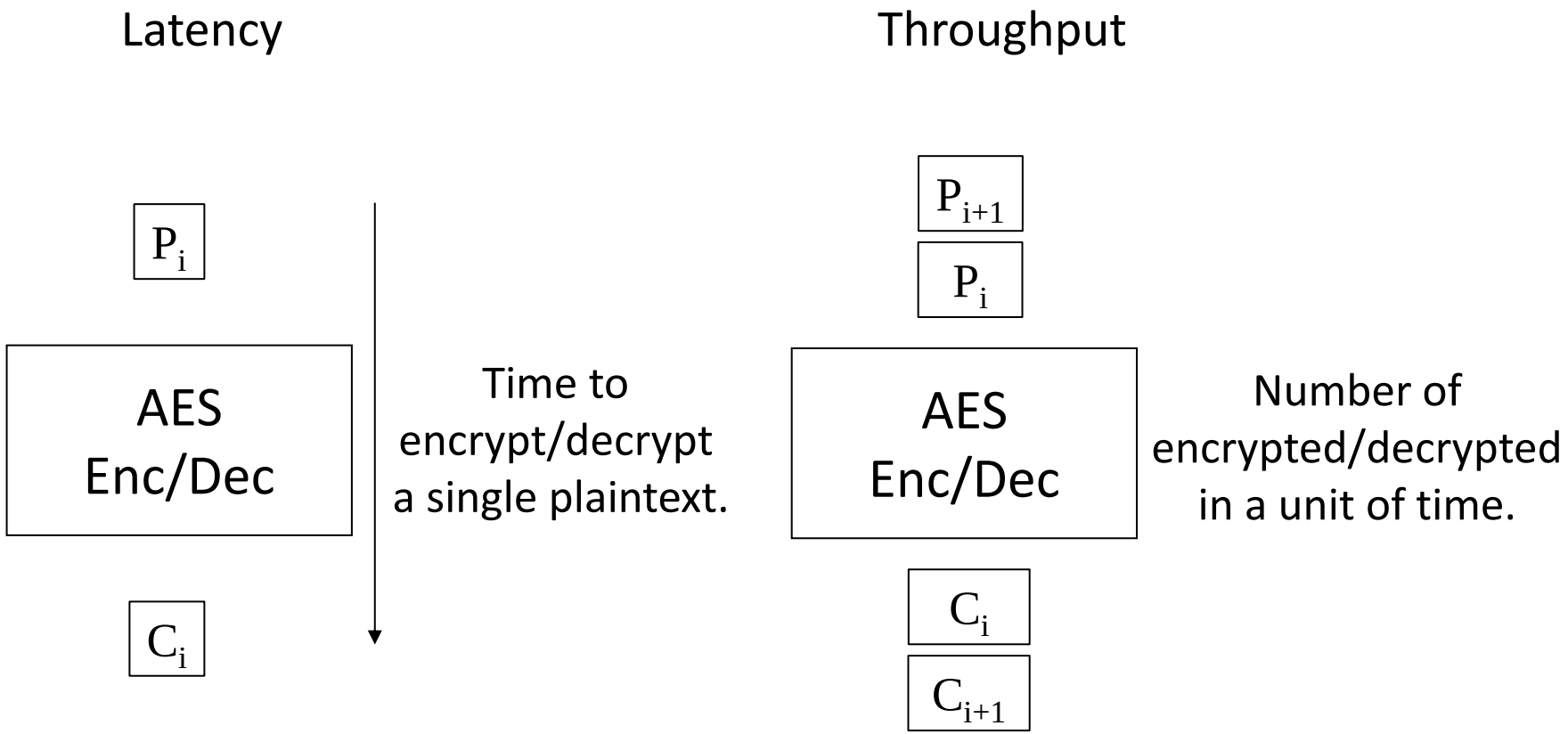
AES Encryption



AES Decryption



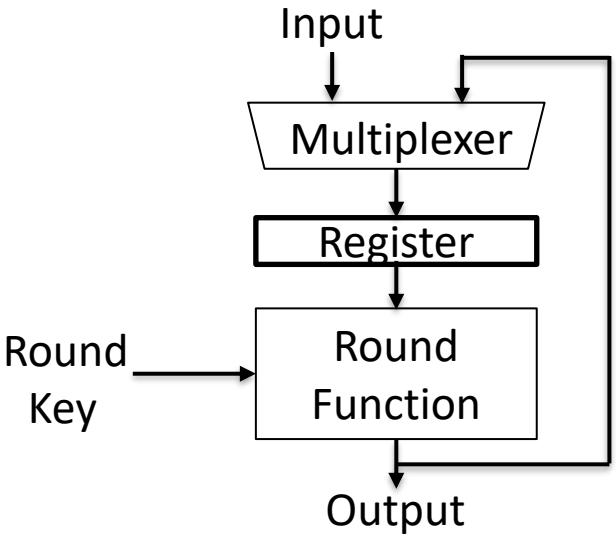
Latency vs Throughput



Block Cipher Implementations: Iterative Approach

Implement the combinational logic required for one round (supplemented with register and multiplexers). Then, use it repeatedly.

- Only one block of data is encrypted at a time.
- The number of clock cycles necessary to encrypt a single block of data is equal to the number of cipher rounds.



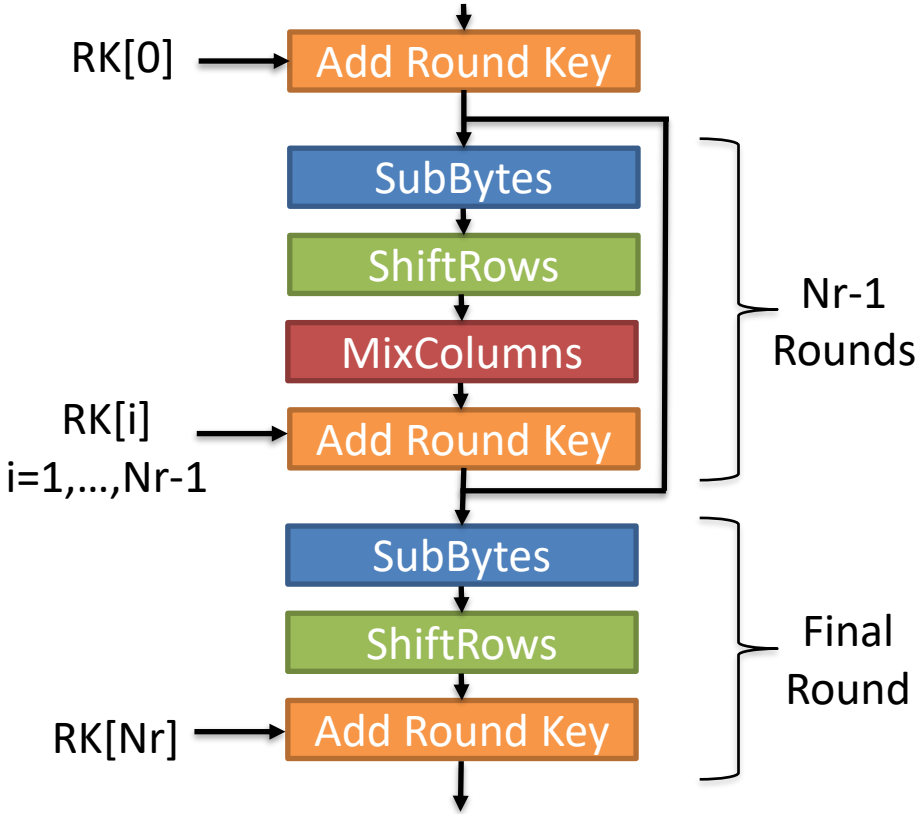
Clock period (t_{clk}) = t

Latency ≈ t *Nr

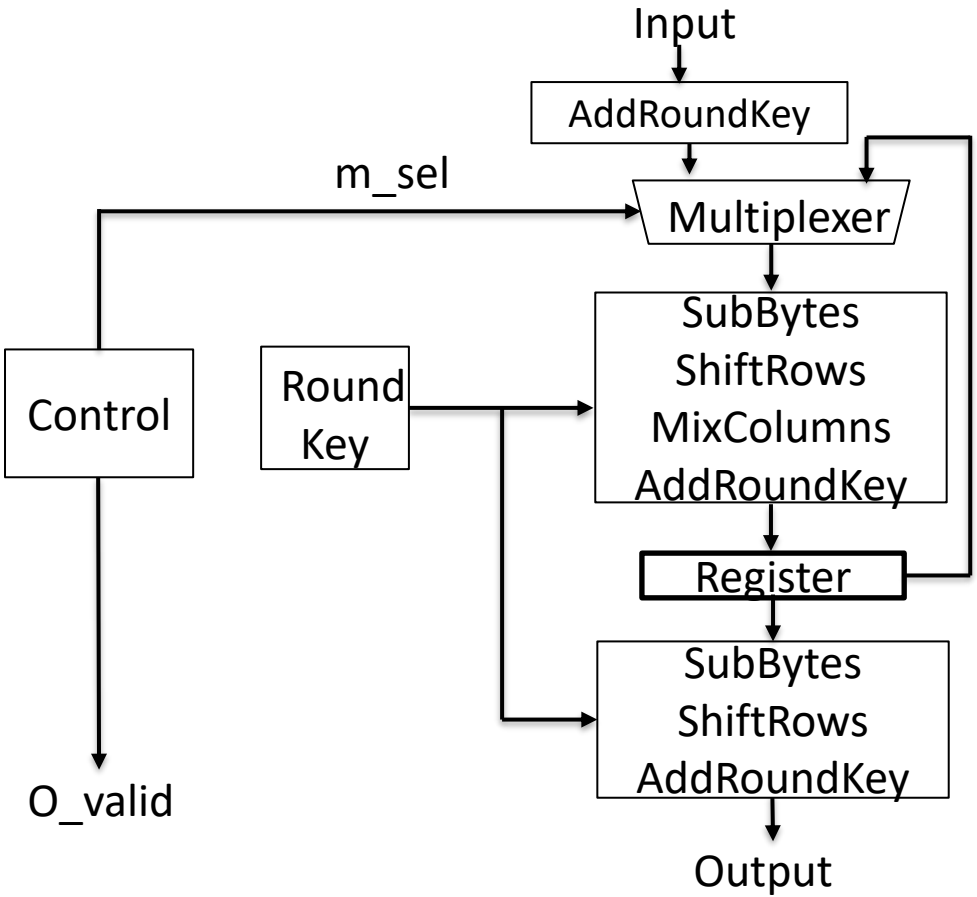
Throughput ≈ 1 / (t*Nr)

AES Implementations: Iterative Approach

- Initialization
- Round (repeated Nr-1 times):
  - SubBytes
  - ShiftRows
  - MixColumns
  - AddRoundKey
- Final Round
  - SubBytes
  - ShiftRows
  - Add Round Key



AES Implementations: Iterative Approach



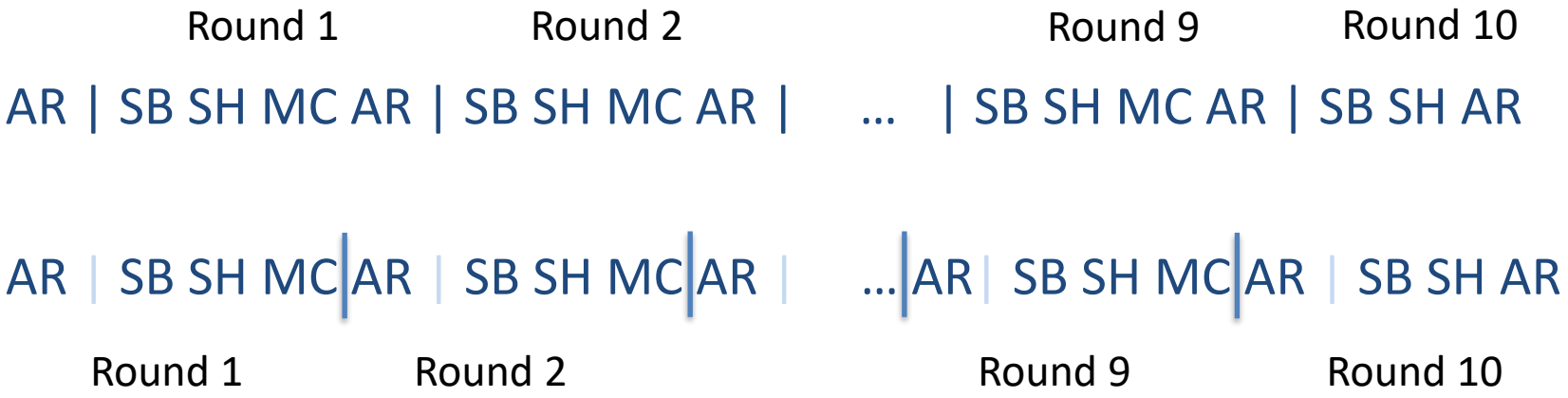
AES Implementations: Iterative Approach

- SubBytes and AddRoundKey are instantiated twice.
  - Can we do better?

10 Rounds of AES

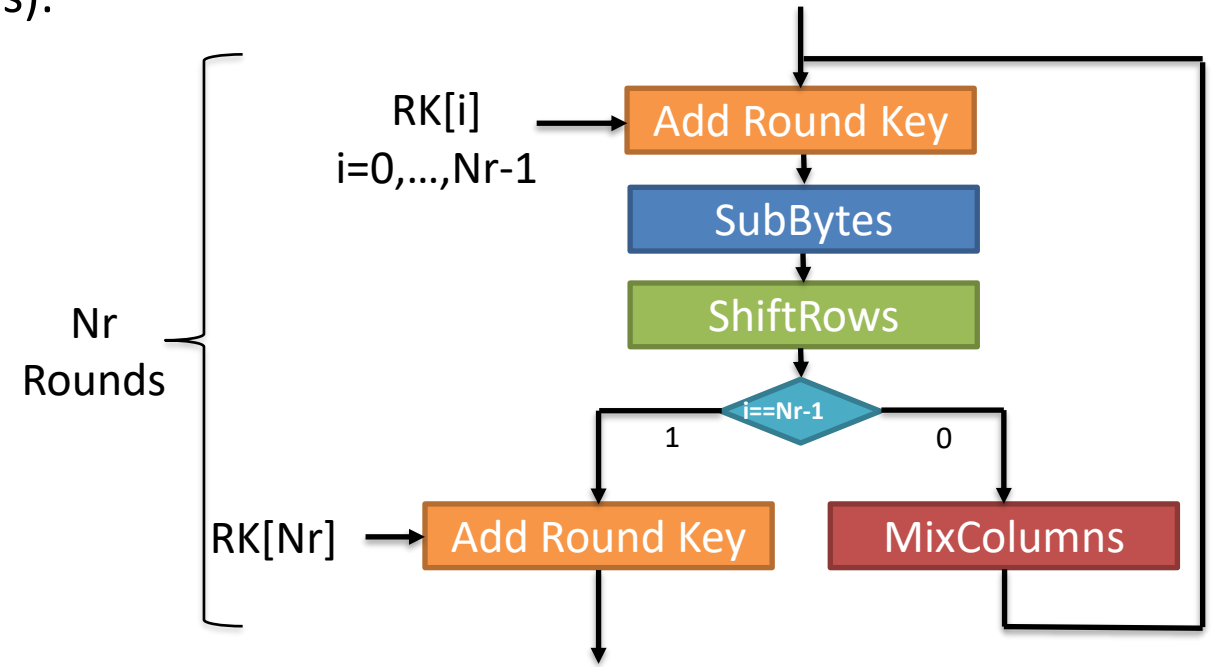
Round 1	Round 2		Round 9	Round 10
AR | SB SH MC AR |	SB SH MC AR |	…	| SB SH MC AR |	SB SH AR

10 Rounds of AES … grouped to increase synergies among rounds



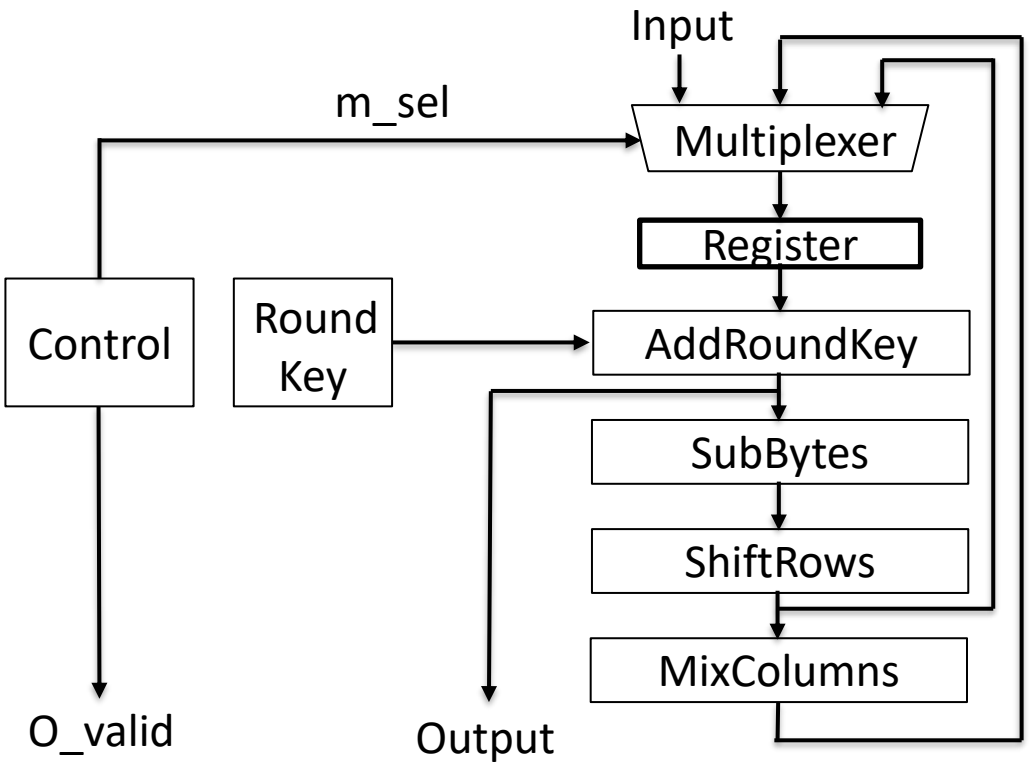
AES Implementations: Iterative Approach

- Round (repeated Nr times):
  - AddRoundKey
  - SubBytes
  - ShiftRows
  - MixColumns
    or
    AddRoundKey



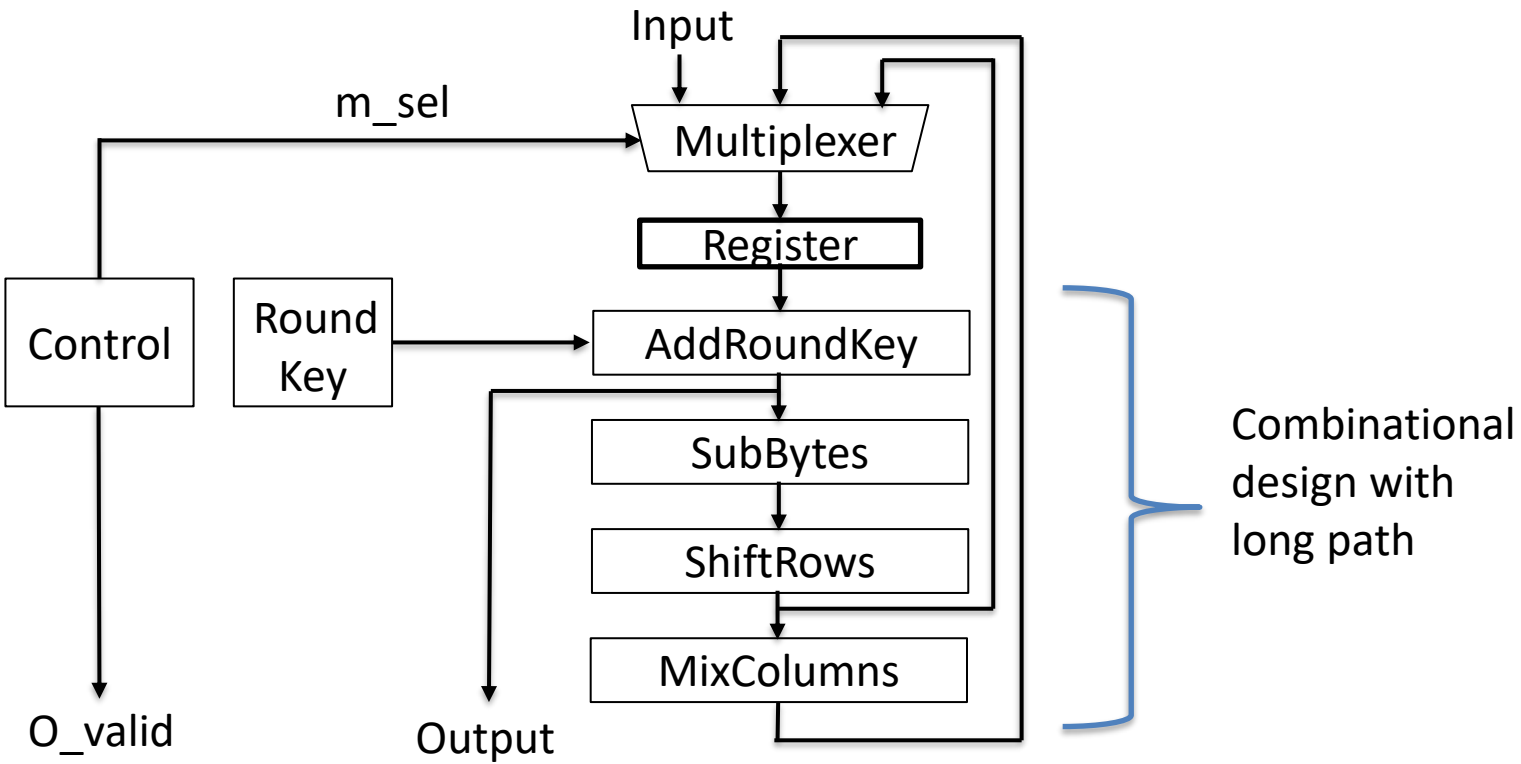
AES Implementations: Iterative Approach

- High-level diagram of the architecture



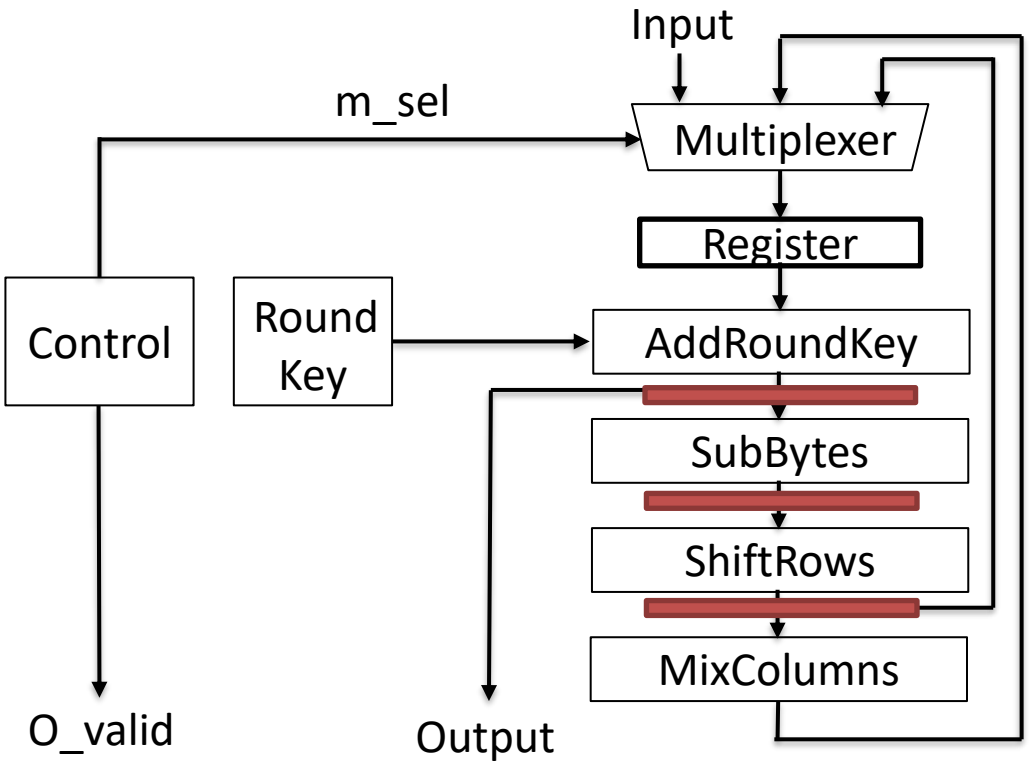
AES Implementations: Iterative Approach

- High-level diagram of the architecture



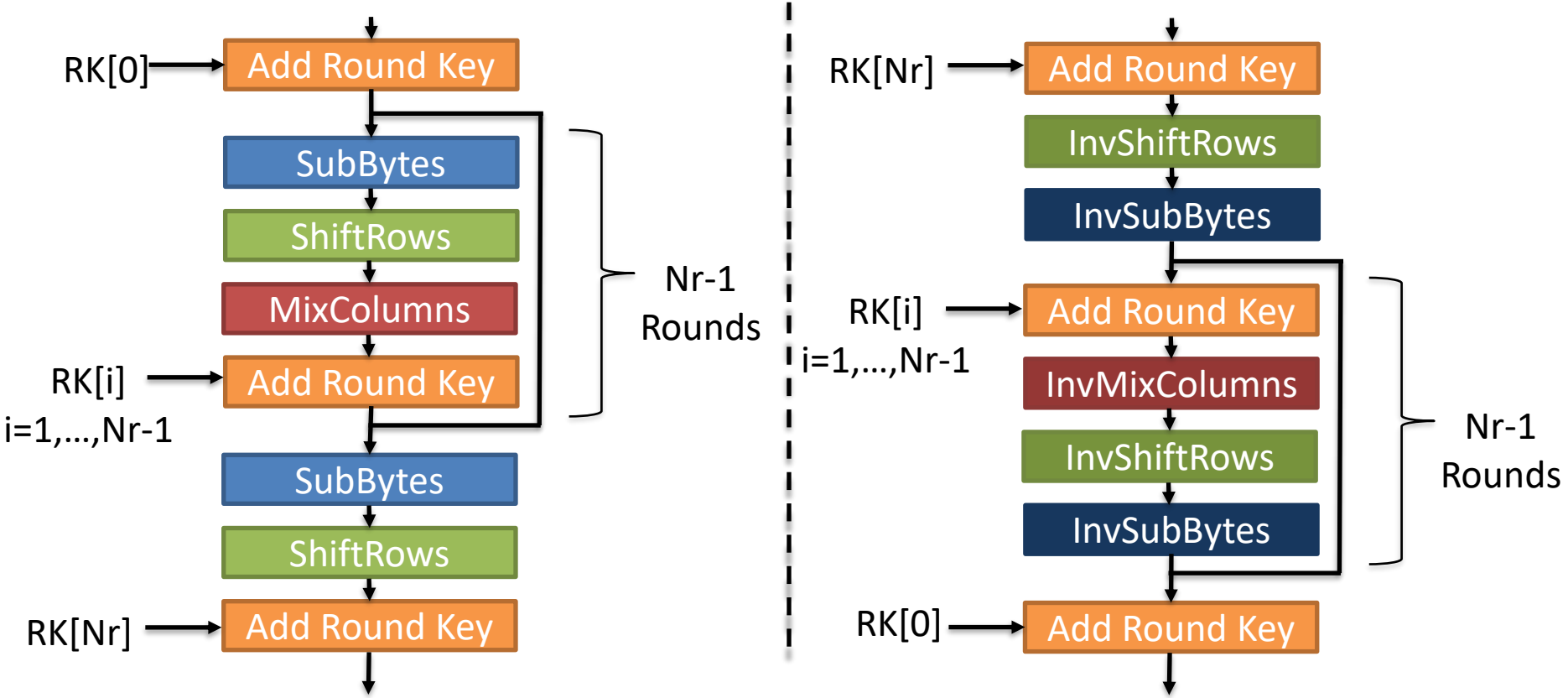
AES Implementations: Iterative Approach

- High-level diagram of the architecture
  - What happens if we divide a round into multiple stages?



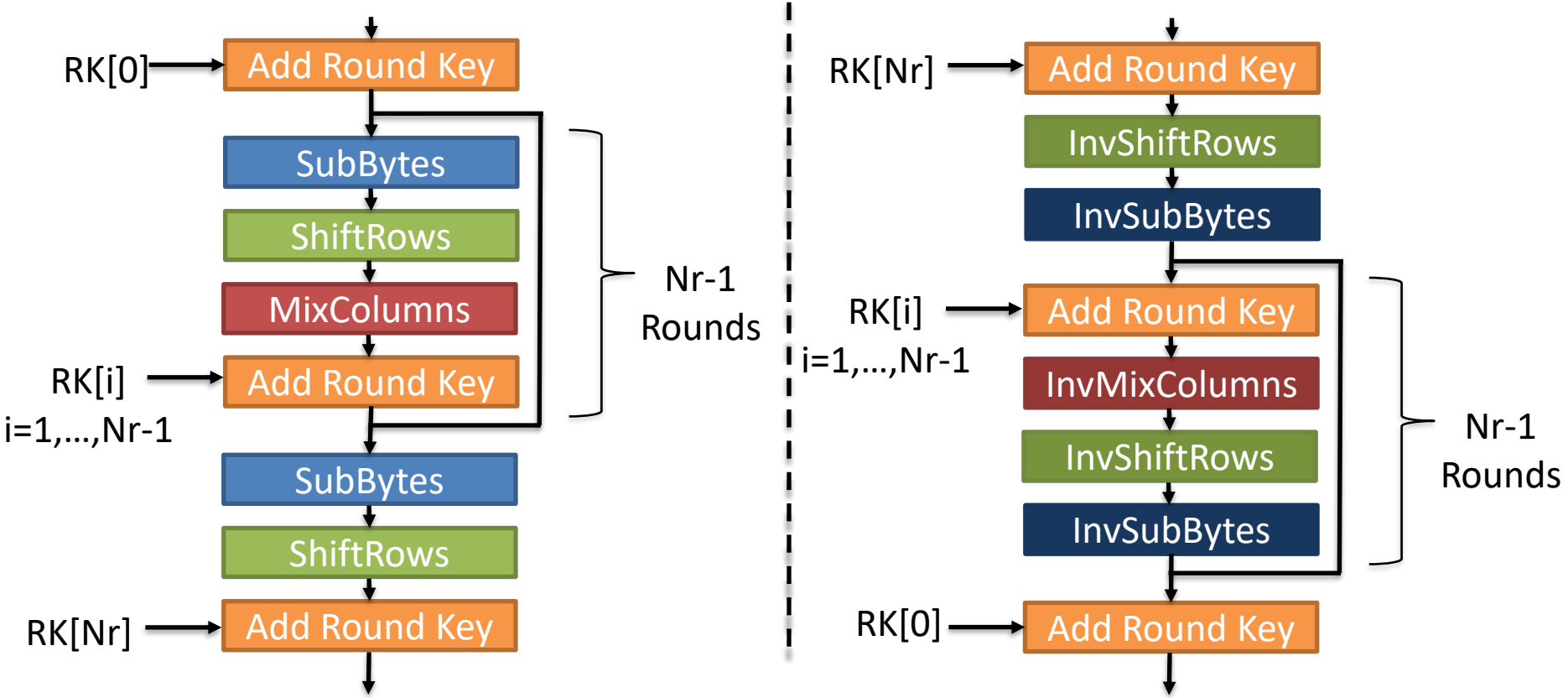
AES Implementations: Hardware

- What about decryption?



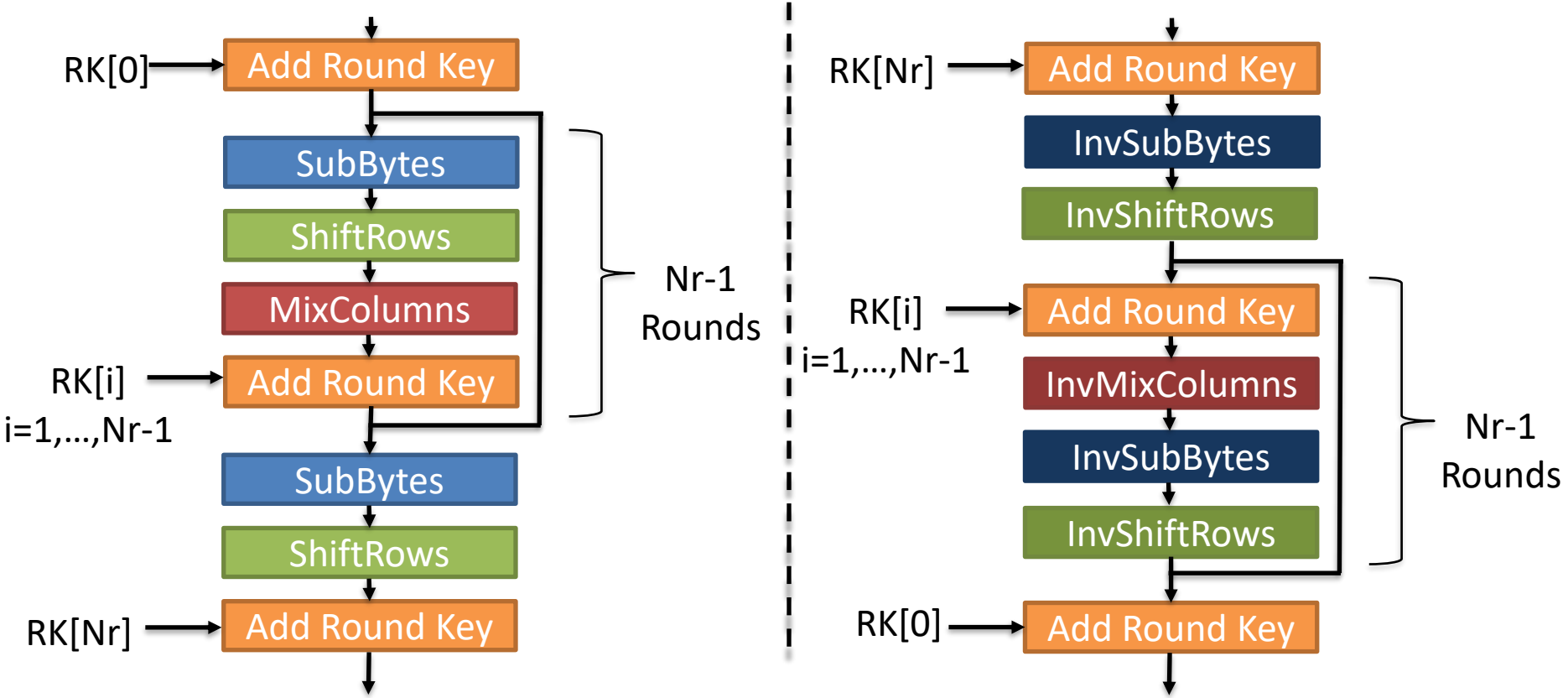
AES Implementations: Hardware

- Can we make Enc. and Dec. look similar?



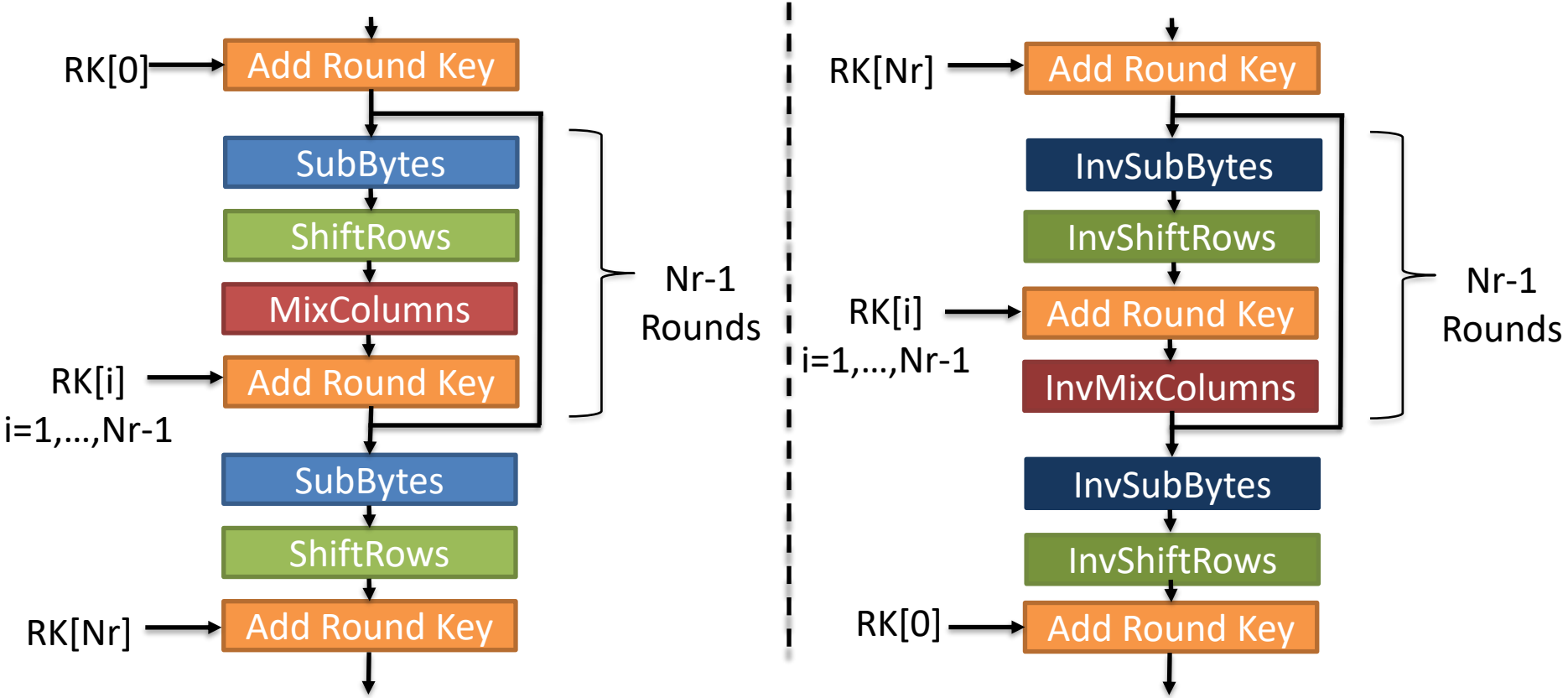
AES Implementations: Hardware

- Swap InvShiftRows and InvSubBytes



AES Implementations: Hardware

- Push InvShiftRows and InvSubBytes down

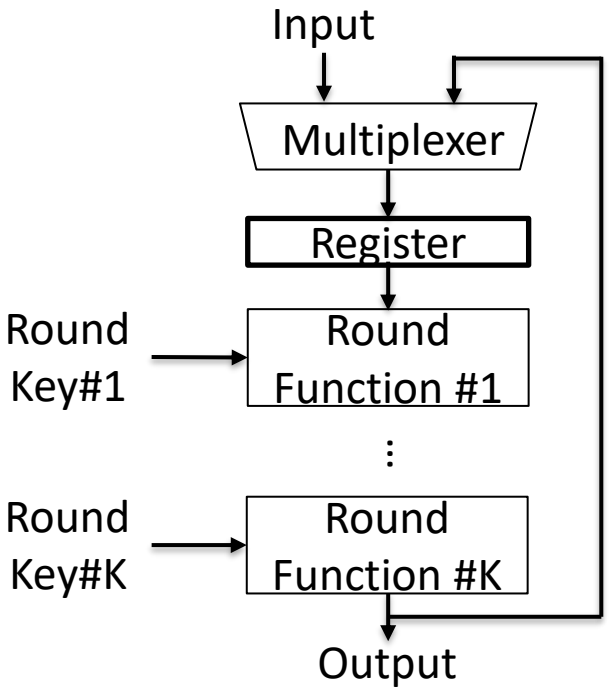


Note:
Reorganization of the internal steps is specific to AES

Applicability to other ciphers may or may not be possible

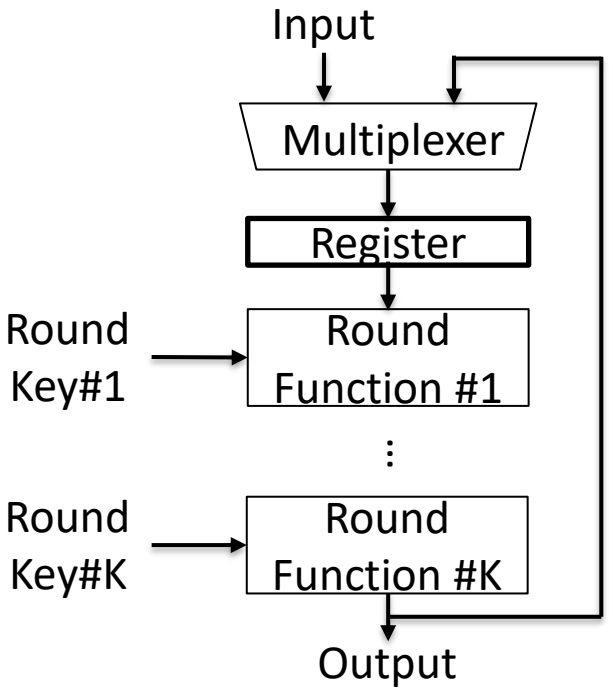
Block Cipher Implementations: Partial Loop Unrolling

- *K* round out of *Nr* round functions are implemented in combinational part.
  - Partial loop unrolling



Block Cipher Implementations: Partial Loop Unrolling

- *K* round out of *Nr* round functions are implemented in combinational part.
  - Partial loop unrolling



Clock period (t_{clk}) ≈ $K * t$

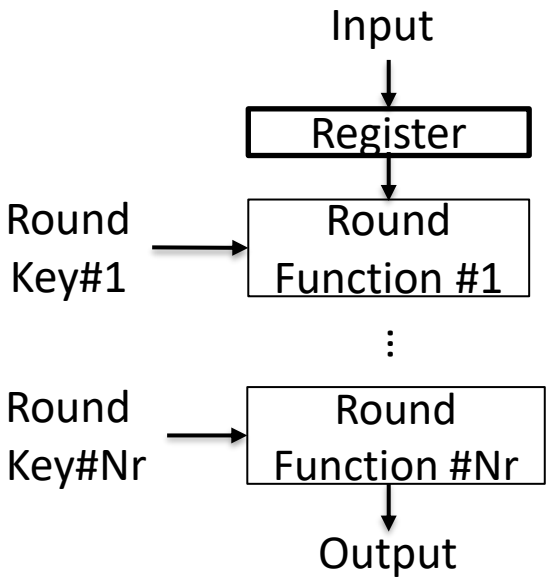
Latency ≈ t * (# of rounds)

Throughput ≈ 1 / (t * (# of rounds))

Without pipelining, unrolling offers no throughput improvement.

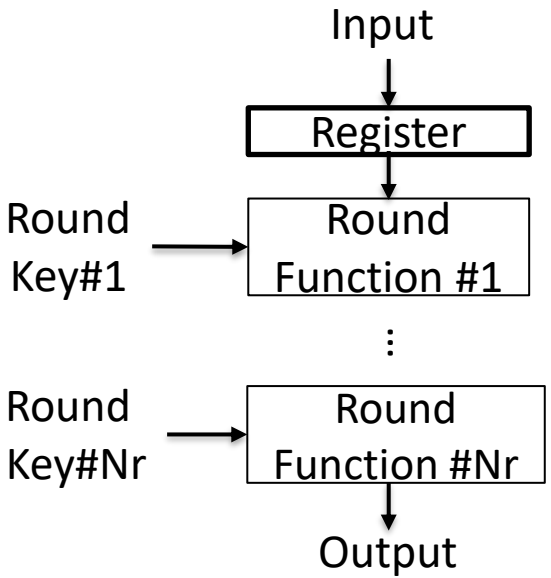
Block Cipher Implementations: Loop Unrolling

- All round functions are implemented in combinational part.
  - Full loop unrolling



Block Cipher Implementations: Loop Unrolling

- All round functions are implemented in combinational part
  - Full loop unrolling



Clock period (t_{clk}) ≈ (# of rounds) * t

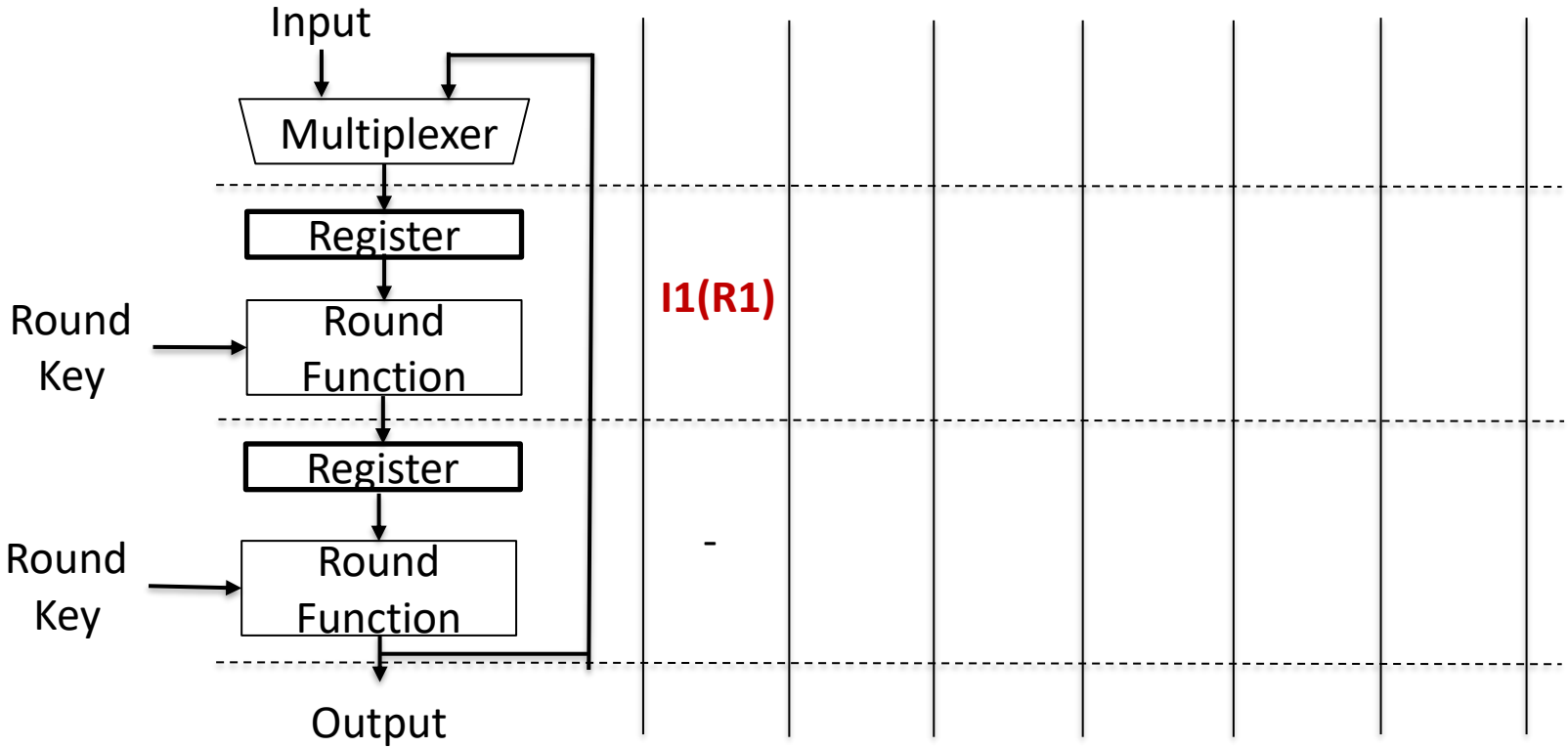
Latency ≈ t * (# of rounds)

Throughput ≈ 1 / (t * (# of rounds))

- Without pipelining, unrolling offers no throughput improvement.

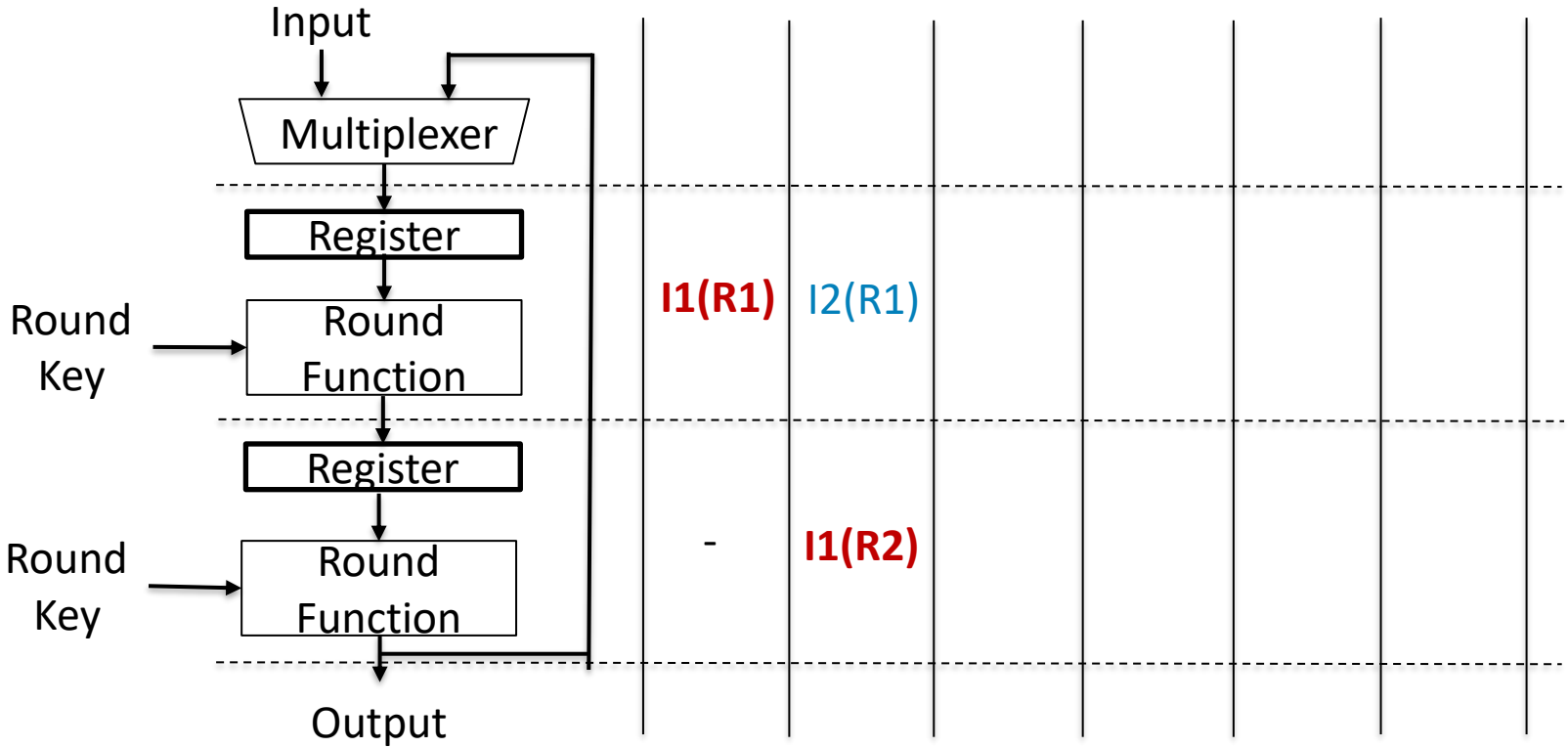
Block Cipher Implementations: Pipelining

- A traditional methodology for design of high-performance implementations.
  - Partial or full outer-loop pipelining (i.e., *K*=2 with *Nr*=4 rounds)



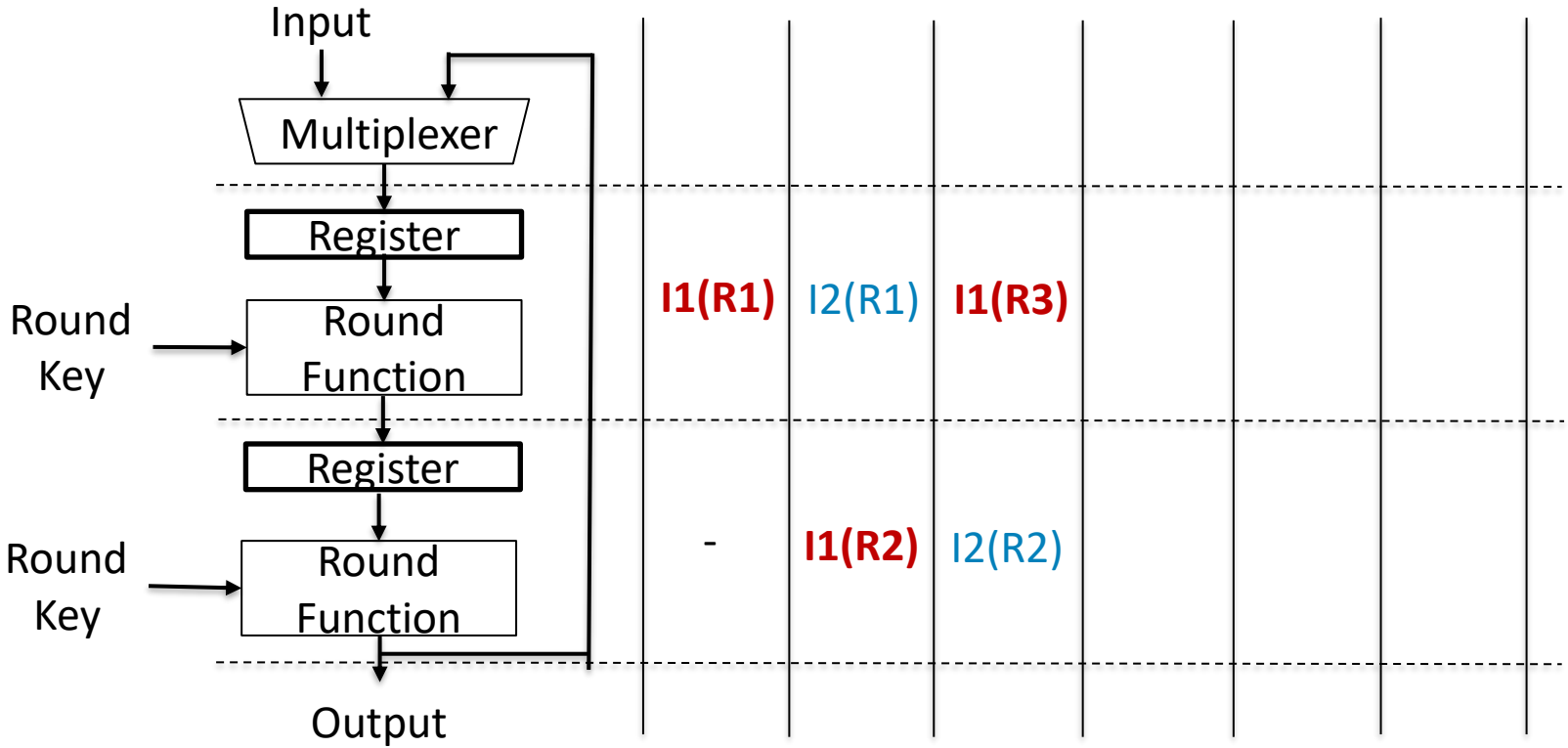
Block Cipher Implementations: Pipelining

- A traditional methodology for design of high-performance implementations.
  - Partial or full outer-loop pipelining (i.e., *K*=2 with *Nr*=4 rounds)



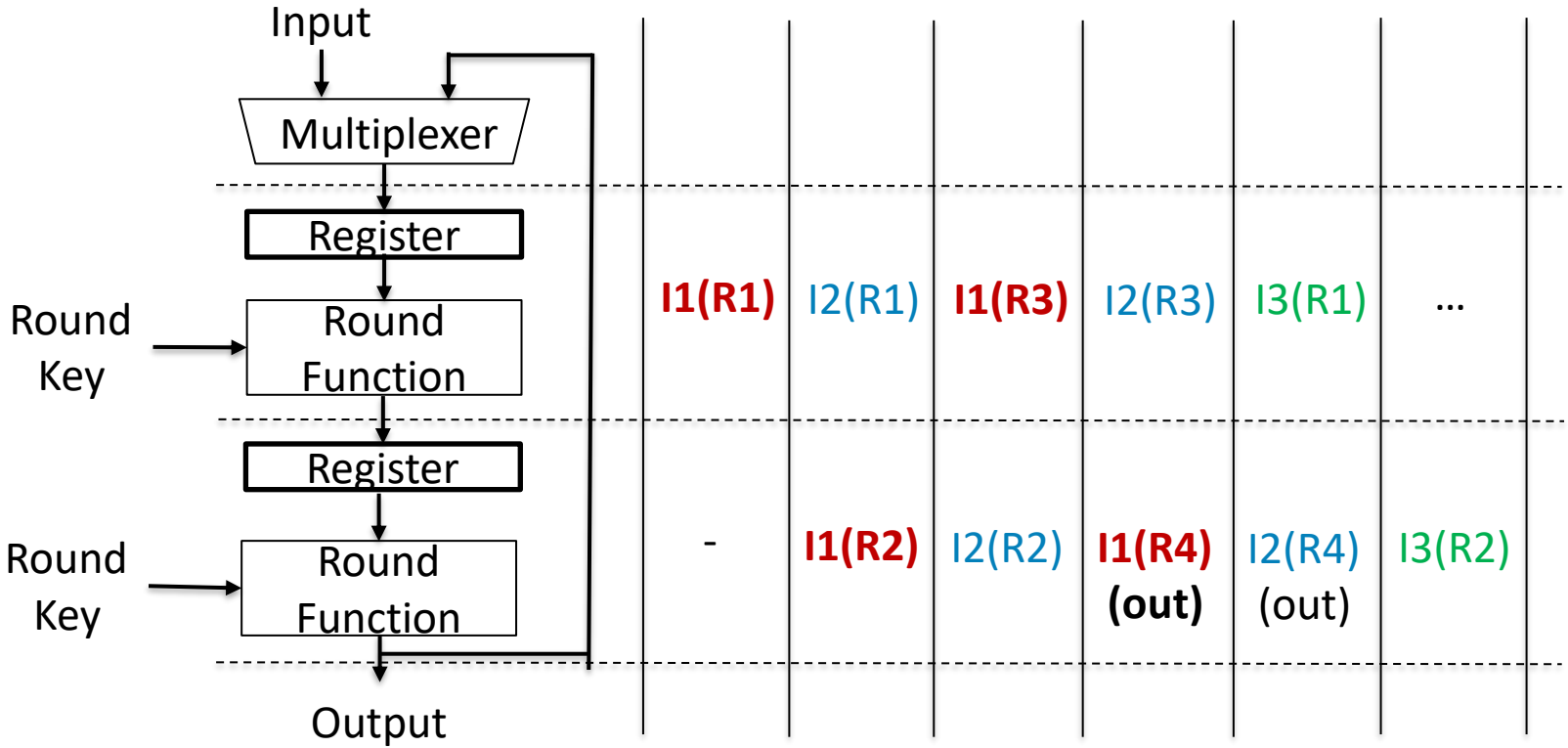
Block Cipher Implementations: Pipelining

- A traditional methodology for design of high-performance implementations.
  - Partial or full outer-loop pipelining (i.e., *K*=2 with *Nr*=4 rounds)



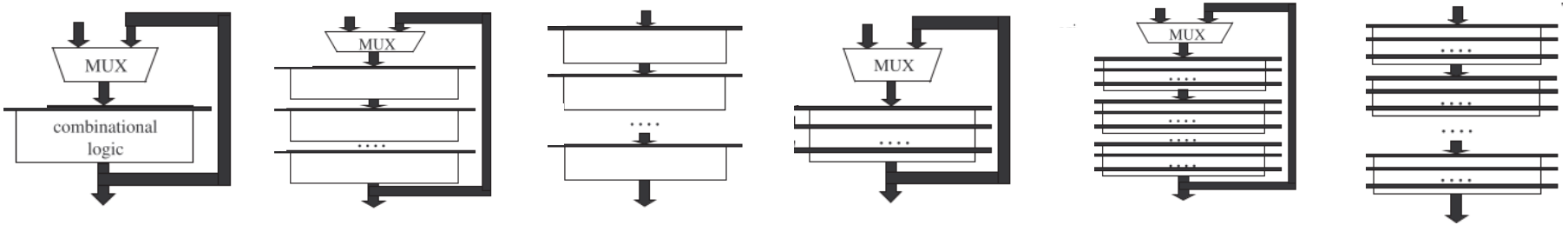
Block Cipher Implementations: Pipelining

- A traditional methodology for design of high-performance implementations.
  - Partial or full outer-loop pipelining (i.e., *K*=2 with *Nr*=4 rounds)



Block Cipher Implementations: Pipelining

- A traditional methodology for design of high-performance implementations.
  - Partial or full outer-loop pipelining.
  - Inner-loop pipelining.
  - Partial or full outer-loop pipelining with inner loop pipelining.

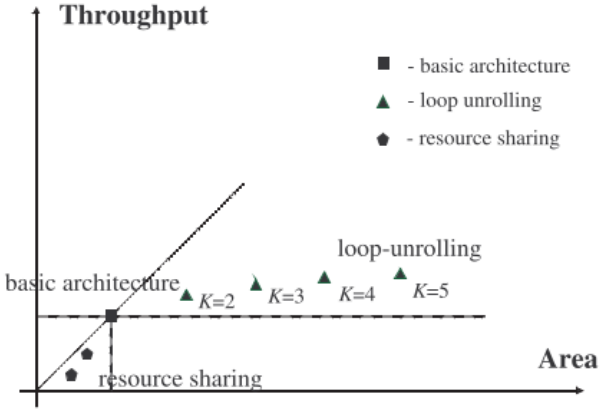


Iterative | Partial unroll | Fully unroll | Iterative with inner pipeline | Partial unroll with inner-outer pipeline | Fully unroll with inner-outer pipeline

[GC2009] K. Gaj, *FPGA and ASIC Implementations of AES*, Cryptographic Engineering, 2009.

Block Cipher Implementations: Summary

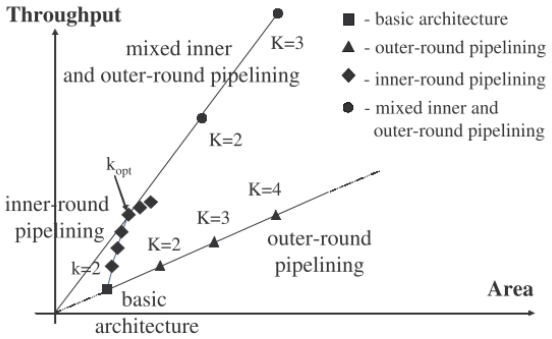
- Summary of implementation methods
  - Iterative
  - Partial unroll
  - Fully unroll

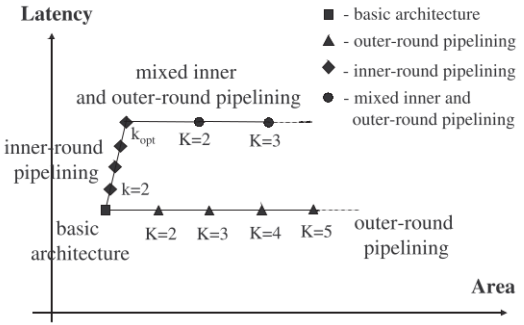


[GC2009] K. Gaj, *FPGA and ASIC Implementations of AES*, Cryptographic Engineering, 2009.

Block Cipher Implementations: Summary

- Summary of implementation methods
  - Iterative
  - Partial unroll
  - Fully unroll
  - Pipelining
    - Inner
    - Outer





[GC2009] K. Gaj, *FPGA and ASIC Implementations of AES*, Cryptographic Engineering, 2009.

References

[H2020] H. M. Heys, *A Tutorial on the Implementation of Block Ciphers: Software and Hardware Applications*, 2020, IACR ePrint 2020/1545.

[AGS2014] A. Aysu *et al.*, *SIMON Says, Break the Area Records for Symmetric Key Block Ciphers on FPGAs*, ESL, 2014.

[WOL2002] J. Wolkerstorfer *et al.*, *An ASIC Implementation of AES SBoxes*, CT-RSA, 2002.

[C2005] D. Canright, *A Very Compact S-Box for AES*, CHES, 2005.

[W2001] J. Wolkerstorfer. *An ASIC implementation of the AES MixColumn operation*. In Proc. Austrochip 2001

[GC2009] K. Gaj, *FPGA and ASIC Implementations of AES*, Cryptographic Engineering, 2009.