

# 05 - AES Key Expansion

Luc Spachmann

FSU Jena

November 23, 2023

- Erzeugt aus 128 Bit Schlüssel 11 Rundenschlüssel
- Funktion arbeitet in 32-Bit Wörtern
- Galois-Multiplikation und S-Box aus AES wiederverwendet
- Erzeugt 44 Wörter  $W[0]$  bis  $W[43]$
- Je 4 sind ein Schlüssel

- Input: Wörter  $K[0], \dots, K[3]$  (Schlüssel)

$$W[i] = \begin{cases} K[i], & \text{falls } i < 4 \\ W[i-4] \oplus \text{rcon}(\frac{i}{4}) \oplus \\ \text{SubWord}(\text{RotWord}(W[i-1])), & \text{falls } i \geq 4 \text{ und } i \bmod 4 = 0 \\ W[i-4] \oplus W[i-1] & \text{sonst} \end{cases}$$

- Output: Wörter  $W[0], \dots, W[43]$  (Rundenschlüssel)

- $\text{SubWord}(W) = \text{SubWord}(b_0, b_1, b_2, b_3) = (S[b_0], S[b_1], S[b_2], S[b_3])$
- $\text{RotWord}(W) = \text{RotWord}(b_0, b_1, b_2, b_3) = (b_1, b_2, b_3, b_0)$
- $\text{rcon}(i) = (rc_i \text{ } 00_{16} \text{ } 00_{16} \text{ } 00_{16})$  mit

$$rc_i = x^{i-1} \text{ in } GF(2^8).$$

- Alternativ  $rc_i$  speichern.

$i$	1	2	3	4	5	6	7	8	9	10
$rc_i$	01	02	04	08	10	20	40	80	1b	36

Table: Alle  $rc_i$  in Hexadezimal und  $i$  in Dezimal

- Erweitert eure Programme um die Schlüsselgenerierung
- Kombiniert eure Programme mit den Betriebsmodi
- Programmname [Betriebsmodus] [Inputdatei] [Schlüsseldatei]  
[Outputdatei] ([IV] falls nötig)