MODULE *aes*

EXTENDS *Naturals*, *Sequences*, *Integers*, *Bitwise*

VARIABLES *state*, *roundKey*, *round*, *Nb*, *Nk*, *Nr*, *encrypt*

$$
\begin{aligned}
&SBox \triangleq \langle \\
&\quad \langle 99, 124, 119, 123, 242, 107, 111, 197, 48, 1, 103, 43, 254, 215, 171, 118\rangle, \\
&\quad \langle 202, 130, 201, 125, 250, 89, 71, 240, 173, 212, 162, 175, 156, 164, 114, 192\rangle, \\
&\quad \langle 183, 253, 147, 38, 54, 63, 247, 204, 52, 165, 229, 241, 113, 216, 49, 21\rangle, \\
&\quad \langle 4, 199, 35, 195, 24, 150, 5, 154, 7, 18, 128, 226, 235, 39, 178, 117\rangle, \\
&\quad \langle 9, 131, 44, 26, 27, 110, 90, 160, 82, 59, 214, 179, 41, 227, 47, 132\rangle, \\
&\quad \langle 83, 209, 0, 237, 32, 252, 177, 91, 106, 203, 190, 57, 74, 76, 88, 207\rangle, \\
&\quad \langle 208, 239, 170, 251, 67, 77, 51, 133, 69, 249, 2, 127, 80, 60, 159, 168\rangle, \\
&\quad \langle 81, 163, 64, 143, 146, 157, 56, 245, 188, 182, 218, 33, 16, 255, 243, 210\rangle, \\
&\quad \langle 205, 12, 19, 236, 95, 151, 68, 23, 196, 167, 126, 61, 100, 93, 25, 115\rangle, \\
&\quad \langle 96, 129, 79, 220, 34, 42, 144, 136, 70, 238, 184, 20, 222, 94, 11, 219\rangle, \\
&\quad \langle 224, 50, 58, 10, 73, 6, 36, 92, 194, 211, 172, 98, 145, 149, 228, 121\rangle, \\
&\quad \langle 231, 200, 55, 109, 141, 213, 78, 169, 108, 86, 244, 234, 101, 122, 174, 8\rangle, \\
&\quad \langle 186, 120, 37, 46, 28, 166, 180, 198, 232, 221, 116, 31, 75, 189, 139, 138\rangle, \\
&\quad \langle 112, 62, 181, 102, 72, 3, 246, 14, 97, 53, 87, 185, 134, 193, 29, 158\rangle, \\
&\quad \langle 225, 248, 152, 17, 105, 217, 142, 148, 155, 30, 135, 233, 206, 85, 40, 223\rangle, \\
&\quad \langle 140, 161, 137, 13, 191, 230, 66, 104, 65, 153, 45, 15, 176, 84, 187, 22\rangle \\
&\rangle
\end{aligned}
$$

$$Rcon \triangleq \langle 1, 2, 4, 8, 16, 32, 64, 128, 27, 54\rangle$$

$$RotWord(word) \triangleq \langle word[2], word[3], word[4], word[1]\rangle$$

$$
\begin{aligned}
SubWord(word) \triangleq \langle &SBox[(word[1]\%16)+1][(word[1]\%16)+1], \\
&SBox[(word[2]\%16)+1][(word[2]\%16)+1], \\
&SBox[(word[3]\%16)+1][(word[3]\%16)+1], \\
&SBox[(word[4]\%16)+1][(word[4]\%16)+1]\rangle
\end{aligned}
$$

RECURSIVE $KeyExpansion(\_, \_)$

$$
\begin{aligned}
&KeyExpansion(initialKey, i) \triangleq \\
&\quad \text{IF } i \leq 4 \text{ THEN } \langle initialKey[1][i], initialKey[2][i], initialKey[3][i], initialKey[4][i]\rangle \\
&\quad \text{ELSE IF } i\%4 = 1 \text{ THEN} \\
&\qquad \text{LET } prevKey \triangleq KeyExpansion(initialKey, i-4) \\
&\qquad\quad temp \triangleq KeyExpansion(initialKey, i-1) \\
&\qquad\quad rconVal \triangleq \langle Rcon[(i \div 4)], 0, 0, 0\rangle \\
&\qquad\quad subRotWord \triangleq SubWord(RotWord(temp)) \\
&\qquad \text{IN } [j \in 1..4 \mapsto Xor(prevKey[j], Xor(subRotWord[j], rconVal[j], 0, subRotWord[j]), 0, prevKey[\\
&\quad \text{ELSE} \\
&\qquad \text{LET } prevKey \triangleq KeyExpansion(initialKey, i-4) \\
&\qquad\quad temp \triangleq KeyExpansion(initialKey, i-1) \\
&\qquad \text{IN } [j \in 1..4 \mapsto Xor(prevKey[j], temp[j], 0, prevKey[j])]
\end{aligned}
$$

```
RECURSIVE GFMul(_, _)
GFMul(a, b) ≜
  LET temp ≜ IF b = 1 THEN a
             ELSE IF b = 2 THEN IF (a * 2) ≥ 256 THEN Xor((a * 2)%256, 27, 0, (a * 2)%256) ELSE
             ELSE IF b = 3 THEN Xor(GFMul(a, 2), a, 0, GFMul(a, 2))
             ELSE IF b = 9 THEN Xor(GFMul(GFMul(GFMul(a, 2), 2), 2), a, 0, GFMul(GFMul(G
             ELSE IF b = 11 THEN Xor(Xor(GFMul(GFMul(GFMul(a, 2), 2), 2), GFMul(a, 2), 0,
             ELSE IF b = 13 THEN Xor(Xor(GFMul(GFMul(GFMul(a, 2), 2), 2), GFMul(GFMul(a
             ELSE IF b = 14 THEN Xor(GFMul(GFMul(GFMul(a, 2), 2), 2), GFMul(GFMul(a, 2),
             ELSE 0
  IN temp

SubBytes(s) ≜
  [i ∈ 1 .. Nb ↦ [j ∈ 1 .. Nk ↦ SBox[(s[i][j]%16) + 1][(s[i][j]%16) + 1]]]

ShiftRows(s) ≜
  [i ∈ 1 .. Nb ↦
    IF i = 1 THEN s[i]
     ELSE [j ∈ 1 .. Nk ↦ s[i][((j + i − 2)%Nk) + 1]]]

InvShiftRows(s) ≜
  [i ∈ 1 .. Nb ↦
    IF i = 1 THEN s[i]
     ELSE [j ∈ 1 .. Nk ↦ s[i][((j − i + Nk)%Nk) + 1]]]

MixColumns(s) ≜
  [i ∈ 1 .. Nk ↦
    LET s0 ≜ s[1][i]
        s1 ≜ s[2][i]
        s2 ≜ s[3][i]
        s3 ≜ s[4][i]
    IN  [j ∈ 1 .. Nb ↦
          IF j = 1 THEN Xor(Xor(Xor(GFMul(s0, 2), GFMul(s1, 3), 0, GFMul(s0, 2)), s2, 0, GFMul(s
           ELSE IF j = 2 THEN Xor(Xor(Xor(s0, GFMul(s1, 2), 0, s0), GFMul(s2, 3), 0, s0), s3, 0, s0
           ELSE IF j = 3 THEN Xor(Xor(Xor(s0, s1, 0, s0), GFMul(s2, 2), 0, s0), GFMul(s3, 3), 0, s0
           ELSE Xor(Xor(Xor(GFMul(s0, 3), s1, 0, GFMul(s0, 3)), s2, 0, GFMul(s0, 3)), GFMul(s3,

InvMixColumns(s) ≜
  [i ∈ 1 .. Nk ↦
    LET s0 ≜ s[1][i]
        s1 ≜ s[2][i]
        s2 ≜ s[3][i]
        s3 ≜ s[4][i]
    IN  [j ∈ 1 .. Nb ↦
          IF j = 1 THEN Xor(Xor(Xor(GFMul(s0, 14), GFMul(s1, 11), 0, GFMul(s0, 14)), GFMul(s2,
           ELSE IF j = 2 THEN Xor(Xor(Xor(GFMul(s0, 9), GFMul(s1, 14), 0, GFMul(s0, 9)), GFMu
```

ELSE IF $j = 3$ THEN $Xor(Xor(Xor(GFMul(s0, 13), GFMul(s1, 9), 0, GFMul(s0, 13)), GFM$
ELSE $Xor(Xor(Xor(GFMul(s0, 11), GFMul(s1, 13), 0, GFMul(s0, 11)), GFMul(s2, 9), 0, G$

$AddRoundKey(s, k) \triangleq$
$[i \in 1 \,..\, Nb \mapsto [j \in 1 \,..\, Nk \mapsto Xor(s[i][j], k[i][j], 0, s[i][j])]]$

$Round(s, k) \triangleq$
LET $newState \triangleq MixColumns(ShiftRows(SubBytes(s)))$
IN $AddRoundKey(newState, k)$

$InvRound(s, k) \triangleq$
LET $newState \triangleq SubBytes(InvShiftRows(InvMixColumns(s)))$
IN $AddRoundKey(newState, k)$

$AESProcess(e, s, k) \triangleq$
IF e THEN $Round(s, k)$
ELSE $InvRound(s, k)$

$NextRound \triangleq$
$\land\ round < Nr$
$\land\ state' = AESProcess(encrypt, state, roundKey)$
$\land\ roundKey' = roundKey$
$\land\ Nb' = Nb$
$\land\ Nk' = Nk$
$\land\ Nr' = Nr$
$\land\ round' = round + 1$
$\land\ encrypt' = encrypt$

$Init \triangleq$
$\land\ state = [i \in 1 \,..\, 4 \mapsto [j \in 1 \,..\, 4 \mapsto (i - 1) * 4 + j]]$
$\land\ round = 0$
$\land\ Nb = 4$
$\land\ Nk = 4$
$\land\ Nr = 10$
$\land\ encrypt = \text{FALSE}$
$\land\ roundKey = [i \in 1 \,..\, (4 * (Nr + 1)) \mapsto KeyExpansion([k \in 1 \,..\, 4 \mapsto [j \in 1 \,..\, 4 \mapsto (k + j + 40)\%256]]$

$Spec \triangleq$
$Init \land \Box[NextRound]_{\langle state, round, roundKey, Nb, Nk, Nr, encrypt \rangle}$