Programming

10. **Programming Exercise.** We construct a Feistel cipher using some of the components of Rijndael. The basic parameters of the cipher are:

    - 32-bit plaintext input M,
    - 32-bit key K,
    - 6 rounds of encryption,
    - The Feistel function works on a 2×2 state matrix each entry a polynomial of degree at most 3. All arithmetic operations are performed in $\mathbb{F}_{2^4}$ with respect to the irreducible polynomial $p(x) = x^4 + x + 1$.

      The single operations are altered as follows:

      SubBytes In the first step take the inverse of the input in $\mathbb{F}_{2^4}$ and then perform the transformation:

      $$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

      MixColumns Perform the transformation for $i = 0, 1$:

      $$\begin{bmatrix} b_{0,i} \\ b_{1,i} \end{bmatrix} = \begin{bmatrix} 0x03 & 0x07 \\ 0x04 & 0x03 \end{bmatrix} \cdot \begin{bmatrix} a_{0,i} \\ a_{1,i} \end{bmatrix}$$

      ShiftRows Do not shift the first row, shift the second row cyclically one left.

      MultRoundKey The key $K_i = k_1 \| k_2 \| k_3 \| k_4$ and the state matrix A,

      $$\begin{bmatrix} k_1 & k_3 \\ k_2 & k_4 \end{bmatrix} \cdot \begin{bmatrix} a_1 & a_3 \\ a_2 & a_4 \end{bmatrix} = \begin{bmatrix} a'_1 & a'_3 \\ a'_2 & a'_4 \end{bmatrix}$$

      where the $k_j, a_j, j = 1, \ldots, 3$ are 4 bit each.

      Key schedule $K_0, K_1, \ldots, K_5$ are computed as follows:

      The algorithm is altered to perform each operation with a word length of 8 bits, i.e., each W_i is of length 8 bits.

      We replace the rotation operation by $T := W_{4i-1} \lll 3$ and compute the round constant as $RC_i = x^{i+3} (\bmod\ x^4 + x + 1)$.

      We define two functions l, r to access the first and last 4 bits of an 8 bit word, respectively, such that $T = l(T) \| r(T)$. The *SubBytes* function is defined as above and applied as $T :=$*SubBytes*$(l(T)) \|$*SubBytes*$(r(T))$.

      We then compute the 2×2 key matrices as

      $$K_i = \begin{bmatrix} [l(W_{4i}) \otimes r(W_{4i})] \oplus l(W_{4i}) \oplus r(W_{4i}) & [l(W_{4i+2}) \otimes r(W_{4i+2})] \oplus l(W_{4i+2}) \oplus r(W_{4i+2}) \\ [l(W_{4i+1}) \otimes r(W_{4i+1})] \oplus l(W_{4i+1}) \oplus r(W_{4i+1}) & [l(W_{4i+3}) \otimes r(W_{4i+3})] \oplus l(W_{4i+3}) \oplus r(W_{4i+3}) \end{bmatrix}$$

      with the $\otimes$ and $\oplus$ operation in the ring $\mathbb{F}_{2^4}$.

The ***Feistel function*** is then computed as follows: Let the input of the i^{th} round be $M_i = L_i \| R_i$, where the right message block $R_i = r_1 \| r_2 \| r_3 \| r_4$ with $r_j, j = 1, \ldots, 4$ are 4 bit each.
$F(R_i, K_i) :=$
$A = \begin{bmatrix} a_1 & a_3 \\ a_2 & a_4 \end{bmatrix} := \begin{bmatrix} r_1 & r_3 \\ r_2 & r_4 \end{bmatrix}$
$A :=$ SubBytes(A)
$A :=$ MultRoundKey(A, K_i)
$A :=$ MixColumns(A)
$A :=$ ShiftRows(A)
$R_{i+1} := a_1 \| a_2 \| a_3 \| a_4$

Implement the cipher together with Cipher Feedback mode (CFB) with initialisation vector $IV = 0xA7E42F5B$ to encrypt 8 messages which will be available online in ASCII format from the website soon.

Cryptography Glossary 4

C2	Short for Cryptomeria.	34
CBC	Short for Cipher Block Chaining	30
CFB	Short for Cipher Feedback mode	31
Cipher Block Chaining mode	A mode of operation for block ciphers that propagates context information.	30
Cipher Feedback mode	A mode of operation that turns a block cipher effectively into a stream cipher by using context information.	31
Counter mode	A mode of operation that turns a block cipher effectively into a stream cipher by using a counter.	33
Cryptomeria	A Feistel Cipher used as content protection mechanism for multimedia DVDs.	34
CTR	Short for Counter mode	33
ECB	Short for Electronic Codebook mode	29
Electronic Codebook mode	A naïve mode of operation for block ciphers that encrypts every block independently.	29
Initial Value	An arbitrary value used to kick off encryption in block cipher modes. Initial values are normally denoted as IV and do not need to be kept secret.	30
Initialisation Vector	See Initial Value.	30
Modes of Operation	Ways of applying block ciphers to encode large messages.	29
Nonce	A number that is only used once. It is usually a random or pseudo-random number.	33
OFB	Short for Output Feedback mode	32
Output Feedback mode	A mode of operation that turns a block cipher effectively into a stream cipher by using context information.	32
Padding	Adding nulls at the end of messages to obtain the required block size. It is important that the nulls do not obscure the message, such that its original length can be recovered. The simplest is padding is achieved by adding null bits.	29