USN | | | | | | | | | | |

Internal Assessment Test V – Feb. 2022

Sub:	**Cryptography**					Sub Code:	**18EC744**	Branch:	**EC**
Date:	**08/02/2022**	Duration:	90 min's	Max Marks:	**50**	Sem / Sec:	**7 A, B, C, D**		OBE

Answer all Questions

		MARKS	CO	RBT
1	Compare AES to DES for each of the following elements of DES: **a)** XOR of subkey material with the input of the f function. **b)** XOR of the f function output with the left half of the block. **c)** f function **d)** Permutation P **e)** Swapping of half of the block.	**[10]**	CO2	L3
2	Given the plaintext [000102030405060708090A0B0C0D0E0F] and the key [01010101010101010101010101010101]. Show the **a)** State matrix **b)** Initial round key **c)** Sub Byte **d)** Shift rows **e)** Mix columns	**[10]**	CO2	L3
3	Illustrate the following with necessary diagram: (i) Feistel encryption and decryption process. (ii) Single DES encryption.	**[10]**	CO2	L1
4	Explain AES key generation algorithm with appropriate block diagram.	**[10]**	CO2	L2
5	Explain the process of AES encryption and decryption with necessary diagram.	**[10]**	CO2	L2

----------All The Best---------

CMR INSTITUTE OF TECHNOLOGY

USN | | | | | | | | | | |

Scheme and Solution of Internal Assesment Test – V

Sub:	**Cryptography**		Sec	7 A, B, C, D			Code:	**18EC744**
Date:	**08/02/2022**	Duration:	90 mins	Max Marks:	50	Sem: VII	Branch:	ECE

Solution

1 *Compare AES to DES for each of the following elements of DES:* **[10 marks]**

a) *XOR of subkey material with the input of the f function.*

b) *XOR of the f function output with the left half of the block.*

c) *f function d) Permutation P e) Swapping of half of the block.*

Ans

Parameter	AES	DES
XOR of subkey material with the input of the f function	The subkey of length 128-bit is applied to the Complex function 'F'. The Complex function operates on complete 128-bit of the message and the XOR of subkey with the input of the function 'F' generates 128-bit output.	The subkey of length 48-bit is applied to the Complex function 'F'. The Complex function operates only on 32-bit of the message and the XOR of subkey with the input of the function 'F' generates 32-bit output.
XOR of the f function output with the left half of the block	This is not the case in AES. The output of the complex function 'F' is provided directly to the next round.	Yes, the output of the Complex function 'F' is XORed with the left half of the message block to generate the new right block of message.
F function	The round function itself can be treated as the complex function, which perform 4 operations.	The complex function consists of Expansion of D-box, XOR, S-Box and the straight S-Box.
Permutation P	There is no such concept of permutation P in AES.	There are 2 permutation choice being performed one is PC-1 which is operated on 64-bit Key and generates the output of 56-bit. The PC-2 operates on the 56-bit and produce an output of 48-bit.
Swapping of half of the block	Not Performed in AES	Yes the swapping of half of the block is performed at the end of each round.

2 *Given the plaintext [00010203040506070809 0A0B0C0D0E0F] and the key [01010101010101010101010101010101]. Show the* **[10 marks]**

State matrix b) Initial round key c) Sub Byte d) Shift rows e) Mix columns

Ans **This Question is very lengthy and not possible to solve within 3 hours if not provided with the s-Box and also Mix column multiplication also require at least require 3hrs to calculate the result as it uses the $GF(2^8)$ multiplication and addition.**

AES Algorithm:

Plain Text = [0001 0203 0405 0607 0809 0A0B 0C0D 0E0F]

Key = [0101 0101 0101 0101 0101 0101 0101 0101] **[10 marks]**

(i) State Matrix:

00	04	08	0C
01	05	09	0D
02	06	0A	0E
03	07	0B	0F

(ii) Initial Round Key:

01	01	01	01
01	01	01	01
01	01	01	01
01	01	01	01

(iii) Sub Bytes:

00	04	08	0C
01	05	09	0D
02	06	0A	0E
03	07	0B	0F

from **S-Box** →

63	F2	30	FE
7C	6B	01	D7
77	6F	67	AB
7B	C5	2B	76

(iv) Shift rows:

63	F2	30	FE
6B	01	D7	7C
67	AB	77	6F
76	7B	C5	2B

(v) Mix Columns:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \cdot \begin{bmatrix} 63 & F2 & 30 & FE \\ 6B & 01 & D7 & 7C \\ 67 & AB & 77 & 6F \\ 76 & 7B & C5 & 2B \end{bmatrix}$$

3 *Illustrate the following with necessary diagram:*

(i) Feistel encryption and decryption process. **[10 marks]**

(ii) Single DES encryption.

Ans **(i) FEISTEL CIPHER STRUCTURE:**

1. The inputs to the encryption algorithm are a plaintext block of length $2w$ bits and a key K.
2. The plaintext block is divided into two halves, $L0$ and $R0$.
3. The two halves of the data pass through n rounds of processing and then combine to produce the ciphertext block.
4. Each round i has as inputs L_{i-1} *and* R_{i-1} derived from the previous round, as well as a subkey K_i derived from the overall K. The subkeys K_i are different from K and from each other.
5. 16 rounds are used, although any number of rounds could be implemented. All rounds have the same structure.
6. A **substitution** is performed on the left half of the data. This is done by applying a *round function* F to the right half of the data and then taking the exclusive-OR of the output of that function and the left half of the data.
7. The round function F has the same general structure for each round. The round function F is represented as $F(RE_i, K_{i+1})$
8. Following this substitution, a **permutation** is performed that consists of the interchange of the two halves of the data.
9. Feistel network depends on the choice of the following parameters and design features: **[10 marks]**
   a) **Block size:** larger block sizes mean greater security, but it reduces encryption/decryption speed for a given algorithm. The greater security is achieved by greater diffusion. Traditionally, a block size of 64 bits has been considered a reasonable tradeoff and was nearly universal in block cipher design. However, the new AES uses a 128-bit block size.
   b) **Key size:** Larger key size means greater security but may decrease encryption decryption speed. The greater security is achieved by greater resistance to brute-force attacks and greater confusion. Key sizes of 64 bits or less are now widely considered being inadequate and 128 bits has become a common size.
   c) **Number of rounds:** The essence of the Feistel cipher is that a single round offers inadequate security but that multiple rounds offer increasing security. A typical size is 16 rounds.
   d) **Subkey generation algorithm:** Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis.
   e) **Round function F:** Again, greater complexity generally means greater resistance to cryptanalysis.
10. There are two other considerations in the design of a Feistel cipher:

a) **Fast software encryption/decryption:** Encryption is embedded in applications hence the speed of execution of the algorithm becomes a concern.
b) **Ease of analysis:** Although we would like to make our algorithm as difficult as possible to cryptanalyze, there is great benefit in making the algorithm easy to analyze. That is, if the algorithm can be concisely and clearly explained, it is easier to analyze that algorithm for cryptanalytic vulnerabilities and therefore develop a higher level of assurance as to its strength. DES, for example, does not have an easily analyzed functionality.

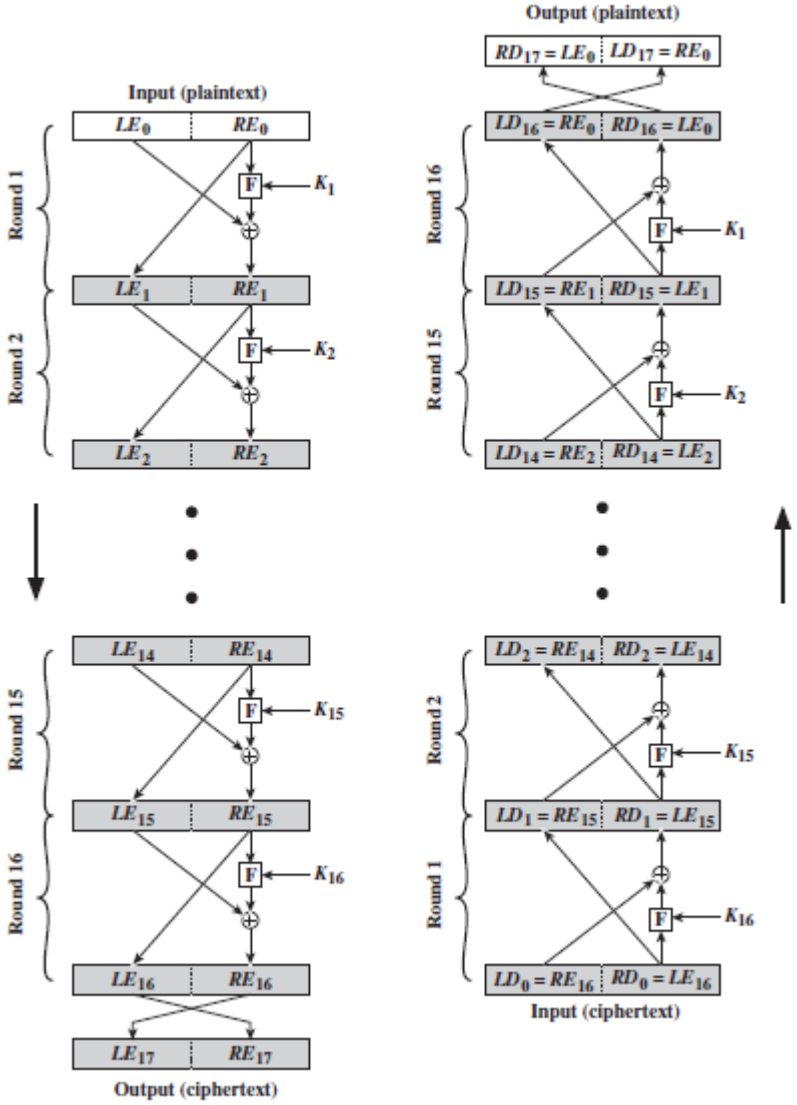


Figure: Feistel Encryption and Decryption (16 rounds)

11. **Feistel Decryption Algorithm:**
    a) Decryption with a Feistel cipher is same as the encryption process.
    b) In decryption the ciphertext is used as input to the algorithm, and the subkeys K_i are used in reverse order.
    c) That is, K_n is used in the first round, K_{n-1} in the second round, and so on, until K_1 is used in the last round. It is an advantage because no need to implement two different algorithms; one for encryption and one for decryption.
    d) For clarity, the notation LE_i *and* RE_i is used for data traveling through the encryption algorithm and LD_i *and* RD_i for data traveling through the decryption algorithm.
    e) The diagram indicates that, at every round, the intermediate value of the decryption process is equal to the corresponding value of the encryption process with the two halves of the value swapped. i.e. $RE_i||LE_i = LD_{16-i}||RD_{16-i}$
    f) Example: (for better clarity)

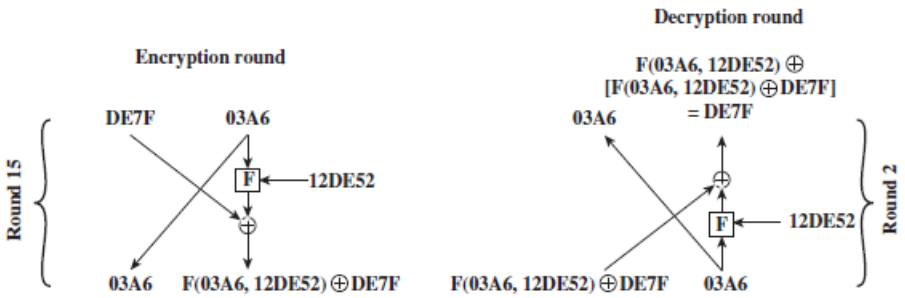
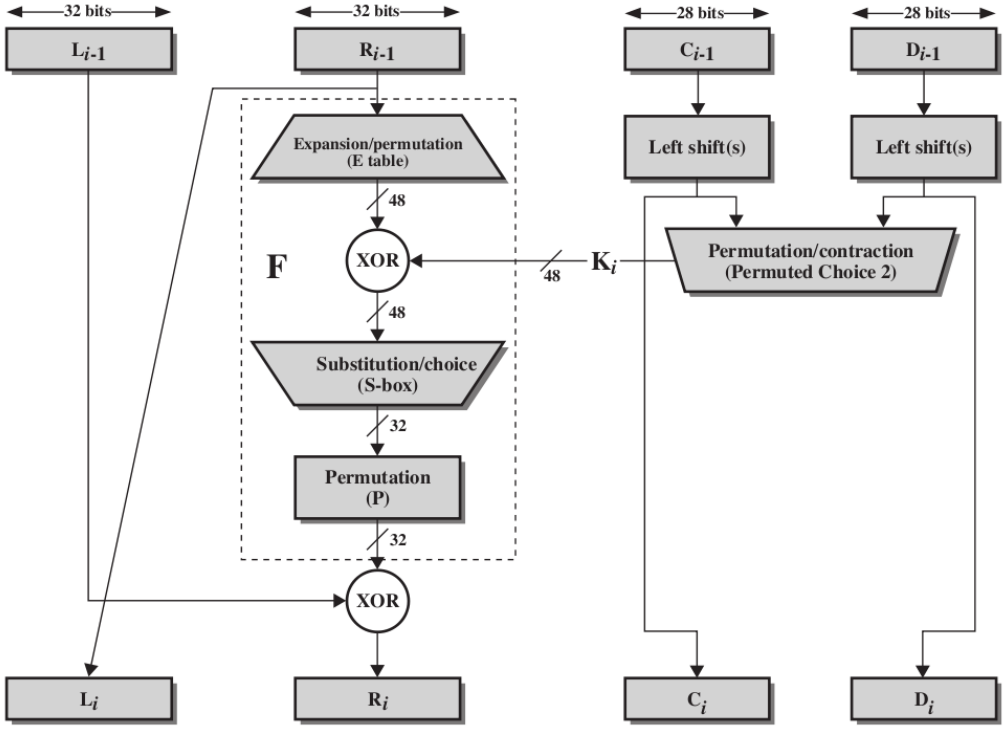


Figure: Feistel Example

(ii) Details of Single Round DES:

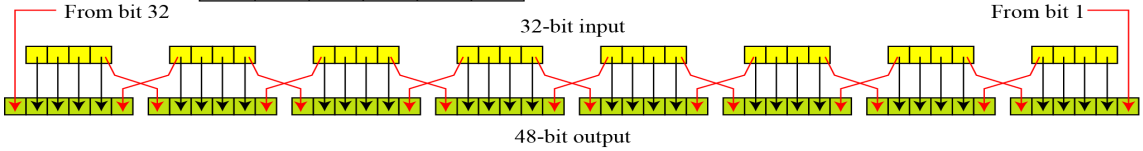
Figure below shows the internal structure of a single round. Again, begin by focusing on the left-hand side of the diagram. The left and right halves of each 64-bit intermediate value are treated as separate 32- bit quantities, labelled L (left) and R (right). As in any classic Feistel cipher, the overall processing at each round can be summarized in the following formulas:

$$L_i = R_{i-1}$$
$$R_i = L_{(i-1)} \oplus F\left(R_{(i-1)}, K_i\right)$$



Expansion: The round key K_i is 48 bits, and the R is 32 bits. The R is first expanded to 48 bits by using permutation plus expansion table as shown below.

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	28
24	25	26	27	28	29
28	29	30	31	32	1



XOR: The resulting 48 bits are XOR with K_i

Substitution Table: These 48 bits are passed through the substitution function that produces a 32 bits output which is permuted based on predefined rule as shown in table below.

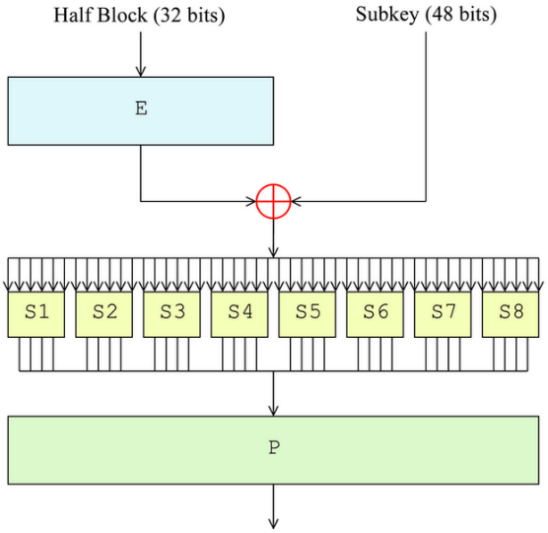
The round key K_i is 48 bits. The R input is 32 bits. This R input is first expanded to 48 bits by using a table that defines a permutation plus an expansion that involves duplication of 16 of the R bits. The resulting 48 bits are XORed with K_i. This 48-bit result passes through a substitution function that produces a 32-bit output. The role of the S-boxes in the function is illustrated in figure shown below. The substitution consists of a set of eight S-boxes, each of which accepts 6 bits as input and produces 4 bits as output.

The substitution consists of 8 S-Boxes, which accepts 6 bits as input and produces 4 bits as output. The 1st and last bit of the input to S-Box S_i forms the row and the remaining 4 bits represents the column.

E.g. In S_1, for the input 011001, the row is 01 i.e. 1st row and 1100 i.e. 12th column, the value at 1st row and 12th column is 9 i.e. 1001.

The output of the S-Boxes is again permuted as

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25



4 *Explain AES key generation algorithm with appropriate block diagram.* **[10 marks]**

Ans **AES KEY EXPANSION:**

Key Expansion Algorithm:

1. The AES key expansion algorithm takes as input a four-word (16-byte) key and produces a linear array of 44 words (176 bytes).
2. The key is copied into the first four words of the expanded key.
3. The remainder of the expanded key is filled in four words at a time.
4. Each added word **w**[i] depends on the immediately preceding word, **w**[i - 1], and the word four positions back, **w**[i - 4] and a simple XOR is used
5. For a word whose position in the **w** array is a multiple of 4, a more complex function 'g' is used.
6. The generation of the expanded key, using the symbol g to represent that complex function.
7. The function 'g' consists of the following sub-functions:
    a. Perform a one-byte left circular rotation. This means that an input word [B0, B1, B2, B3] is transformed into [B1, B2, B3, B0].
    b. Perform a byte substitution using the S-box table.
    c. The result of step 1 and step 2 is XORed with a Round Constant RC[j]

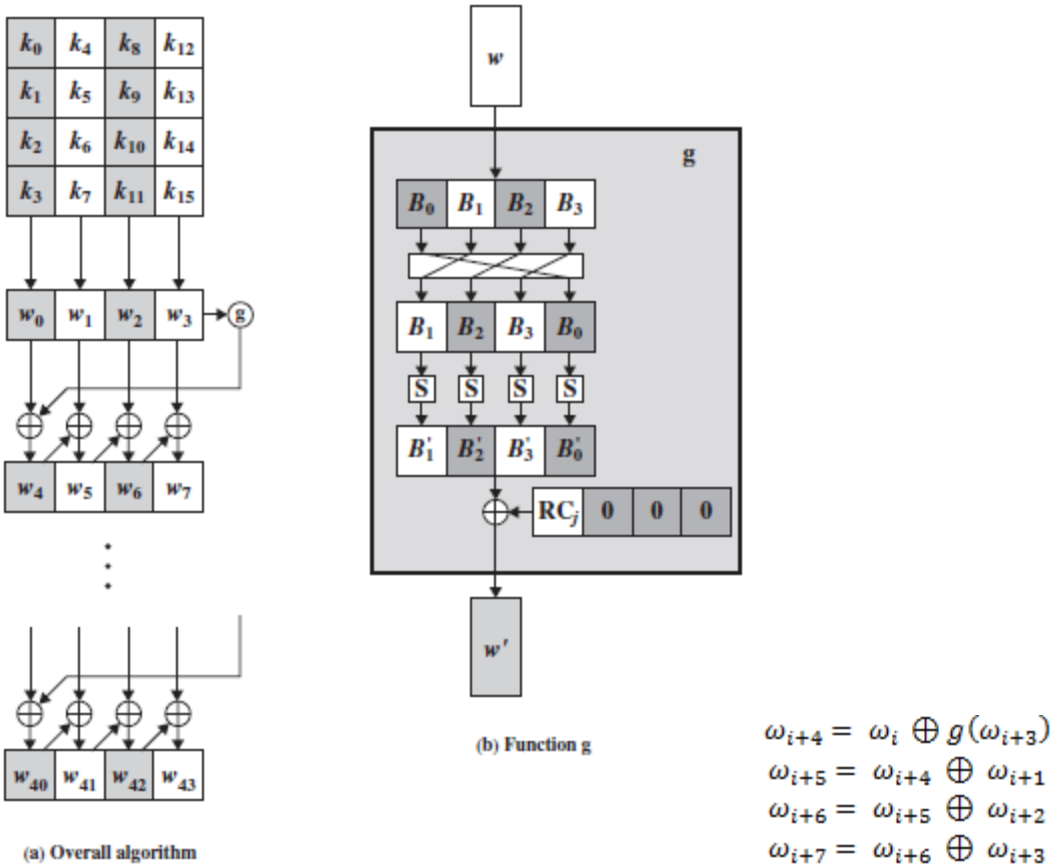


Figure: AES Key Expansion

8. The round constant is a word in which the three rightmost bytes are always 0.

Rcon Constants (Base 16)			
Round	Constant(Rcon)	Round	Constant(Rcon)
1	01 00 00 00	6	20 00 00 00
2	02 00 00 00	7	40 00 00 00
3	04 00 00 00	8	80 00 00 00
4	08 00 00 00	9	1B 00 00 00
5	10 00 00 00	10	36 00 00 00

5 *Explain the process of AES encryption and decryption with necessary diagram.* **[10 marks]**

Ans

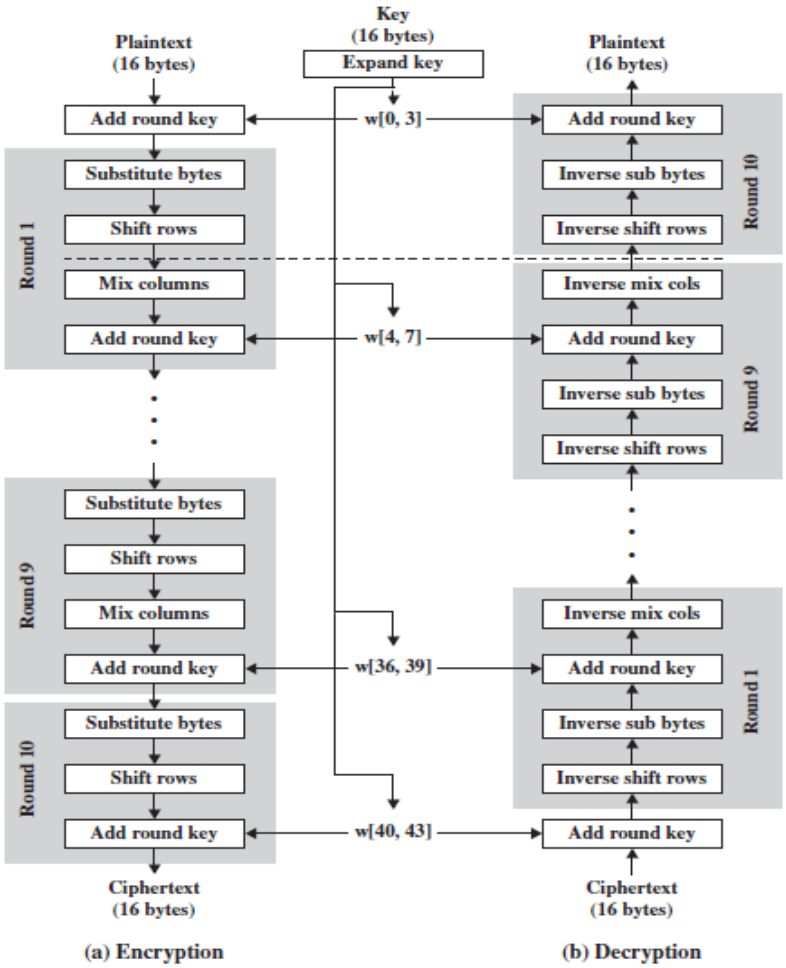


Figure: AES Encryption and Decryption

AES doesn't use the Feistel structure. Feistel structure, half of the data block is used to modify the other half of the data block and then the halves are swapped. AES instead processes the entire data block as a single matrix during each round using substitutions and permutation. The key that is provided as input is expanded into an array of forty-four 32-bit words, **w**[*i*]. Four different stages are used, one of permutation and three of substitution:

a) **Substitute bytes:** Uses an S-box to perform a byte-by-byte substitution of the block
b) **ShiftRows:** A simple permutation
c) **MixColumns:** A substitution that makes use of arithmetic over GF(28)
d) **AddRoundKey:** A simple bitwise XOR of the current block with a portion of the expanded key

The cipher begins with an AddRoundKey stage, followed by nine rounds that each includes all four stages, followed by a tenth round of three stages. AddRoundKey stage makes use of the key.The cipher begins and ends with an AddRoundKey stage. Each stage is easily reversible. For the Substitute Byte, ShiftRows, and MixColumns stages, an inverse function is used in the decryption algorithm. For the AddRoundKey stage, the inverse is achieved by XORing the same round key to the block, using the result that A $\oplus$ B $\oplus$ B = A. In AES, the decryption algorithm is not identical to the encryption algorithm.

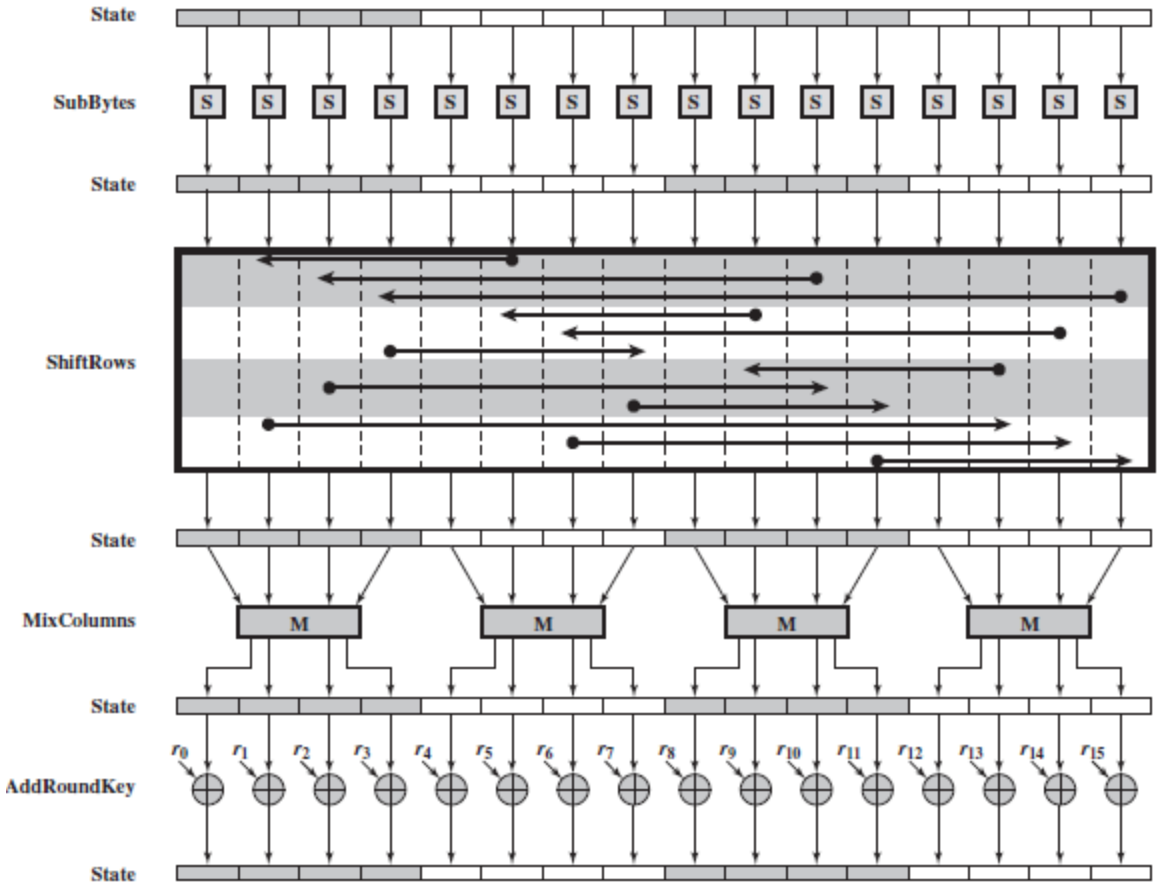


Figure: AES Encryption Round

As all stages are reversible, it is easy to perform decryption to recover the plain text. Encryption and decryption going in opposite vertical directions. The first *N* - 1 rounds consist of four distinct transformation functions:

- SubBytes,
- ShiftRows,
- MixColumns,
- AddRoundKey

The final round contains only three transformations those are SubBytes, ShiftRows and AddRoundKey, and there is an initial single transformation (AddRoundKey) before the first round, which can be considered Round 0.