AES
Advanced Encryption Standards

Reference: Chapter5, Cryptography and Network Security, William Stallings

AES

- NIST Request New Algorithm in 1997
- AES is proposed by Dr. Joan Daemen and Dr. Vincent Rijmen, Belgium
- NIST Select AES (Rijndael) in 2001
- AES Replaces DES and 3DES

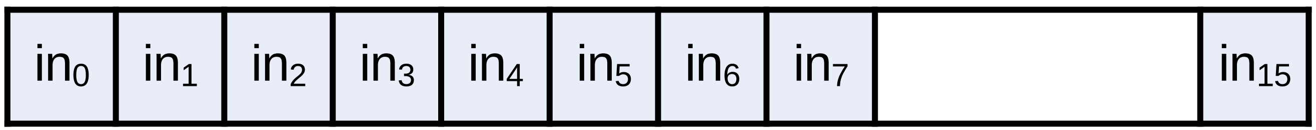
The AES Cipher

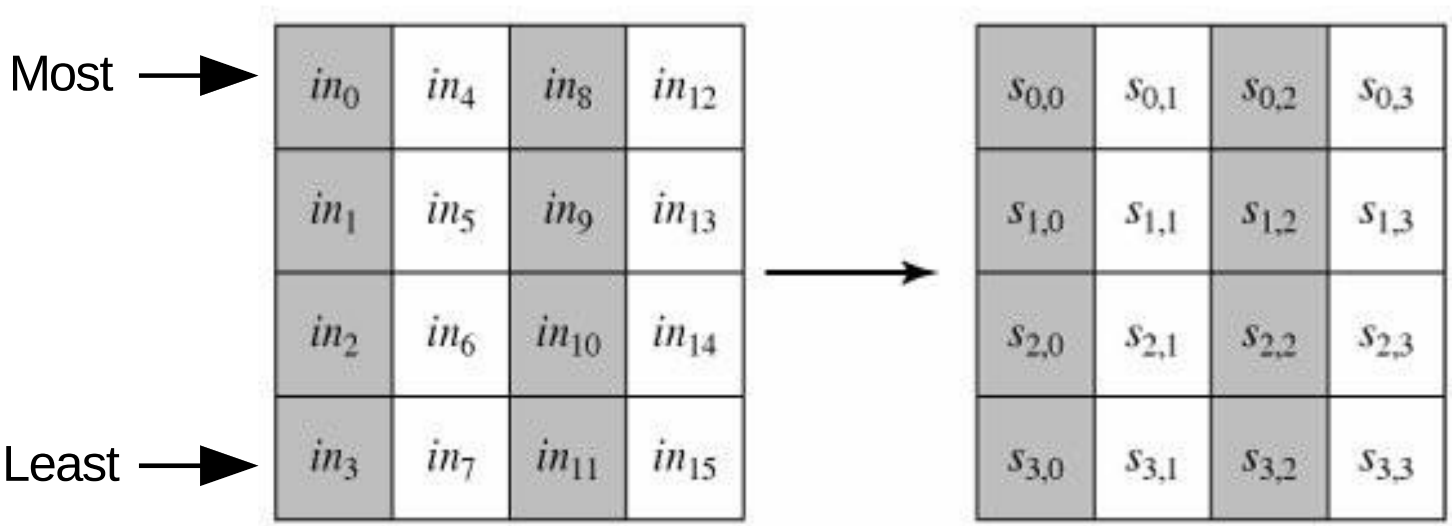
- Not Feistel Cipher
- Uses Modular Polynomial Arithmetic GF(2^8)

$$m(x) = x^8 + x^4 + x^3 + x + 1$$

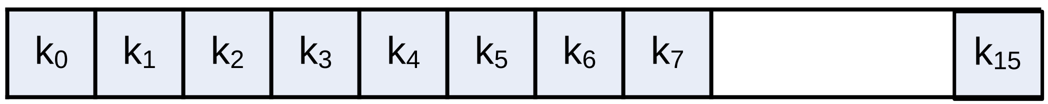
• Plaintext Block Size	16	Byte		
• Variable Key Size	<u>**16**</u>	24	32	Byte
• Number of Rounds	<u>**10**</u>	12	14	Round
• Round Key Size	<u>**16**</u>	16	16	Byte

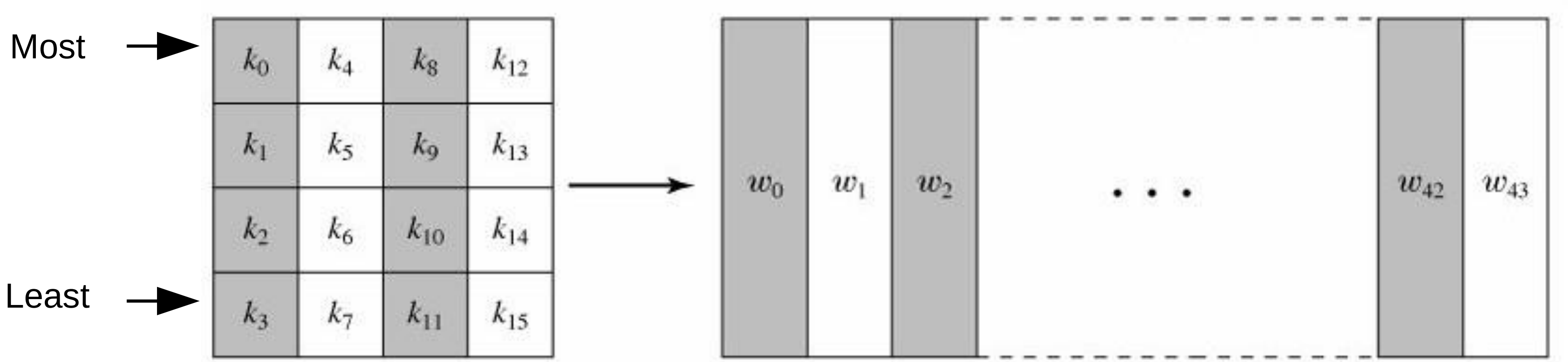
Input Preparation



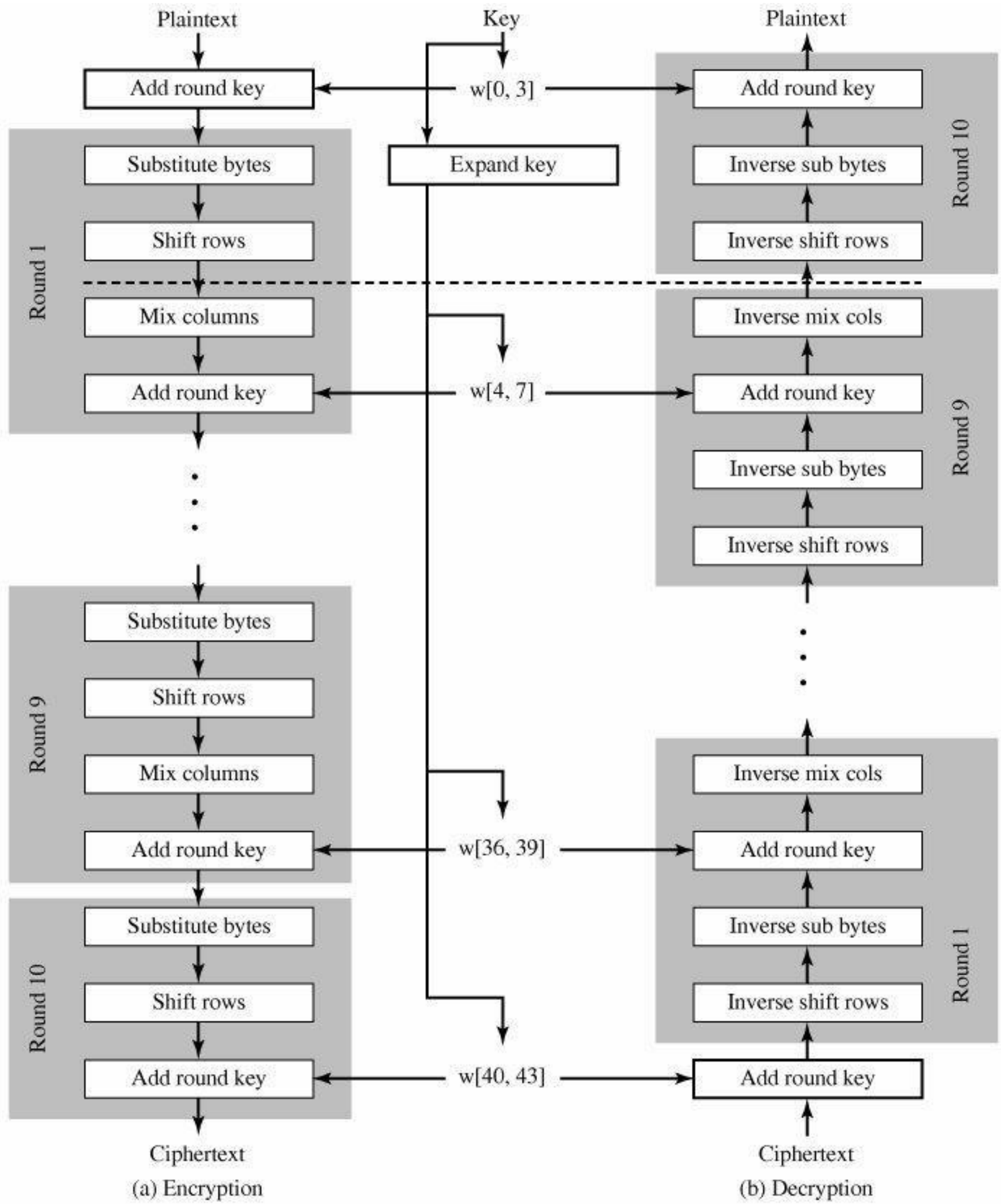


AES Key Expansion





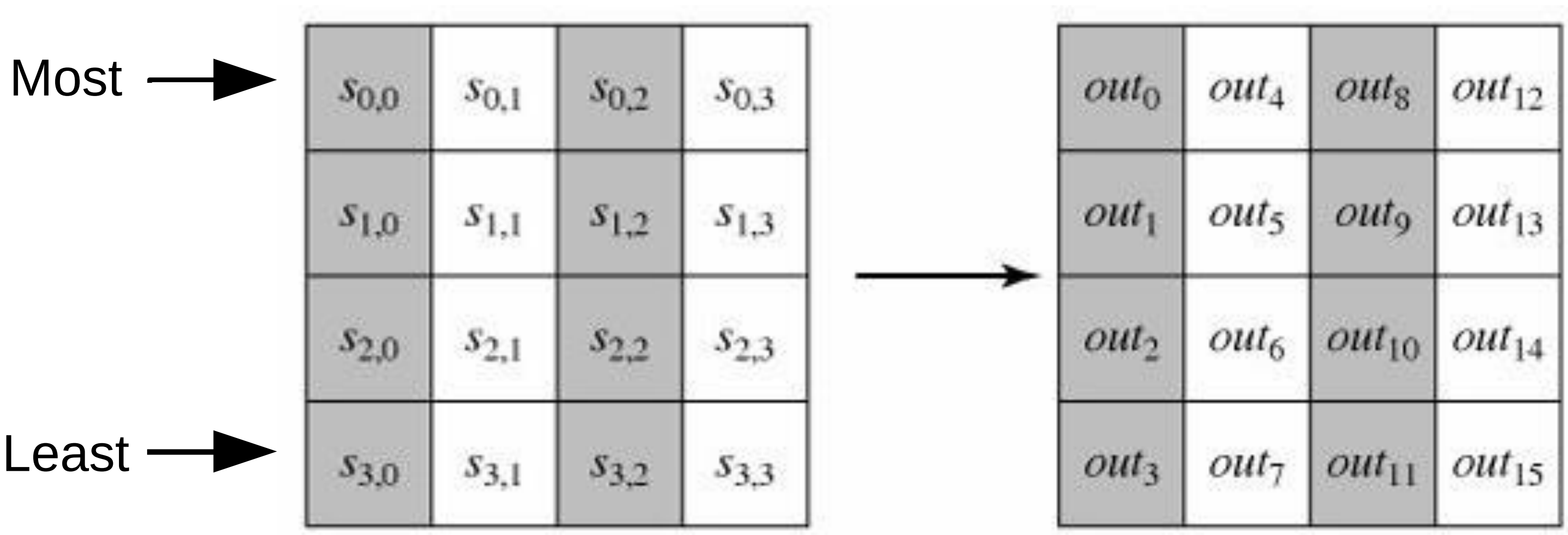
(b) Key and expanded key



AES Structure

- Round Steps
  - Substitute Bytes
  - Shift Rows
  - Mix Columns
  - Add Round Key
- All Rounds is Similar Except the Last (3 Steps)

Output Construction

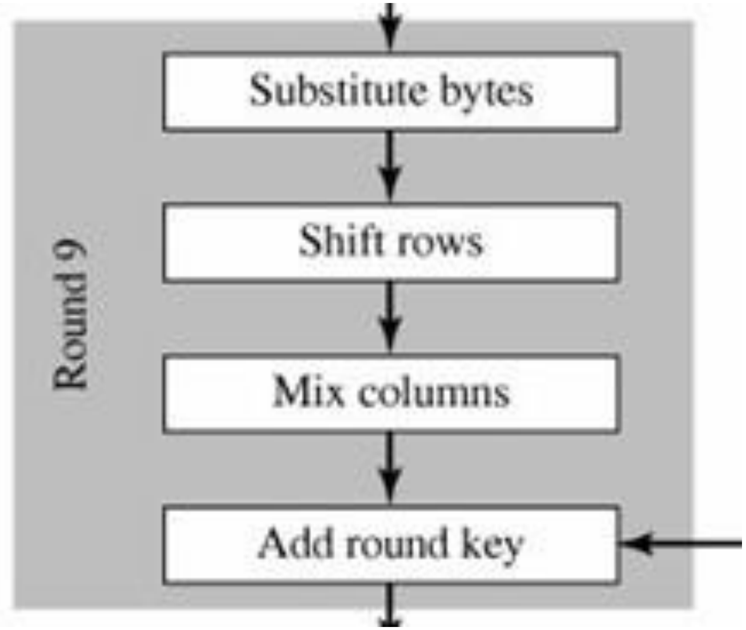


out_0	out_1	out_2	out_3	out_4	out_5	out_6	out_7		out_{15}

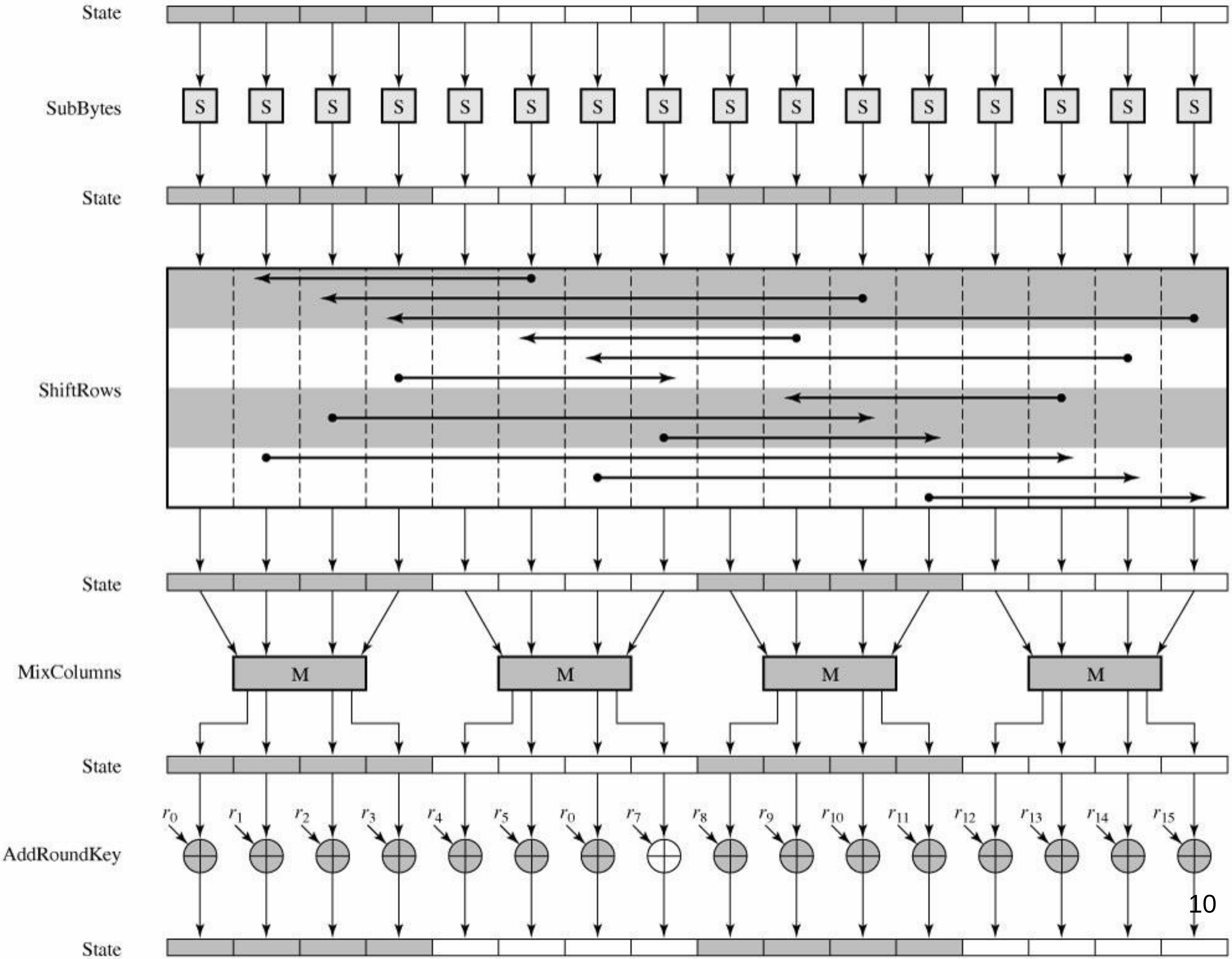
AES Round

Description

AES Round



- Substitute Bytes
- Shift Rows
- Mix Columns
- Add Round Key



State
SubBytes
S
State
ShiftRows
State
MixColumns
M
State
AddRoundKey
r_0 r_1 r_2 r_3 r_4 r_5 r_0 r_7 r_8 r_9 r_{10} r_{11} r_{12} r_{13} r_{14} r_{15}
State

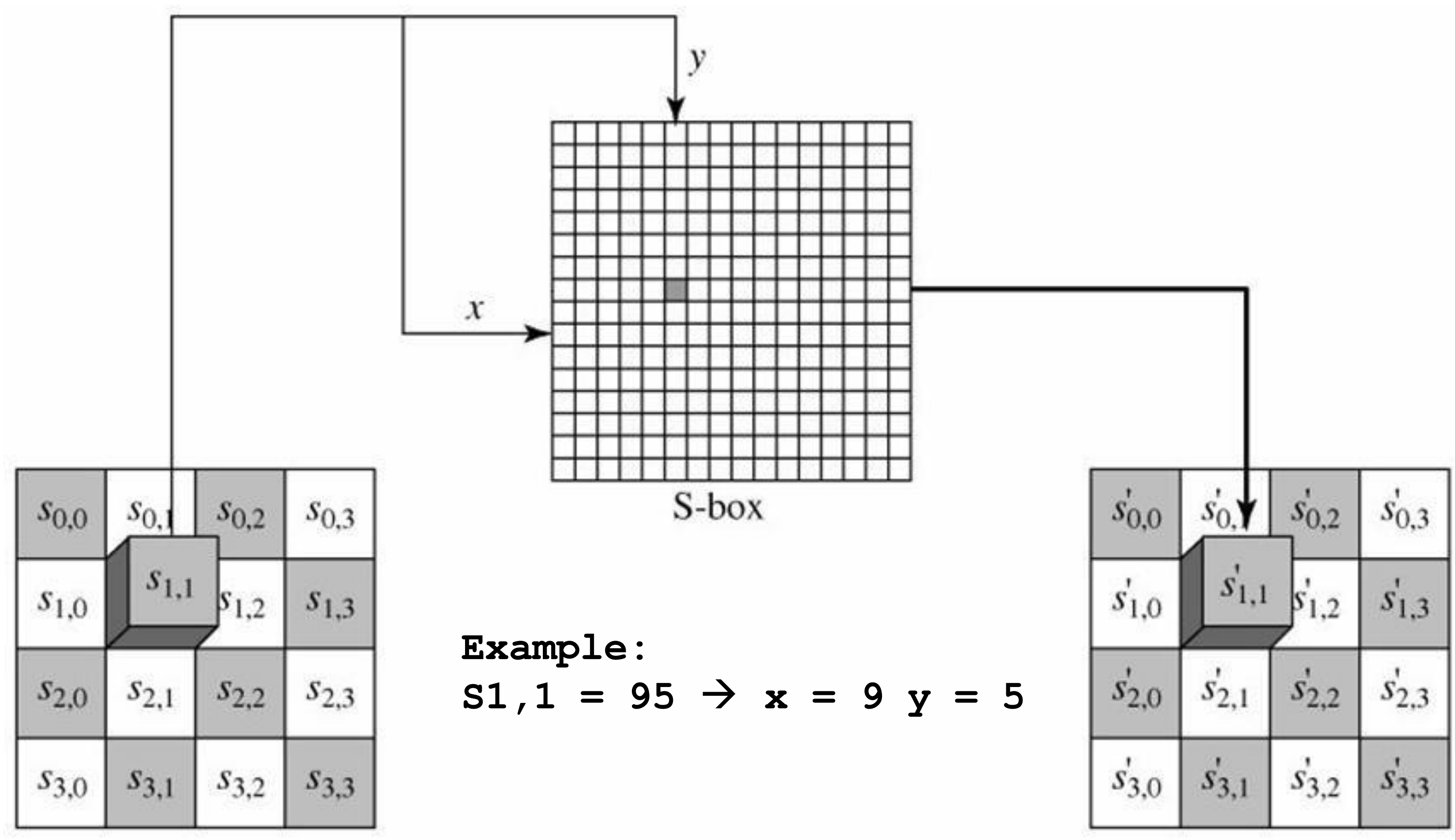
AES Round

- Substitute Bytes
  - Substitution using AES S-Box Per Single Byte
- Shift Rows
  - Simple Permutation
- Mix Columns
  - Substitution on $GF(2^8)$ Per Column (4 Bytes)
- Add Round Key
  - Simple XOR with the Scheduled Key

AES Round

- All Stages Is Reversible
  - Substitute Bytes uses Inverse AES S-Box
  - Shift Rows uses Opposite Shift Operations
  - Mix Columns uses Inverse Arithmetic in $GF(2^8)$
  - Add Round Key uses XOR (Reversible)

Substitute Bytes



Example:
S1,1 = 95 → x = 9 y = 5

Substitute Bytes – One Byte Example

95

x \ y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	[illegible]	[illegible]	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	[illegible]	[illegible]	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	[illegible]	[illegible]	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

$\boxed{\textbf{2A}}$

AES S-Box for Encryption

Substitute Bytes – One Byte Example

2A

95

		y															
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
x	**0**	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
	1	7C	E3	39	82	9B	2F	[illegible]	[illegible]	34	8E	43	44	C4	DE	E9	CB
	2	54	7B	94	32	A6	C2	[illegible]	[illegible]	EE	4C	95	0B	42	FA	C3	4E
	3	08	2E	A1	66	28	D9	24	[illegible]	76	5B	A2	49	6D	8B	D1	25
	4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
	5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
	6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
	7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
	8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
	9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
	A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
	B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
	C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
	D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
	E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
	F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

AES Inverse S-Box for Decryption

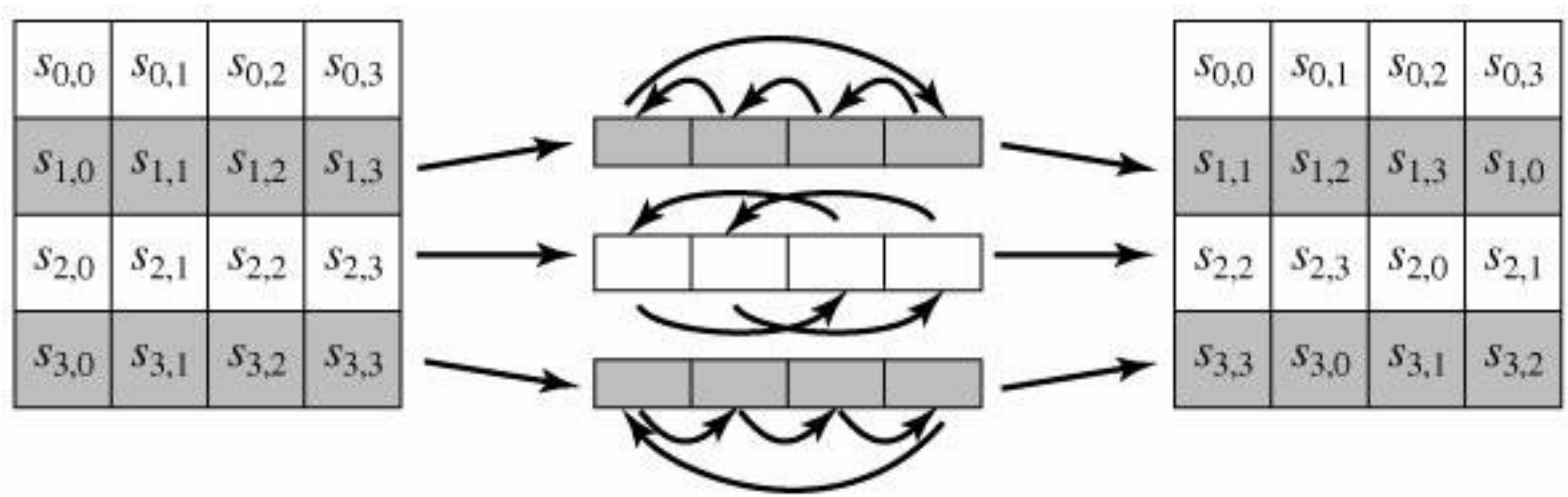
Substitute Bytes – Block Example

EA	04	65	85
83	45	5D	96
5C	33	98	B0
F0	2D	AD	C5

→

87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6

Shift Rows



Row	Shift for Encryption	Shift for Encryption
0	Circulate Left 0 Byte	Circulate Right 0 Byte
1	Circulate Left 1 Byte	Circulate Right 1 Byte
2	Circulate Left 2 Byte	Circulate Right 2 Byte
3	Circulate Left 3 Byte	Circulate Right 3 Byte

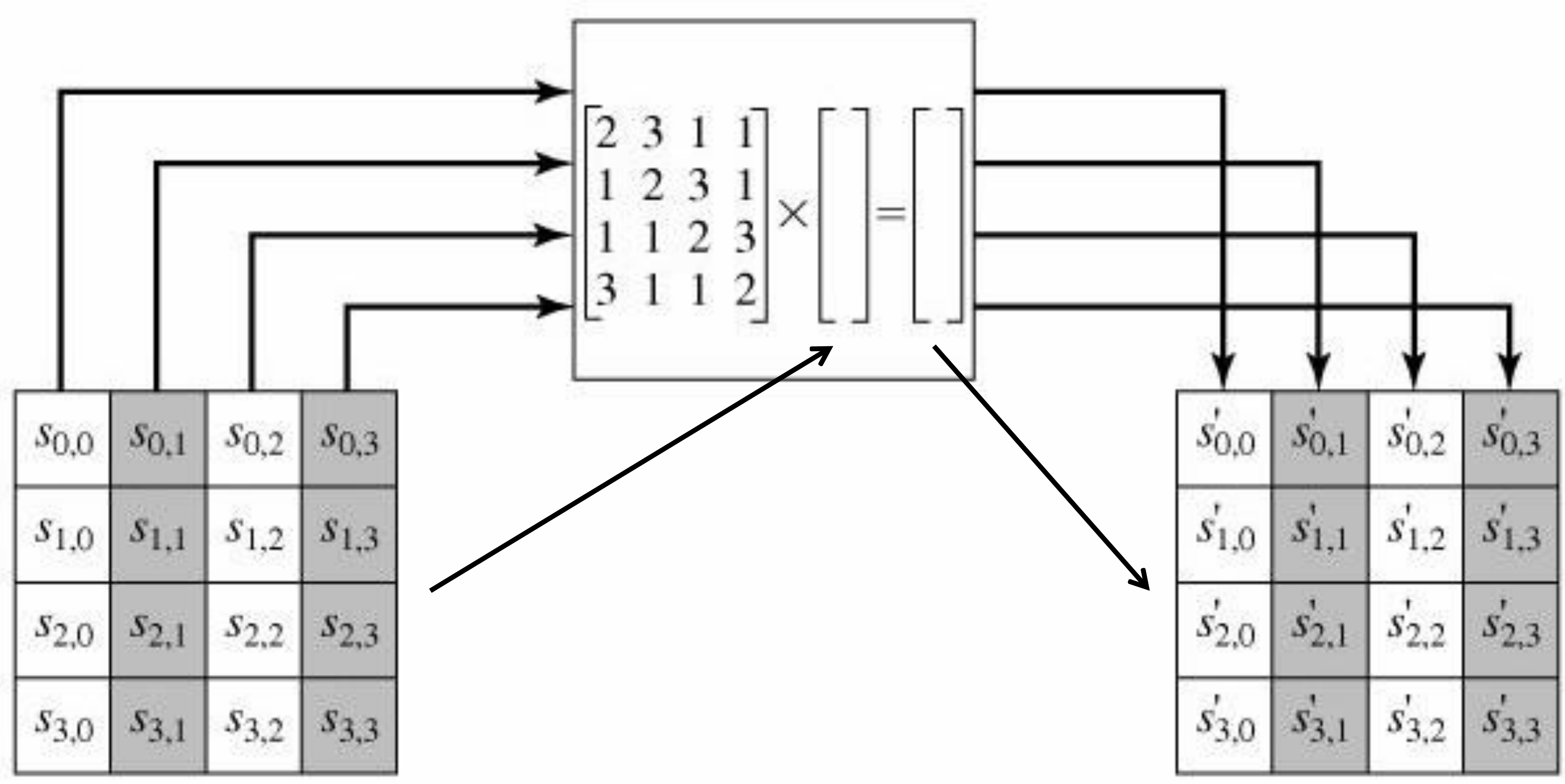
Shift Rows – Block Example

87	F2	4D	97
EC	6E	4C	90
4A	C3	46	E7
8C	D8	95	A6

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

Row	Shift for Encryption	Shift for Encryption
0	Circulate Left 0 Byte	Circulate Right 0 Byte
1	Circulate Left 1 Byte	Circulate Right 1 Byte
2	Circulate Left 2 Byte	Circulate Right 2 Byte
3	Circulate Left 3 Byte	Circulate Right 3 Byte

Mix Columns



$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \times \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix} = \begin{bmatrix} \\ \\ \\ \\ \end{bmatrix}$
$s'_{0,0}$ $s'_{0,1}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{1,1}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,1}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,1}$ $s'_{3,2}$ $s'_{3,3}$

Mix Columns

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

$$S'_{0,j} = (2 \cdot S_{0,j}) \oplus (3 \cdot S_{1,j}) \oplus S_{2,j} \oplus S_{3,j}$$
$$S'_{1,j} = S_{0,j} \oplus (2 \cdot S_{1,j}) \oplus (3 \cdot S_{2,j}) \oplus S_{3,j}$$
$$S'_{2,j} = S_{0,j} \oplus S_{1,j} \oplus (2 \cdot S_{2,j}) \oplus (3 \cdot S_{3,j})$$
$$S'_{3,j} = (3 \cdot S_{0,j}) \oplus S_{1,j} \oplus S_{2,j} \oplus (2 \cdot S_{3,j})$$

Multiplication Performed in GF(2^8)

Mix Columns Example

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \bullet$$

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

$$\begin{bmatrix} 02 & 03 & 01 & 01 \end{bmatrix} \times \begin{bmatrix} 87 \\ 6E \\ 46 \\ A6 \end{bmatrix} =$$

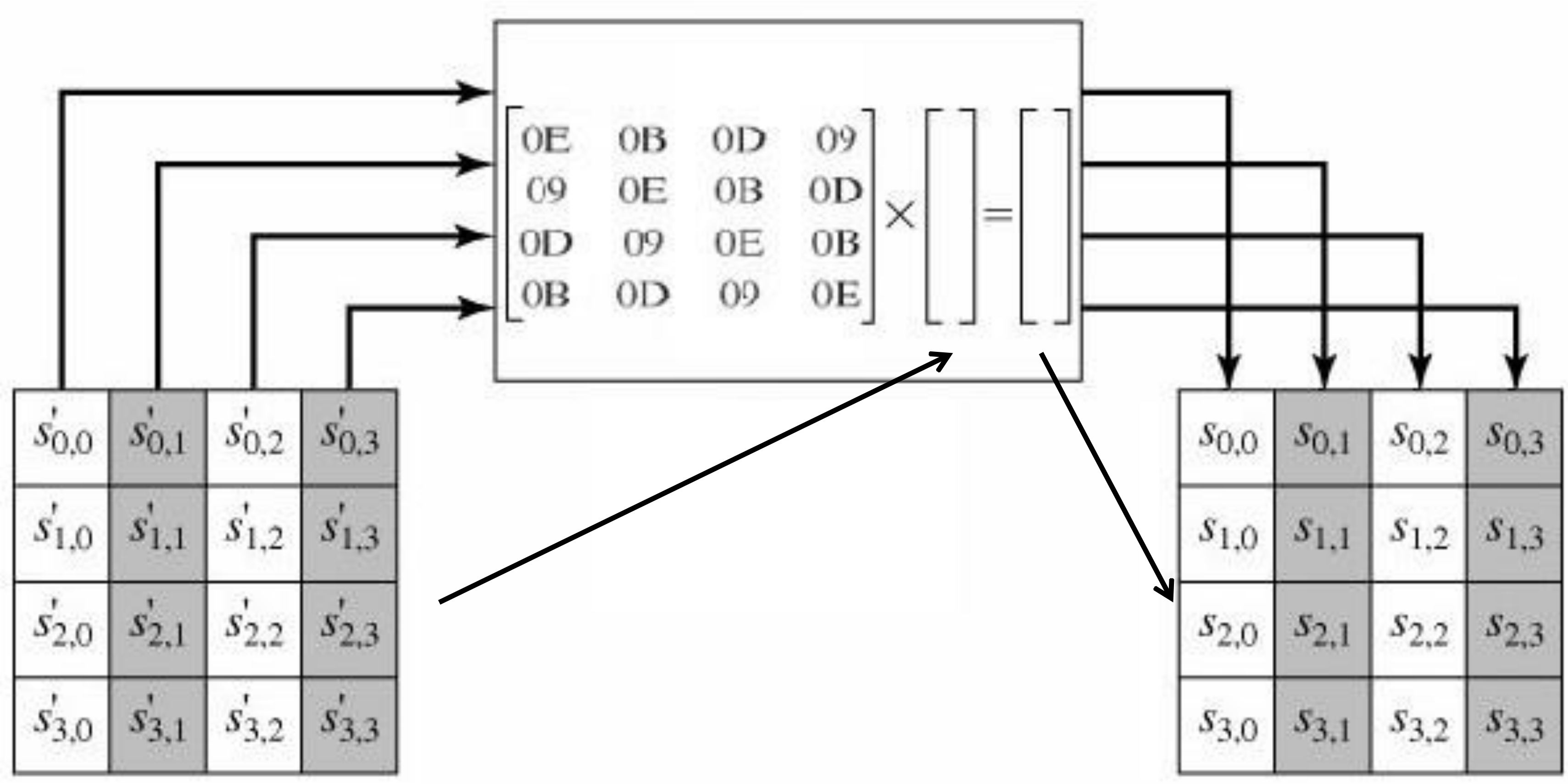
47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

$$= (02 \times 87) \oplus (03 \times 6E) \oplus (01 \times 46) \oplus (01 \times A6)$$

$$= 47$$

Multiplication Performed in GF(2^8)

Inverse Mix Columns



Inverse Mix Columns

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

Inverse Mix Columns Example

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \bullet$$

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

$$\begin{bmatrix} 0E & 0B & 0D & 09 \end{bmatrix} \times \begin{bmatrix} 47 \\ 37 \\ 94 \\ ED \end{bmatrix} =$$

87	F2	4D	97
6E	4C	90	EC
46	E7	4A	C3
A6	8C	D8	95

$$= (0E \times 47) \oplus (0B \times 37) \oplus (0D \times 94) \oplus (09 \times ED)$$

$$= 87$$

Multiplication Performed in GF(2^8)

Mix Column Transformations

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Add Round Key

$$
\begin{bmatrix}
s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\
s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\
s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\
s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3}
\end{bmatrix}
\oplus
\begin{bmatrix}
w_i & w_{i+1} & w_{i+2} & w_{i+3}
\end{bmatrix}
=
\begin{bmatrix}
s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\
s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\
s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\
s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3}
\end{bmatrix}
$$

47	40	A3	4C
37	D4	70	9F
94	E4	3A	42
ED	A5	A6	BC

$\oplus$

AC	19	28	57
77	FA	D1	5C
66	DC	29	00
F3	21	41	6A

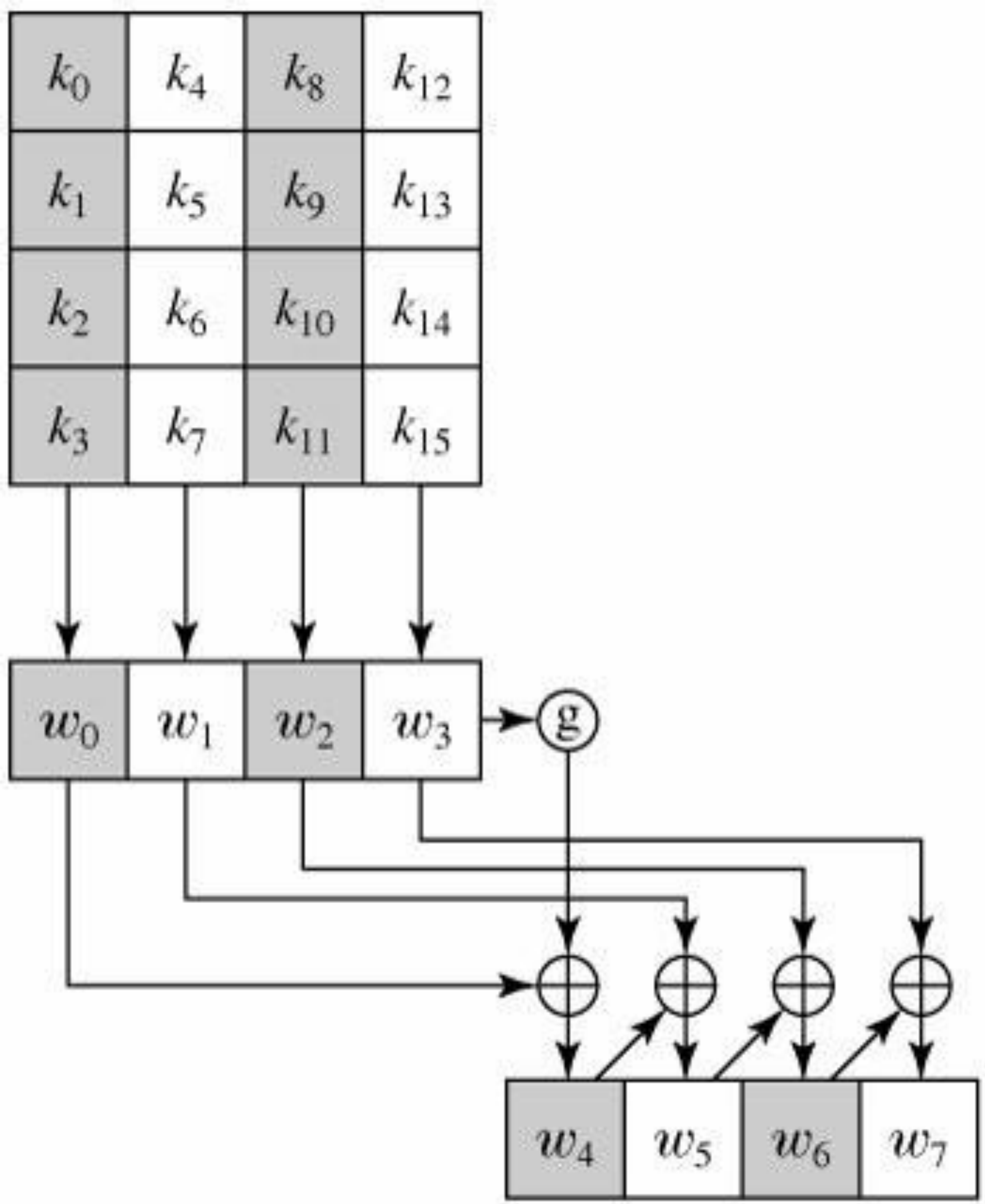
$=$

EB	59	8B	1B
40	2E	A1	C3
F2	38	13	42
1E	84	E7	D2

AES Key Expansion

Description

AES Key Expansion



AES Key Expansion

```
//key length 16
//n rounds 10
//n words = (10+1)*4 = 44
KeyExpansion (byte key[16], word w[44])
{
    word temp
    for (i = 0; i < 4; i++)
            w[i] = (key[4*i], key[4*i+1], key[4*i+2], key[4*i+3]);

    for (i = 4; i < 44; i++)
      {
            //temp = g(w[i - 1])
            temp = w[i - 1];
            if (i mod 4 = 0)
                  temp = SubWord (RotWord (temp)) xor Rcon[i/4];
            w[i] = w[i - 4] xor temp
      }
}
```

AES Key Expansion - g()

```
//temp = g(w[i - 1])
temp = w[i - 1];
if (i mod 4 = 0)
      temp = SubWord (RotWord (temp)) xor Rcon[i/4];
```

- RotWord: [b0, b1, b2, b3] →[b1, b2, b3, b0]
- SubWord: Byte Substitution using AES S-Box
- Rcon[j] = (RC[j], 0, 0, 0)

j	1	2	3	4	5	6	7	8	9	10
RC[j]	01	02	04	08	10	20	40	80	1B	36

Where RC[1] = 1 and RC[j] = RC[j] x 2 in GF(2^8)

AES Theoretical Topics

Description

AES Theoretical Topics

- AES S-Box Construction

AES S-Box Construction

1. Create Empty S-Box 16x16 Byte

2. Fill S-Box
   - row 0 with {00}→{0F}
   - …
   - row 15 with {F0}→{FF}

3. Replace Each Byte with its Multiplicative inverse in GF(2^8) (Consider 00→00)

AES S-Box Construction

4. Apply Following Formula to Each Byte Bit

$$b' = b_i \oplus b_{(i+4)\mathrm{mod}8} \oplus b_{(i+5)\mathrm{mod}8} \oplus b_{(i+6)\mathrm{mod}8} \oplus b_{(i+7)\mathrm{mod}8} \oplus c_i$$

$$c = (c_7 c_6 c_5 c_4 c_3 c_2 c_1 c_0) = (01100011)$$

Alternatively

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

AES S-Box Example

- {95}
- $\{95\}^{-1}$ in $GF(2^8)$ = {8A}

$$\begin{bmatrix} 1&0&0&0&1&1&1&1 \\ 1&1&0&0&0&1&1&1 \\ 1&1&1&0&0&0&1&1 \\ 1&1&1&1&0&0&0&1 \\ 1&1&1&1&1&0&0&0 \\ 0&1&1&1&1&1&0&0 \\ 0&0&1&1&1&1&1&0 \\ 0&0&0&1&1&1&1&1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \oplus \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \oplus \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

- Apply the Transformation to {8A} → {2A}

Inverse AES S-Box Construction

1. Create Empty S-Box 16x16 Byte

2. Fill S-Box
   - row 0 with {00}→{0F}
   - row 1 with {10}→{1F}
   - …
   - row 15 with {F0}→{FF}

Inverse AES S-Box Construction

3. Apply Following Formula to Each Byte Bit

$$b' = b_{(i+2)\bmod 8} \oplus b_{(i+5)\bmod 8} \oplus b_{(i+7)\bmod 8} \oplus d_i$$

$$d = (d_7 d_6 d_5 d_4 d_3 d_2 d_1 d_0) = (00000101)$$

Alternatively

$$\begin{bmatrix} b'_0 \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Inverse AES S-Box Construction

- Replace Each Byte with its Multiplicative inverse in $GF(2^8)$ (Consider 00→00)

S-Box Transformation

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

AES Implementation on 8 Bit Processors

AES Implementation on 8 Bit Processors

- Substitutes Bytes → **Bytes** Transformation using S-Box (16x16 **Byte**)
- Shift Rows → **Bytes** Transposition
- Add Round → **Bytes** XOR
- Mix Columns

→ Simple **Bytes** Shift

→ Condition

→ **Bytes** XOR

Mix Columns Simplification for 8 Bits Processors

$$S'_{0,j} = (2 \cdot S_{0,j}) \oplus (3 \cdot S_{1,j}) \oplus S_{2,j} \oplus S_{3,j}$$
$$S'_{1,j} = S_{0,j} \oplus (2 \cdot S_{1,j}) \oplus (3 \cdot S_{2,j}) \oplus S_{3,j}$$
$$S'_{2,j} = S_{0,j} \oplus S_{1,j} \oplus (2 \cdot S_{2,j}) \oplus (3 \cdot S_{3,j})$$
$$S'_{3,j} = (3 \cdot S_{0,j}) \oplus S_{1,j} \oplus S_{2,j} \oplus (2 \cdot S_{3,j})$$

Consider $(3 \cdot S_{i,j}) = (2 \cdot S_{i,j}) \oplus S_{i,j}$ and $S_{i,j} \oplus S_{i,j} = 0$

$$S'_{0,j} = (2 \cdot S_{0,j}) \oplus (2 \cdot S_{1,j}) \oplus S_{1,j} \oplus S_{2,j} \oplus S_{3,j} \quad \text{then add} \quad S_{0,j} \oplus S_{0,j}$$
$$S'_{1,j} = S_{0,j} \oplus (2 \cdot S_{1,j}) \oplus (2 \cdot S_{2,j}) \oplus S_{2,j} \oplus S_{3,j} \quad \text{then add} \quad S_{1,j} \oplus S_{1,j}$$
$$S'_{2,j} = S_{0,j} \oplus S_{1,j} \oplus (2 \cdot S_{2,j}) \oplus (2 \cdot S_{3,j}) \oplus S_{3,j} \quad \text{then add} \quad S_{2,j} \oplus S_{2,j}$$
$$S'_{3,j} = (2 \cdot S_{0,j}) \oplus S_{0,j} \oplus S_{1,j} \oplus S_{2,j} \oplus (2 \cdot S_{3,j}) \quad \text{then add} \quad S_{3,j} \oplus S_{3,j}$$

Mix Columns Simplification for 8 Bits Processors

$$\text{Define } Tmp = S_{0,j} \oplus S_{1,j} \oplus S_{2,j} \oplus S_{3,j}$$

$$S'_{0,j} = S_{0,j} \oplus Tmp \oplus \left[2 \cdot \left(S_{0,j} \oplus S_{1,j}\right)\right]$$
$$S'_{1,j} = S_{1,j} \oplus Tmp \oplus \left[2 \cdot \left(S_{1,j} \oplus S_{2,j}\right)\right]$$
$$S'_{2,j} = S_{2,j} \oplus Tmp \oplus \left[2 \cdot \left(S_{2,j} \oplus S_{3,j}\right)\right]$$
$$S'_{3,j} = S_{3,j} \oplus Tmp \oplus \left[2 \cdot \left(S_{3,j} \oplus S_{0,j}\right)\right]$$

$$\text{Define Lookup Table of 256 Entry} \Rightarrow X2[i] = (2 \cdot x)$$

$$S'_{0,j} = S_{0,j} \oplus Tmp \oplus X2\left[S_{0,j} \oplus S_{1,j}\right]$$
$$S'_{1,j} = S_{1,j} \oplus Tmp \oplus X2\left[S_{1,j} \oplus S_{2,j}\right]$$
$$S'_{2,j} = S_{2,j} \oplus Tmp \oplus X2\left[S_{2,j} \oplus S_{3,j}\right]$$
$$S'_{3,j} = S_{3,j} \oplus Tmp \oplus X2\left[S_{3,j} \oplus S_{0,j}\right]$$