

Additional notes to accompany AES specification paper

Note that in these additional notes, the (sub)section numbers refer to the appropriate (sub)section of the AES specification paper.

1. Notation and Conventions

1.1 Rijndael was designed to allow an input (plaintext) block size of 128, 160, 192, 224 or 256 bits. AES is Rijndael with an input (plaintext) block size of 128 bits. In our examples, we will assume that the input (plaintext) is 128 bits, the key is 128 bits and the output (ciphertext) is 128 bits.

1.2 A byte is 8 bits. We denote a byte as $\{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$. A byte represents an element of the finite field $\text{GF}(256)$. We will use three representations of a byte:

- bitstring, for example $\{01100011\}$;
- polynomial of degree at most 7, for example $x^6 + x^5 + x + 1$;
- 2-digit hex number, for example $\{63\}$.

To distinguish between decimal representations and 2-digit hex numbers, the hex numbers will be in curly brackets, for example $\{xy\}$.

Exercise 1 Determine the bitstring and polynomial representations of the field elements $\{57\}$ and $\{83\}$.

1.3 Rijndael operates on a state array. For AES it is a 4×4 array of bytes (usually written in hex form). (Look at the start of the example on page 21 of the paper to see how the input is read into the state array.)

1.4 The four bytes in each column of the state are sometimes thought of as a single 32-bit word.

2. Finite Field Operations

2.1 Addition of bytes (field elements) is as defined in the lecture notes as addition of polynomials over K :

- To add two bitstrings, perform an XOR operation bitwise, for example $\{01010111\} \oplus \{10000011\} = \{11010100\}$.
- To add two polynomials, add the corresponding coefficients modulo 2, for example $(x^6 + x^4 + x^2 + x + 1) \oplus (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2$.
- To add two 2-digit hex numbers, convert to bitstrings and perform XOR operation, for example $\{57\} \oplus \{83\} = \{d4\}$.

2.2 Multiplication of bytes (field elements) is as defined in our previous lecture, working modulo the irreducible polynomial $m(x) = x^8 + x^4 + x^3 + x + 1$. Note that since this polynomial has degree 8 it cannot be written as a single byte, so in the paper it is sometimes written as the bitstring $1\{00011011\}$ or as the hex number $1\{1b\}$.

Exercise 2 Determine the product $\{57\} \cdot \{83\}$.

2.3 Multiplication can also be performed by repeated shifts of bitstrings. Note that if you start with a polynomial that does not have an x^7 term, then multiplication by x corresponds to left-shifting the bitstring by one position.

Example $x^6 + x^4 + x^2 + x + 1$ corresponds to the bitstring $\{01010111\}$. Multiplying it by x gives $x^7 + x^5 + x^3 + x^2 + x$ which corresponds to the bitstring $\{10101110\}$.

Note that if your polynomial does have an x^7 term, then you need to reduce mod $m(x)$ after multiplying by x .

Example $\{57\} \cdot \{83\} = (x^6 + x^4 + x^2 + x + 1) \cdot (x^7 + x + 1)$ so we need to find (by shifting) $x^7\{57\}$, $x\{57\}$ and $1\{57\}$ and then XOR these three bitstrings together. See Table 1 on page 4 of the paper.

2.4 Multiplication can also be performed by adding powers of a generator. A generator of a finite field having n elements is an element g of the field for which the powers $\{g, g^2, g^3, g^4, \dots, g^{n-1}\}$ generate all $n - 1$ non-zero elements of the field. The field $\text{GF}(256)$ can be generated by $\{03\}$. Thus every non-zero element of the field can be written in polynomial form as $(x + 1)^{\text{power}}$. To multiply two field elements $a = g^\alpha$ and $b = g^\beta$, we get $a \cdot b = g^{\alpha+\beta}$, so we can use tables of powers to do multiplication. Note that here $\alpha + \beta$ is regular addition, not the \oplus operation defined for field elements. Since the integer 255 is $\{ff\}$ and $g^{\{ff\}} = \{01\}$ for any generator, if $\alpha + \beta$ results in an integer larger than 255, then 255 can be subtracted from the power before referring to Table 3, that is $g^{\alpha+\beta} = g^{\{ff\} + \{xy\}} = g^{\{xy\}}$.

Exercise 3 Determine the product $\{57\} \cdot \{83\}$ using Table 2 and Table 3.

We can also use the tables to find multiplicative inverses of field elements. Recall that for any generator g , we have $g^{\{ff\}} = \{01\}$, or in decimal form $g^{255} = 1$. Thus we have that $g^{\{x\}} \cdot g^{\{ff\} - \{x\}} = g^{\{ff\}} = \{01\}$ so the inverse of $g^{\{x\}}$ is $g^{\{ff\} - \{x\}}$ (where $\{x\}$ is a 2-digit hex number).

Exercise 4 Find the inverse of $\{19\}$.

You can check your answer to Exercise 4 by multiplying the polynomials $x^4 + x^3 + 1$ and $x^5 + x^4 + x^3 + x^2 + x + 1$.

2.5 We can define polynomials whose coefficients are elements of $\text{GF}(256)$ (rather than just elements of $\{0, 1\}$). Some parts of the cipher use these but we won't worry about them.

3. The Cipher

AES has the following general description:

Copy input into 4×4 state array.

Add initial round key to get the starting state.

Perform 9 rounds of the following:

Substitute Bytes

Shift Rows

Mix Columns

Add Round Key

Perform a final round of:

Substitute Bytes

Shift Rows

Add Round Key

Output final 4×4 state array.

3.1 The Substitute Bytes routine uses an S-box. Each byte $\{xy\}$ in the current state array is replaced by the value in row x , column y of Table 5 on page 8 of the paper. The values in the S-box are computed by finding the multiplicative inverse of $\{xy\}$ and then transforming that byte $\{b_7b_6b_5b_4b_3b_2b_1b_0\}$ according to the transformation given in matrix form as equation (3.1.2) on page 8 of the paper.

Example Determine the S-box replacement for $\{19\}$. The inverse of $\{19\}$ is $\{3f\}$ (see previous example). The bitstring representation of $\{3f\}$ is $\{00111111\} = \{b_7b_6b_5b_4b_3b_2b_1b_0\}$. Thus the new bit seven is calculated as

$$b'_7 = 1b_7 \oplus 1b_6 \oplus 1b_5 \oplus 1b_4 \oplus 1b_3 \oplus 0b_2 \oplus 0b_1 \oplus 0b_0 \oplus 0 = 0 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \oplus 0 = 1.$$

Continuing in this way we see that $\{19\}$ is replaced by $\{11010100\} = \{d4\}$.

See the example on page 21 for a full Substitute Bytes routine in round one of AES.

3.2 The Shift Rows routine for AES takes the current state array, shifts the second row to the left by 1 position (wrapping around), shifts the third row to the left by 2 positions (wrapping around), and shifts the fourth row to the left by 3 positions (wrapping around).

See the example on page 21 for a full Shift Rows routine in round one of AES.

3.3 The Mix Columns routine takes each column of the current state array and transforms it according to the transformation given in matrix form as equation (3.3.1) on page 9 of the paper.

Example Determine the effect of Mix Columns on the column $[\{d4\} \{bf\} \{5d\} \{30\}]^T$. According to equation (3.3.1) we must compute

$$\begin{aligned}
& \begin{bmatrix} \{02\} & \{01\} & \{01\} & \{03\} \\ \{03\} & \{02\} & \{01\} & \{01\} \\ \{01\} & \{03\} & \{02\} & \{01\} \\ \{01\} & \{01\} & \{03\} & \{02\} \end{bmatrix} \begin{bmatrix} \{30\} \\ \{5d\} \\ \{bf\} \\ \{d4\} \end{bmatrix} \\
= & \begin{bmatrix} x(x^5 + x^4) + 1(x^6 + x^4 + x^3 + x^2 + 1) + 1(x^7 + x^5 + x^4 + x^3 + x^2 + x + 1) + (x + 1)(x^7 + x^6 + x^4 + x^2) \\ (x + 1)(x^5 + x^4) + x(x^6 + x^4 + x^3 + x^2 + 1) + 1(x^7 + x^5 + x^4 + x^3 + x^2 + x + 1) + 1(x^7 + x^6 + x^4 + x^2) \\ 1(x^5 + x^4) + (x + 1)(x^6 + x^4 + x^3 + x^2 + 1) + x(x^7 + x^5 + x^4 + x^3 + x^2 + x + 1) + 1(x^7 + x^6 + x^4 + x^2) \\ 1(x^5 + x^4) + 1(x^6 + x^4 + x^3 + x^2 + 1) + (x + 1)(x^7 + x^5 + x^4 + x^3 + x^2 + x + 1) + x(x^7 + x^6 + x^4 + x^2) \end{bmatrix} \\
= & \begin{bmatrix} x^7 + x^6 + x^5 + x^2 + 1 \\ x^7 + 1 \\ x^6 + x^5 + x^2 + x \\ x^2 \end{bmatrix} \\
= & \begin{bmatrix} \{e5\} \\ \{81\} \\ \{66\} \\ \{04\} \end{bmatrix}
\end{aligned}$$

Thus the new column is $[\{04\} \{66\} \{81\} \{e5\}]^T$.

3.4 The Add Round Key takes the current state array and XORs each cell with the corresponding cell in the round key array.

Section 4 of the paper explains how to calculate the key schedule. We won't worry about it.