Chapter 4

FINITE FIELDS

RAPEEPORN PUNYAYUTTHAKARN
CISSP, CISA, OSSA
Introduction

- will now introduce finite fields
- of increasing importance in cryptography
  - AES, Elliptic Curve, IDEA, Public Key
- concern operations on “numbers”
  - where what constitutes a “number” and the type of operations varies considerably
- start with concepts of groups, rings, fields from abstract algebra
Groups, Rings, Fields
Group

- a set of elements or “numbers”
- with some operation whose result is also in the set (closure)
- obeys:
  - associative law: $(a \cdot b) \cdot c = a \cdot (b \cdot c)$
  - has identity $e$: $e \cdot a = a \cdot e = a$
  - has inverses $a^{-1}$: $a \cdot a^{-1} = e$
- if commutative $a \cdot b = b \cdot a$
  - then forms an abelian group
First example: the integers

One of the most familiar groups is the set of integers \( \mathbb{Z} \) which consists of the numbers

\[ ..., -4, -3, -2, -1, 0, 1, 2, 3, 4, ... \] \[ \textsuperscript{[3]} \]

The following properties of integer addition serve as a model for the abstract group axioms given in the definition below.

1. For any two integers \( a \) and \( b \), the sum \( a + b \) is also an integer. In other words, the process of adding integers two at a time can never yield a result that is not an integer. This property is known as \( \text{closure} \) under addition.

2. For all integers \( a, b \) and \( c \), \( (a + b) + c = a + (b + c) \). Expressed in words, adding \( a \) to \( b \) first, and then adding the result to \( c \) gives the same final result as adding \( a \) to the sum of \( b \) and \( c \), a property known as \( \text{associativity} \).

3. If \( a \) is any integer, then \( 0 + a = a + 0 = a \). Zero is called the \( \text{identity element} \) of addition because adding it to any integer returns the same integer.

4. For every integer \( a \), there is an integer \( b \) such that \( a + b = b + a = 0 \). The integer \( b \) is called the \( \text{inverse element} \) of the integer \( a \) and is denoted \( -a \).
Cyclic Group

- A cyclic group is a group all of whose elements are powers of a particular element.
- A group is cyclic if every element is a power of some fixed element.
  - \( b = a^k \) for some \( a \) and every \( b \) in group.
- \( a \) is said to be a generator of the group.
- Define exponentiation as repeated application of operator.
  - Example: \( a^3 = a \cdot a \cdot a \).
- And let identity be: \( e = a^0 \).
Ring

- a set of “numbers”
- with *two operations* (addition and multiplication) which form:
  - an abelian group with addition operation
  - and multiplication:
    - has closure
    - is associative
    - distributive over addition:
      - \( a (b+c) = ab + ac \)
      - \( (a+b) c = ac + bc \)
Ring

- If multiplication operation is commutative, it forms a **commutative ring**

**Integral domain** - If multiplication operation has
  - an identity and
  - no zero divisors
    - If \( ab = 0 \) then either \( a = 0 \), or \( b = 0 \)
Field

- a set of numbers
- with two operations
  - Addition
  - Multiplication
- is an *Integral Domain*
- has *Multiplication Inverse*
- Finite field – Field with a finite number of elements

*Real Number??, Complex Number??, Integer ???
Group, Ring, Field

(A1) Closure under addition:

If $a$ and $b$ belong to $S$, then $a + b$ is also in $S$

$\forall a, b \in S$, $a + b \in S$

(A2) Associativity of addition:

$a + (b + c) = (a + b) + c$ for all $a, b, c \in S$

(A3) Additive identity:

There is an element $0$ in $R$ such that

$a + 0 = 0 + a = a$ for all $a \in S$

(A4) Additive inverse:

For each $a \in S$ there is an element $-a$ in $S$ such that

$a + (-a) = (-a) + a = 0$

(A5) Commutativity of addition:

$a + b = b + a$ for all $a, b \in S$

(M1) Closure under multiplication:

If $a$ and $b$ belong to $S$, then $ab$ is also in $S$

(M2) Associativity of multiplication:

$a(bc) = (ab)c$ for all $a, b, c \in S$

(M3) Distributive laws:

$a(b + c) = ab + ac$ for all $a, b, c \in S$

$(a + b)c = ac + bc$ for all $a, b, c \in S$

(M4) Commutativity of multiplication:

$a b = b a$ for all $a, b \in S$

(M5) Multiplicative identity:

There is an element $1$ in $S$ such that

$al = la = a$ for all $a \in S$

(M6) No zero divisors:

If $a, b \in S$ and $ab = 0$, then either

$a = 0$ or $b = 0$

(M7) Multiplicative inverse:

If $a$ belongs to $S$ and $a \neq 0$, there is an element $a^{-1}$ in $S$ such that $aa^{-1} = a^{-1}a = 1$
Modular Arithmatic
Modular Arithmetic

- define **modulo operator** “\( a \mod n \)” to be remainder when \( a \) is divided by \( n \)
  - The remainder is called **residue** of \( a \mod n \)
- if \( b \) is a **residue** of \( a \mod n \)
  - we can always write: \( a = qn + b \)
  - usually chose smallest positive remainder as residue
    - ie. \( 0 \leq b \leq n-1 \)
  - process is known as **modulo reduction**
    - eg. \(-12 \mod 7 = -5 \mod 7 = 2 \mod 7\)
Modular Arithmetic

- use the term **congruence** for: \( a \equiv b \pmod{n} \)
  - *when divided by* \( n \), *\( a \) & \( b \) have same remainder*
  - *eg. 100 \( \equiv \) 34 (mod 11 )*

- **Note:** the “mod \( n \)” operator maps all integers into the set of integers
  \[
  \{ 0, 1, 2, \ldots, (n-1) \}
  \]
Divisors

- say a non-zero number $b$ divides $a$ if for some $m$ have $a = mb$ ($a$, $b$, $m$ all integers)
- that is $b$ divides into $a$ with no remainder
- denote this $b | a$
- and say that $b$ is a divisor of $a$
- eg. divisors of 24 are 1,2,3,4,6,8,12,24

- Note: if $a \equiv 0 \pmod{n}$ then $n | a$
Modular Arithmetic Operations

- is 'clock arithmetic'
  - the “mod n” operator maps all integers into the set of integers \{0, 1, 2, \ldots, (n-1)\}
  - arithmetic operations within the confines of the set
- uses a finite number of values, and loops back from either end
- modular arithmetic is when do addition & multiplication and modulo reduce answer
- can do reduction at any point, ie
  - \((a+b) \mod n = [a \mod n + b \mod n] \mod n\)
  - \((a\times b) \mod n = [a \mod n \times b \mod n] \mod n\)
Example – Modulo 8

(a) Addition modulo 8

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

(b) Multiplication modulo 8

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>1</td>
<td>4</td>
<td>7</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>5</td>
<td>2</td>
<td>7</td>
<td>4</td>
<td>1</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

(c) Additive and multiplicative inverses modulo 8

<table>
<thead>
<tr>
<th>w</th>
<th>(-w)</th>
<th>(w^{-1})</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>—</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>—</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>—</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>—</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>7</td>
</tr>
</tbody>
</table>
Properties of Modular Arithmetic

- Define the set of nonnegative integers less than n
  \[ Z_n = \{0, 1, \ldots, n-1\} \]

- Residue Classes

- Perform Modular Arithmetic within \( Z_n \), the followings hold
  - Closure (+, *)
  - Associative (+, *)
  - Identities (+, *)
  - Inverse (+)
  - Distributive
  - Commutative (+, *)
Modular Arithmetic

- $\mathbb{Z}_n$ form a commutative ring
- with a multiplicative identity
- note some peculiarities
  - if $(a+b) \equiv (a+c) \pmod{n}$
    then $b \equiv c \pmod{n}$
  - but if $(a\times b) \equiv (a\times c) \pmod{n}$
    then $b \equiv c \pmod{n}$

only if $a$ is relatively prime to $n$
To see this, consider an example in which the condition of Equation (4.3) does not hold. The integers 6 and 8 are not relatively prime, since they have the common factor 2. We have the following:

\[6 \times 3 = 18 \equiv 2 \pmod{8}\]

\[6 \times 7 = 42 \equiv 2 \pmod{8}\]

Yet \[3 \not\equiv 7 \pmod{8}\].
Modular Arithmetic

**Reason**
- The multiplier $a$ fails to produce the complete set of residues if $a$ and $n$ have any factors in common.

With $a = 6$ and $n = 8$,

<table>
<thead>
<tr>
<th>$Z_8$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiply by 6</td>
<td>0</td>
<td>6</td>
<td>12</td>
<td>18</td>
<td>24</td>
<td>30</td>
<td>36</td>
<td>42</td>
</tr>
<tr>
<td>Residues</td>
<td>0</td>
<td>6</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
</tbody>
</table>

Because we do not have a complete set of residues when multiplying by 6, more than one integer in $Z_8$ maps into the same residue. Specifically, $6 \times 0 \equiv 0 \mod 8$, $6 \times 1 \equiv 0 \mod 8$, $6 \times 2 \equiv 0 \mod 8$, and so on. Because this is a many-to-one mapping, there is not a unique inverse to the multiply operation.

However, if we take $a = 5$ and $n = 8$, whose only common factor is 1,

<table>
<thead>
<tr>
<th>$Z_8$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiply by 6</td>
<td>0</td>
<td>5</td>
<td>10</td>
<td>15</td>
<td>20</td>
<td>25</td>
<td>30</td>
<td>35</td>
</tr>
<tr>
<td>Residues</td>
<td>0</td>
<td>5</td>
<td>2</td>
<td>7</td>
<td>4</td>
<td>1</td>
<td>6</td>
<td>3</td>
</tr>
</tbody>
</table>

The line of residues contains all the integers in $Z_8$, in a different order.
Euclidean Algorithm
Greatest Common Divisor (GCD)

- Euclidean Algorithm – Determining GCD
- GCD (a, b) of a and b is the largest number that divides evenly into both a and b
  - eg GCD(60, 24) = 12
- often want **no common factors** (except 1) and hence numbers are **relatively prime**
  - eg GCD(8, 15) = 1
  - hence 8 & 15 are relatively prime
Euclidean Algorithm

- an efficient way to find the GCD(a,b)
- uses theorem that:
  - \( \text{GCD}(a, b) = \text{GCD}(b, a \mod b) \)
- Euclidean Algorithm to compute GCD(a,b) is:

  EUCLID(a,b)
  1. \( A = a; B = b \)
  2. if \( B = 0 \) return \( A = \text{gcd}(a, b) \)
  3. \( R = A \mod B \)
  4. \( A = B \)
  5. \( B = R \)
  6. goto 2
Example GCD(1970, 1066)

1970 = 1 \times 1066 + 904
gcd(1066, 904)

1066 = 1 \times 904 + 162
gcd(904, 162)

904 = 5 \times 162 + 94
gcd(162, 94)

162 = 1 \times 94 + 68
gcd(94, 68)

94 = 1 \times 68 + 26
gcd(68, 26)

68 = 2 \times 26 + 16
gcd(26, 16)

26 = 1 \times 16 + 10
gcd(16, 10)

16 = 1 \times 10 + 6
gcd(10, 6)

10 = 1 \times 6 + 4
gcd(6, 4)

6 = 1 \times 4 + 2
gcd(4, 2)

4 = 2 \times 2 + 0
gcd(2, 0)
Galois Field - GF(p)
Galois Fields

- Finite Field – Field with a finite number of elements
- finite fields play a key role in cryptography
- can show number of elements in a finite field must be a power of a prime \( p^n \)
- known as Galois fields
- denoted \( \text{GF}(p^n) \)
- in particular often use the fields:
  - \( \text{GF}(p) \)
  - \( \text{GF}(2^n) \)
Galois Fields GF(p)

- GF(p) is the set of integers \{0,1, \ldots, p-1\} with arithmetic operations modulo prime p
- these form a finite field
  - since have multiplicative inverses
  - why???
- hence arithmetic is “well-behaved” and can do addition, subtraction, multiplication, and division without leaving the field GF(p)

- \((a*b) \equiv (a*c) \pmod{p}\) then \(b \equiv c \pmod{p}\)
GF(7) Example

(a) Addition modulo 7

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

(b) Multiplication modulo 7

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>1</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>2</td>
<td>5</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>2</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

(c) Additive and multiplicative inverses modulo 7

<table>
<thead>
<tr>
<th></th>
<th>w</th>
<th>-w</th>
<th>w⁻¹</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>6</td>
<td>6</td>
</tr>
</tbody>
</table>
Finding Inverses

EXTENDED EUCLID($m$, $b$)

1. $(A_1, A_2, A_3) = (1, 0, m)$;
   $(B_1, B_2, B_3) = (0, 1, b)$

2. if $B_3 = 0$
   return $A_3 = \gcd(m, b)$; no inverse

3. if $B_3 = 1$
   return $B_3 = \gcd(m, b)$; $B_2 = b^{-1} \mod m$

4. $Q = A_3 \div B_3$

5. $(T_1, T_2, T_3) = (A_1 - Q B_1, A_2 - Q B_2, A_3 - Q B_3)$

6. $(A_1, A_2, A_3) = (B_1, B_2, B_3)$

7. $(B_1, B_2, B_3) = (T_1, T_2, T_3)$

8. goto 2
Inverse of 550 in GF(1759)

<table>
<thead>
<tr>
<th>Q</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
</tr>
</thead>
<tbody>
<tr>
<td>—</td>
<td>1</td>
<td>0</td>
<td>1759</td>
<td>0</td>
<td>1</td>
<td>550</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>550</td>
<td>1</td>
<td>–3</td>
<td>109</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>–3</td>
<td>109</td>
<td>–5</td>
<td>16</td>
<td>5</td>
</tr>
<tr>
<td>21</td>
<td>–5</td>
<td>16</td>
<td>5</td>
<td>106</td>
<td>–339</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>106</td>
<td>–339</td>
<td>4</td>
<td>–111</td>
<td>355</td>
<td>1</td>
</tr>
</tbody>
</table>
Polynomial Arithmetic
Polynomial Arithmetic

- A polynomial degree \( n \)

\[ f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 = \sum a_i x^i \]
  - not interested in any specific value of \( x \)
  - which is known as the indeterminate

- several alternatives/classes available
  - ordinary polynomial arithmetic
  - poly arithmetic with coefficients performed mod \( p \)
  - poly arithmetic with coefficients performed mod \( p \) and polynomials mod \( m(x) \)
Ordinary Polynomial

- add or subtract corresponding coefficients
- multiply all terms by each other
- eg
  
  let $f(x) = x^3 + x^2 + 2$ and $g(x) = x^2 - x + 1$
  $f(x) + g(x) = x^3 + 2x^2 - x + 3$
  $f(x) - g(x) = x^3 + x + 1$
  $f(x) \times g(x) = x^5 + 3x^2 - 2x + 2$
Polynomial Arithmetic with Modulo Coefficients

- Polynomial which coef are elements of filed
  - forms a polynomial ring
- could be modulo any prime
- but we are most interested in mod 2
  - ie all coefficients are 0 or 1
  - eg. let $f(x) = x^3 + x^2$ and $g(x) = x^2 + x + 1$
    - $f(x) + g(x) = x^3 + x + 1$
    - $f(x) \times g(x) = x^5 + x^2$
Polynomial Division

- can write any polynomial in the form: $f(x) = q(x)g(x) + r(x)$
  - can interpret $r(x)$ as being a remainder
  - $r(x) = f(x) \mod g(x)$
- if have no remainder say $g(x)$ divides $f(x)$
- if $g(x)$ has no divisors other than itself & 1 say it is **irreducible** (or **prime polynomial**)
- arithmetic modulo an irreducible polynomial forms a field
Polynomial GCD

- can find greatest common divisor for polys
  - \( c(x) = \text{GCD}(a(x), b(x)) \) if \( c(x) \) is the poly of greatest degree which divides both \( a(x), b(x) \)
- can adapt Euclid’s Algorithm to find it:

  EUCLID\([a(x), b(x)]\]

1. \( A(x) = a(x); B(x) = b(x) \)
2. if \( B(x) = 0 \) return \( A(x) = \text{gcd}[a(x), b(x)] \)
3. \( R(x) = A(x) \mod B(x) \)
4. \( A(x) \div B(x) \)
5. \( B(x) \div R(x) \)
6. goto 2
Finite Field – GF($2^n$)
Finite Field – GF($2^n$)

- Motivation - Need to work on a field
  - In case the encryption algo. needs division operation
- Motivation – Would like to work with integers that fit exactly into a given number of bits

- EX: 8-bits Data => integers from 0–255
- $2^8 = 256$ => not a prime so $\mathbb{Z}_{256}$ is not a field
- The closet prime number is 251 => Operate on $\mathbb{Z}_{251}$
- Waste 251–255
Finite Field – GF($2^n$)

- **Motivation** – the occurrences of non-zero integers is not uniform
  - Weak for encryption algorithm

<table>
<thead>
<tr>
<th>Integer</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Occurrences in $\mathbb{Z}_8$</td>
<td>4</td>
<td>8</td>
<td>4</td>
<td>12</td>
<td>4</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>Occurrences in GF($2^3$)</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
</tbody>
</table>

- Finding a Filed in form $2^n$
**GF(2^3)**

### (a) Addition

<table>
<thead>
<tr>
<th>+</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>010</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>7</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>011</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>100</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>101</td>
<td>5</td>
<td>4</td>
<td>7</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>110</td>
<td>6</td>
<td>7</td>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

### (b) Multiplication

<table>
<thead>
<tr>
<th>×</th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>001</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>010</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>3</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>011</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>5</td>
<td>7</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>100</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>3</td>
<td>7</td>
<td>6</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>101</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>4</td>
<td>2</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>110</td>
<td>6</td>
<td>0</td>
<td>6</td>
<td>7</td>
<td>1</td>
<td>5</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>111</td>
<td>7</td>
<td>0</td>
<td>7</td>
<td>5</td>
<td>2</td>
<td>1</td>
<td>6</td>
<td>4</td>
</tr>
</tbody>
</table>

### (c) Additive and multiplicative inverses

<table>
<thead>
<tr>
<th>w</th>
<th>-w</th>
<th>w^{-1}</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>--</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>4</td>
</tr>
</tbody>
</table>
Modular Polynomial Arithmetic

- can compute in field GF(2^n)
  - polynomials with coefficients modulo 2
  - whose degree is less than n
  - hence must reduce modulo an irreducible poly of degree n (for multiplication only)
- form a finite field
- can always find an inverse
  - can extend Euclid’s Inverse algorithm to find
Construct $\text{GF}(2^3)$

- Need to choose irreducible polynomial of degree 3
  - $x^3 + x + 1$
Construct \(GF(2^3)\)

### Table 4.6 Polynomial Arithmetic Modulo \((x^3 + x + 1)\)

#### (a) Addition

<table>
<thead>
<tr>
<th></th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>(x^2)</td>
<td>(x^2 + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
</tr>
<tr>
<td>1</td>
<td>(x + 1)</td>
<td>0</td>
<td>(x)</td>
<td>(x^2 + 1)</td>
<td>(x^2)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + 1)</td>
</tr>
<tr>
<td>(x)</td>
<td>(x + 1)</td>
<td>0</td>
<td>1</td>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2)</td>
<td>(x^2 + 1)</td>
<td>(x^2 + x)</td>
</tr>
<tr>
<td>(x + 1)</td>
<td>(x)</td>
<td>1</td>
<td>0</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2)</td>
<td>(x^2 + 1)</td>
<td>1</td>
</tr>
<tr>
<td>(x^2)</td>
<td>(x^2 + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
<td>0</td>
<td>1</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>(x)</td>
</tr>
<tr>
<td>(x^2 + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + x)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>(x)</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>(x^2 + x + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>(x)</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

#### (b) Multiplication

<table>
<thead>
<tr>
<th></th>
<th>000</th>
<th>001</th>
<th>010</th>
<th>011</th>
<th>100</th>
<th>101</th>
<th>110</th>
<th>111</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(x)</td>
<td>0</td>
<td>(x^2)</td>
<td>(x^2 + x)</td>
<td>(x + 1)</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>(x + 1)</td>
<td>1</td>
<td>(x^2)</td>
<td>(x^2 + x)</td>
<td>(x + 1)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
</tr>
<tr>
<td>(x)</td>
<td>0</td>
<td>(x)</td>
<td>(x^2)</td>
<td>(x^2 + x)</td>
<td>(x + 1)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
</tr>
<tr>
<td>(x + 1)</td>
<td>(x + 1)</td>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2)</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
<td>(x)</td>
<td></td>
</tr>
<tr>
<td>(x^2)</td>
<td>0</td>
<td>(x^2)</td>
<td>(x + 1)</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + x)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>1</td>
</tr>
<tr>
<td>(x^2 + 1)</td>
<td>0</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>(x^2 + x + 1)</td>
<td>(x + 1)</td>
<td>(x^2 + x)</td>
<td></td>
</tr>
<tr>
<td>(x^2 + x)</td>
<td>0</td>
<td>(x^2 + x)</td>
<td>(x^2 + x + 1)</td>
<td>1</td>
<td>(x^2 + 1)</td>
<td>(x + 1)</td>
<td>(x)</td>
<td>(x^2)</td>
</tr>
<tr>
<td>(x^2 + x + 1)</td>
<td>0</td>
<td>(x^2 + x + 1)</td>
<td>(x^2 + 1)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>(x)</td>
<td>(x + 1)</td>
<td>1</td>
</tr>
</tbody>
</table>
Computational Considerations

- since coefficients are 0 or 1, can represent any such polynomial as a bit string
- addition becomes XOR of these bit strings
- multiplication is shift & XOR
  - cf long-hand multiplication
- modulo reduction done by repeatedly substituting highest power with remainder of irreducible poly (also shift & XOR)
in $\text{GF}(2^3)$ have $(x^2+1)$ is $101_2$ & $(x^2+x+1)$ is $111_2$

so addition is
- $(x^2+1) + (x^2+x+1) = x$
- $101 \ XOR \ 111 = 010_2$

and multiplication is
- $(x+1).(x^2+1) = x.(x^2+1) + 1.(x^2+1)$
  $= x^3+x+x^2+1 = x^3+x^2+x+1$
- $011.101 = (101)<<1 \ XOR \ (101)<<0 = 1010 \ XOR \ 101 = 1111_2$

polynomial modulo reduction (get $q(x)$ & $r(x)$) is
- $(x^3+x^2+x+1) \ mod \ (x^3+x+1) = 1.(x^3+x+1) + (x^2) = x^2$
- $1111 \ mod \ 1011 = 1111 \ XOR \ 1011 = 0100_2$
Using a Generator

- equivalent definition of a finite field
- a **generator** $g$ is an element whose powers generate all non-zero elements
  - in $F$ have $0, g^0, g^1, ..., g^{q-2}$
- can create generator from **root** of the irreducible polynomial, $f(x)$
  - $g$ must satisfy $f(x) = 0$
- then implement multiplication by adding exponents of generator
Generator for GF(2^3)

- \( f(x) = x^3 + x + 1 \)
- \( f(g) = g^3 + g + 1 = 0 \) \( \Rightarrow \) \( g^3 = g + 1 \)
- \( g = g \)
- \( g1 = g1 \)
- \( g2 = g2 \)
- \( g3 = g + 1 \)

\[
\begin{align*}
g^4 &= g(g^3) = g(g + 1) = g^2 + g \\cr
g^5 &= g(g^4) = g(g^2 + g) = g^3 + g^2 = g^2 + g + 1 \\cr
g^6 &= g(g^5) = g(g^2 + g + 1) = g^3 + g^2 + g = g^2 + g + g + 1 = g^2 + 1 \\cr
g^7 &= g(g^6) = g(g^2 + 1) = g^3 + g = g + g + 1 = 1 = g^0
\end{align*}
\]
Generator for $\text{GF}(2^3)$

<table>
<thead>
<tr>
<th>Power Representation</th>
<th>Polynomial Representation</th>
<th>Binary Representation</th>
<th>Decimal (Hex) Representation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>000</td>
<td>0</td>
</tr>
<tr>
<td>$g^0$ ( = $g^7$)</td>
<td>1</td>
<td>001</td>
<td>1</td>
</tr>
<tr>
<td>$g^1$</td>
<td>$g$</td>
<td>010</td>
<td>2</td>
</tr>
<tr>
<td>$g^2$</td>
<td>$g^2$</td>
<td>100</td>
<td>4</td>
</tr>
<tr>
<td>$g^3$</td>
<td>$g + 1$</td>
<td>011</td>
<td>3</td>
</tr>
<tr>
<td>$g^4$</td>
<td>$g^2 + g$</td>
<td>110</td>
<td>6</td>
</tr>
<tr>
<td>$g^5$</td>
<td>$g^2 + g + 1$</td>
<td>111</td>
<td>7</td>
</tr>
<tr>
<td>$g^6$</td>
<td>$g^2 + 1$</td>
<td>101</td>
<td>5</td>
</tr>
</tbody>
</table>
This makes multiplication easier

To multiply, add the power modulo 7

\[ x^4 + x^6 = x^{(10 \mod 7)} = x^3 = x+1 \]

Construct GF(2^3) table again!
have considered:

- concept of groups, rings, fields
- modular arithmetic with integers
- Euclid’s algorithm for GCD
- finite fields GF(p)
- polynomial arithmetic in general and in GF(2^n)