**Lecture 3a: A Ring on Irreducible Polynomials**

**LEARNING OUTCOME**

**By the end of the lesson the student will be able to:**

1. understand a concept of irreducible polynomial
2. compute a greatest common divisor between two polynomials via Euclidean algorithm
3. compute an inverse of polynomial a modulo irreducible polynomial b via an extended Euclidean algorithm.

A prime number is divisible by another positive integer. It cannot be factored into more than one integers greater than 1.

The prime number list is 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, … and so on.

Let us see a polynomial over finite field modulo 2, F2.

Let us take a number,

P(*x*) = *an⋅ xn* + …+ *a*2*x*2 + *a*1*x*1 + *a*0*x*0 written in little endian.

A mathematical programmer will see the number in terms of big endian.

P(*x*) = *a*0*x*0+ *a*1*x*1+ *a*2*x*2 +…+*an⋅ xn* in an array [*a*0, *a*1, *a*2,…, *an*].

An index number system(ins):

Let us take a number, A = 847510 = 8*⋅* 103 + 4*⋅* 102 + 7⋅101 + 5⋅100 = [8 4 7 5].

B = 936210 = 9*⋅* 103 + 3*⋅* 102 + 6⋅101 + 2⋅100 = [9 3 6 2].

Let us add A + B = 17*⋅* 103 + 7*⋅* 102 + 13⋅101 + 7⋅100 = [17 7 13 7].

= 1*⋅* 104 + 7*⋅* 103 + 8*⋅* 102 + 3⋅101 + 7⋅100 = [1 7 8 3 7].

This is called a carry problem. It is an inefficient process. Starting from 1980’s, a polynomial in finite field has been introduced in cryptography.

A finite filed F2 is an *n*-bit polynomial

P(*x*) = *an*−1*⋅ xn*−1 + …+ *a*2*x*2 + *a*1*x*1 + *a*0*x*0 = [*an−*1,…,*a*2, *a*1, *a*0] written in little endian modulo 2.

|  |  |  |
| --- | --- | --- |
| *i* | Binary | Polynomial |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 2 | 10 | *x* |
| 3 | 11 | *x* +1 |
| 4 | 100 | *x*2 |
| 5 | 101 | *x*2+1 |
| 6 | 110 | *x*2+*x* |
| 7 | 111 | *x*2+*x*+1 |
| 8 | 1000 | *x*3 |
| 9 | 1001 | *x*3+1 |
| 10 | 1010 | *x*3+*x* |
| 11 | 1011 | *x*3+*x*+1 |
| 12 | 1100 | *x*3+ *x*2 |

Let us take a number, A = 29910 = [100101011]= *x*8 + *x*5 + *x*3 + *x* + 1.

B = 28310 = [100011011]= *x*8 + *x*4 + *x*3 + *x* + 1.

Let us add A + B = = [200112022]= 2*x*8 + *x*5 + *x*4 + 2*x*3 + 2*x* + 2.

= [ 110000]= *x*5 + *x*4 (mod 2)

Similarly, an irreducible polynomial cannot be factored into the product of two polynomials. The property of irreducibility depends on the field or ring to which the coefficients are considered to belong.

A ring is a group with operations addition and multiplication. There must be also an identity for each operation. Traditionally, an identity for addition is a point zero. And an identity for multiplication is one.

Let *x* be a member of ring F2, *x* ∈ F2. An additive identity *e* ∈ F2 so that

*e + x = x + e = x*.

An multiplicative identity *i* ∈ F2 so that *i \* x = x \* i = x*. A good counter example we have seen in ECC, the identity is a point at infinity (+∞, +∞).

Then there is an additive inverse (*−x*) *+ x = x +* (*−x*) *= e*. And there is a multiplicative inverse

*x*−1⋅  *x = x ⋅ x*−1 *= i*.

A ring modulo irreducible polynomial over an integer has been practically popular since 1985.

It has been used in several modern cryptosystems,

1. ECC (1985), 2. AES(2000), and 3. NTRU(2005).

We are in F2. Let us review *x*2 − 1 = (*x*+1)⋅(*x*−1) ≡ *x*2 + 1 is not an irreducible polynomial.

Let us take another example  Let 

≡ *x*5 + 1

Let us see  = 



=  is not an irreducible polynomial.

Today, a ring modulo irreducible polynomial over positive integer coefficients is useful when an addition operation is taken as an exclusive-or.

Now, let us see how to generate the irreducible polynomial.

Prime versus Irreducible Polynomial in F2

|  |  |  |
| --- | --- | --- |
| *i* | Binary | Polynomial |
| 0 | 0 | 0 |
| 1 | 1 | 1 |
| 2 | 10 | *x* |
| 3 | 11 | *x* +1 |
| 4 | 100 | *x*2 |
| 5 | 101 | *x*2+1 |
| 6 | 110 | *x*2+*x* |
| 7 | 111 | *x*2+*x*+1 |
| 8 | 1000 | *x*3 |
| 9 | 1001 | *x*3+1 |
| 10 | 1010 | *x*3+*x* |
| 11 | 1011 | *x*3+*x*+1 |
| 12 | 1100 | *x*3+ *x*2 |
| 13 | 1101 | *x*3+ *x*2 +1 |

In this case of 5, *x*2+1 is divisible by *x*+1. Since *x*2+1=(*x*+1)(*x*−1)= (*x*+1)(*x*+1)

**A Ring in Finite Field**

A ring consists of 2 operations, addition and multiplication.

The addition operation is exclusive-or.

*a*(*x*) = *x*3+*x*+1 and b(*x*) = *x*2+*x*+1

*a* = 1011 and b = 111

*a*(*x*) + b(*x*) = *x*3+*x*+1 + *x*2+*x*+1 *= x*3+ *x*2

*a* ⊕ b = 1011+111 =1100

111

1100

Let us see multiplication. In the future, you will see a concept convolution for multiplication. We want to set a multiplier on the left with smaller number of ones. On the right we will set a longer string with more ones. This abelian operation,

*a* ⊗ b = 1011⊗111

= 111⊗1011

= 1011

1011

1011

= 112221 =110001= *x*5 + *x*4 +1.

Let us take an example from S-box AES.

Let an irreducible polynomial *m* = 28310 = 256+16+8+2+1=1000110112

In polynomial *m*(*x*) = *x*8+ *x*4+ *x*3+ *x*+1.

Let review and take an element *a* = 42= 32+8+2 = 1010102, *a*(*x*) = *x*5+ *x*3+ *x*.

We want to compute *a*−1(*x*) mod *m*(*x*).

Let us review on how to compute greatest common divisor(gcd) in integer division

We write and express *b* = *a*⋅*q* + r.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Euclidean Algorithm | | |  | |
| *b* = | *a*⋅ | *q* + | *r* |
| 283 | 42 | 6 | 31 |
| 42 | 31 | 1 | 11 |
| 31 | 11 | 2 | 9 |
| 11 | 9 | 1 | 2 |
| 9 | 2 | 4 | 1 |
| 2 | 1 | 2 | 0 |

The answer is g = gcd(42,283) =1. When the gcd is 1, they are linearly independent.

42 and 283 are relatively prime or coprime. The element 42 is invertible modulo 283.

There is an inverse.

Let us see the Extended Euclidean Algorithm

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Euclidean Algorithm | | |  | Extended |  |  |
| *b* = | *a* ⋅ | *q* + | *r* | *u* | *v* | *w*=*u*−*v⋅q* |
| 283 | 42 | 6 | 31 | 0 | 1 | -6 |
| 42 | 31 | 1 | 11 | 1 | -6 | 7 |
| 31 | 11 | 2 | 9 | -6 | 7 | -20 |
| 11 | 9 | 1 | 2 | 7 | -20 | 27 |
| 9 | 2 | 4 | 1 | -20 | 27 | -128 |
| 2 | 1 | 2 | 0 | 27 | -128 | 283 |

*a*1  mod b = −128 +283 = 155.

We always check *a* ⋅ *a*1 ≡ 1 (mod b)

42⋅155 = 6510 = 23⋅283 +1 ≡ 1 (mod 283)

Let us go into another domain in finite field F2.

Let *a*(*x*) and *b*(*x*) be the polynomials respectively. In binary, *a* = 42 = 32+8+2 = 101010 = *x*5+*x*3+*x*

And *b* = 283 = 256 +16+8+2+1=100011011 = *x*8+*x*4+*x*3+*x*+1.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Line | An Extended Euclidean Algorithm | | | |  |  | |  |  |
| 0 | *b* = | *a* ⋅ | *q* + | *r* | *u* | | *v* | | *w*=*u*−*v*⋅*q* |
| 1 | 100011011 | 101010 | 1010 | 11111 | 0 | | 1 | | 1010 |
| 2 | 101010 | 11111 | 11 | 1011 | 1 | | 1010 | | 11111 |
| 3 | 11111 | 1011 | 11 | 10 | 1010 | | 11111 | | 101011 |
| 4 | 1011 | 10 | 101 | 1 | 11111 | | 101011 | | 10011000 |
| 5 |  |  |  |  |  | |  | |  |
| 6 |  |  |  |  |  | |  | |  |

Line 1: 100011011 quotient

**1**01010 **1**000

1001011

**1**01010 10**1**0

11111

Let us go back to school:



|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Line | An Extended Euclidean Algorithm | | | |  | |  | |  |  |
| 0 | *b* = | *a* ⋅ | *q* + | *r* | | *u* | | *v* | | *w*=*u*−*v*⋅*q* |
| 1 | 100011011 | 101010 | 1010 | 11111 | | 0 | | 1 | | 1010 |
| 2 | 101010 | 11111 | 11 | 1011 | | 1 | | 1010 | | 11111 |
| 3 | 11111 | 1011 | 11 | 10 | | 1010 | | 11111 | | 101011 |
| 4 | 1011 | 10 | 101 | 1 | | 11111 | | 101011 | | 10011000 |
| 5 |  |  |  |  | |  | |  | |  |
| 6 |  |  |  |  | |  | |  | |  |

Line 2: 101010 quotient vq = 11\*1010 u-vq =11110

11111 10 = 1010 1

10100 1010 =11111

11111 11 11110

1011

Line 3: 11111 quotient vq = 11111\*11 u-vq =100001

1011 10 11111 1010

1001 11111 101011

1011 11 100001

10

Line 4: 1011 quotient vq = 101011\*101 u-vq =10000111

10 100 101011 11111

11 101011 =10011000

10 101 10000111

1

Let us check *a*(*x*) ⋅ *a*−1(*x*) = 1 mod *m*(*x*)

101010\*10011000

= 10011000

10011000

100110000

1011011110000 mod 100011011

100011011

11101000000

100011011

1100101100

100011011

100011010

100011011

1

Yes, *a*−1(*x*) = *x*7 + *x*4 + *x*3 is in fact an inverse of *a*(*x*) = *x*5+*x*3+*x*. Now, let us continue with the S-box.

From the *a* = 4210 = 1010102 = 2A16,

then *a*−1 = 15210 =100110002 = 9816.

This inverse will be plugged into an affine transform matrix.

One more time,

Let *a*(*x*) and *b*(*x*) be the polynomials respectively. In binary, *a* = 4310 = 32+8+2+1 = 101011 = *x*5+*x*3+*x*+1

And b = 28310 = 256 +16+8+2+1=100011011 = *x*8+*x*4+*x*3+*x*+1.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Line | Euclidean Algorithm | | | |  | |  | |  |  |
| 0 | b= | a\* | q+ | r | | u | | v | | w = u−vq |
| 1 | 100011011 | 101011 | 1010 | 10101 | | 0 | | 1 | | 1010 |
| 2 | 101011 | 10101 | 10 | 1 | | 1 | | 1010 | | 10101 |
| 3 |  |  |  |  | |  | |  | |  |
| 4 |  |  |  |  | |  | |  | |  |

Line 1:

1010 quotient

100011011

101011 1000

1000011

101011 1010

10101

Line 2: w=u-vq =1 – (*x*3+ *x*)⋅*x* = 1 + (*x*4 + *x*2)

10 = 1-(1010)⋅10=1+10100

101011

10101

1

The Extended Euclidean algorithm will stop when *r* reaches 1. The inverse is given by *w*.

We always check *a* ⋅ *a*1 ≡ 1 (mod b)

101011⋅10101= 101011

101011

101011

1000110111 mod 100011011

100011011

1

In AES algorithm, this irreducible polynomial is

=1000110112 = or {01}{1B} in hexadecimal notation.

In the S-box of AES, take the multiplicative inverse in the finite field GF(28) first where

element {00} is mapped to itself {00}.

Let us take  from the top left corner of the S-box and then = 011000112 = 6316.

Note: A matrix multiplication here is written in big-endian.

One more time, Let us take  from the bottom right corner of the S-box and then we need to take multiplicative inverse first modulo the irreducible polynomial =1000110112

=100111002=9C16.

An S-box used in the **SubBytes()** transformation is presented in hexadecimal index.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | y | | | | | | | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| x | 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 | 17 | C4 | A7 | 7E | 3D | 64 | 5D | 19 | 73 |
| 9 | 60 | 81 | 4F | DC | 22 | 2A | 90 | 88 | 46 | EE | B8 | 14 | DE | 5E | 0B | DB |
| A | E0 | 32 | 3A | 0A | 49 | 06 | 24 | 5C | C2 | 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 |

**Tutorial 3**: An inverse of irreducible polynomial.

Take y = matrix ID mod 100 as the last 2 digit of ID number.

1. Take *a* = 100 + y, compute *a*1 (mod *b*)
2. Convert *a* into a polynomial over F2.
3. Take *a*(*x*) as a polynomial, compute *a*1 mod irreducible polynomial *b*(*x*) = *x* 8+ *x*4+ *x*3+*x*+1 where *b*(2) = 28310.
4. Plug in an inverse into an Affine Transform to get an output for AES S-box.

An Affine Transform in AES is given as



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *a*-1 |  | **0** |  | **1** |  | **2** |  | **3** |  | **4** |  | **5** |  | **6** |  | **7** |  | **8** |  | **9** |  | **A** |  | **B** |  | **C** |  | **D** |  | **E** |  | **F** |
| **0** | 0 | 0 | 0 | 1 | 8 | D | F | 6 | C | B | 5 | 2 | 7 | B | D | 1 | E | 8 | 4 | F | 2 | 9 | C | 0 | B | 0 | E | 1 | E | 5 | C | 7 |
| **1** | 7 | 4 | B | 4 | A | A | 4 | B | 9 | 9 | 2 | B | 6 | 0 | 5 | F | 5 | 8 | 3 | F | F | D | C | C | F | F | 4 | 0 | E | E | B | 2 |
| **2** | 3 | A | 6 | E | 5 | A | F | 1 | 5 | 5 | 4 | D | A | 8 | C | 9 | C | 1 | 0 | A | 9 | 8 | 1 | 5 | 3 | 0 | 4 | 4 | A | 2 | C | 2 |
| **3** | 2 | C | 4 | 5 | 9 | 2 | 6 | C | F | 3 | 3 | 9 | 6 | 6 | 4 | 2 | F | 2 | 3 | 5 | 2 | 0 | 6 | F | 7 | 7 | B | B | 5 | 9 | 1 | 9 |
| **4** | 1 | D | F | E | 3 | 7 | 6 | 7 | 2 | D | 3 | 1 | F | 5 | 6 | 9 | A | 7 | 6 | 4 | A | B | 1 | 3 | 5 | 4 | 2 | 5 | E | 9 | 0 | 9 |
| **5** | E | D | 5 | C | 0 | 5 | C | A | 4 | C | 2 | 4 | 8 | 7 | B | F | 1 | 8 | 3 | E | 2 | 2 | F | 0 | 5 | 1 | E | C | 6 | 1 | 1 | 7 |
| **6** | 1 | 6 | 5 | E | A | F | D | 3 | 4 | 9 | A | 6 | 3 | 6 | 4 | 3 | F | 4 | 4 | 7 | 9 | 1 | D | F | 3 | 3 | 9 | 3 | 2 | 1 | 3 | B |
| **7** | 7 | 9 | B | 7 | 9 | 7 | 8 | 5 | 1 | 0 | B | 5 | B | A | 3 | C | B | 6 | 7 | 0 | D | 0 | 0 | 6 | A | 1 | F | A | 8 | 1 | 8 | 2 |
| **8** | 8 | 3 | 7 | E | 7 | F | 8 | 0 | 9 | 6 | 7 | 3 | B | E | 5 | 6 | 9 | B | 9 | E | 9 | 5 | D | 9 | F | 7 | 0 | 2 | B | 9 | A | 4 |
| **9** | D | E | 6 | A | 3 | 2 | 6 | D | D | 8 | 8 | A | 8 | 4 | 7 | 2 | 2 | A | 1 | 4 | 9 | F | 8 | 8 | F | 9 | D | C | 8 | 9 | 9 | A |
| **A** | F | B | 7 | C | 2 | E | C | 3 | 8 | F | B | 8 | 6 | 5 | 4 | 8 | 2 | 6 | C | 8 | 1 | 2 | 4 | A | C | E | E | 7 | D | 2 | 6 | 2 |
| **B** | 0 | C | E | 0 | 1 | F | E | F | 1 | 1 | 7 | 5 | 7 | 8 | 7 | 1 | A | 5 | 8 | E | 7 | 6 | 3 | D | B | D | B | C | 8 | 6 | 5 | 7 |
| **C** | 0 | B | 2 | 8 | 2 | F | A | 3 | D | A | D | 4 | E | 4 | 0 | F | A | 9 | 2 | 7 | 5 | 3 | 0 | 4 | 1 | B | F | C | A | C | E | 6 |
| **D** | 7 | A | 0 | 7 | A | E | 6 | 3 | C | 5 | D | B | E | 2 | E | A | 9 | 4 | 8 | B | C | 4 | D | 5 | 9 | D | F | 8 | 9 | 0 | 6 | B |
| **E** | B | 1 | 0 | D | D | 6 | E | B | C | 6 | 0 | E | C | F | A | D | 0 | 8 | 4 | E | D | 7 | E | 3 | 5 | D | 5 | 0 | 1 | E | B | 3 |
| **F** | 5 | B | 2 | 3 | 3 | 8 | 3 | 4 | 6 | 8 | 4 | 6 | 0 | 3 | 8 | C | D | D | 9 | C | 7 | D | A | 0 | C | D | 1 | A | 4 | 1 | 1 | C |

Table 3.2 An inverse table *a*(*x*) mod *m*(*x*) = *x*8+*x*4+*x*3+*x*+1.

Let *a*(*x*) = *x*5+*x*3+*x* and *m*(*x*) = *x*8+*x*4+*x*3+*x*+1 be the polynomials respectively.

For *a* = 4210 = 32+8+2 = 1010102 = *x*5+*x*3+*x* = 2A16. Trace along row 2 and column A, an answer is 9816.

From the *a* = 4210 = 1010102 = 2A16, then *a*−1 = 15210 =100110002 = 9816. Thus, *a*−1(*x*) = *x*7+*x*4+*x*3.

=11100101=E516.

One more time, take *a* = 4310 = 32+8+2+1 = 1010112 = *x*5+*x*3+*x*+1= 2B16. Trace along row 2 and column B, an answer is 1516. From the *a* = 4310 = 1010112 = 2B16, then *a*−1  =101012 = 1516. Thus, *a*−1(*x*) = *x*4+*x*2+1.

Let us take another example: Suppose *a*= 200 = 128+64+8= 110010002 = C816.

Then from an inverse table, we will get *a−*1 = A916 = 101010012

==111010002=E816.