

## Lecture 7: Finite Fields (PART 4)

### PART 4: Finite Fields of the Form $GF(2^n)$

#### Theoretical Underpinnings of AES and ECC

#### Lecture Notes on “Computer and Network Security”

by Avi Kak (kak@purdue.edu)

February 19, 2009

©2009 Avinash Kak, Purdue University

Goal of this part:

- To review finite fields of the form  $GF(2^n)$
- To show how arithmetic operations can be carried out by directly operating on the bit patterns for the elements of  $GF(2^n)$

## 7.1: Consider Again the Polynomials Over $GF(2)$

- Here are some examples:

$$x + 1$$

$$x^2 + x + 1$$

$$x^2 + 1$$

$$x^3 + 1$$

$$x$$

$$1$$

$$x^5$$

$$x^{10000}$$

We could also shown polynomials with negative coefficients, but recall that in  $GF(2)$ ,  $-1$  is the same as  $+1$ .

- Obviously, the number of such polynomials is infinite.
- The polynomials can be subject to the algebraic operations of addition and multiplication in which the coefficients are added and multiplied according to the rules that apply to  $GF(2)$ .

- As stated in the previous lecture, the set of such polynomials forms a **ring**, called the **polynomial ring**.

## 7.2: Modular Polynomial Arithmetic

Let's now add one more twist to the algebraic operations we carry out on all the polynomials over  $GF(2)$ :

- We will first choose a particular **irreducible polynomial**, as for example

$$x^3 + x + 1$$

(By the way there exist **only two** irreducible polynomials of degree 3 over  $GF(2)$ . The other is  $x^3 + x^2 + 1$ .)

- We will now consider all polynomials defined over  $GF(2)$  modulo the irreducible polynomial  $x^3 + x + 1$ .
- In particular, when an algebraic operation (*we are obviously talking about polynomial multiplication*) results in a polynomial whose degree equals or exceeds that of the irreducible polynomial, we will take for our result the remainder modulo the irreducible polynomial.

- For example,

$$\begin{aligned}
& (x^2 + x + 1) \times (x^2 + 1) \text{ mod } (x^3 + x + 1) \\
&= (x^4 + x^3 + x^2) + (x^2 + x + 1) \text{ mod } (x^3 + x + 1) \\
&= (x^4 + x^3 + x + 1) \text{ mod } (x^3 + x + 1) \\
&= -x^2 - x \\
&= x^2 + x
\end{aligned}$$

Recall that  $1 + 1 = 0$  in  $GF(2)$ . This is what we used in getting to the second expression on the right hand side.

- For the division by the modulus in the above example, we used the result

$$\frac{(x^4 + x^3 + x + 1)}{(x^3 + x + 1)} = x + 1 + \frac{-x^2 - x}{x^3 + x + 1}$$

Obviously, for the division on the left hand side, our first quotient term is  $x$ . Multiplying the divisor by  $x$  yields  $x^4 + x^2 + x$  that when subtracted from the dividend gives us  $x^3 - x^2 + 1$ . This dictates that the next term of the quotient be 1, and so on.

### 7.3: How Large is the Set of Polynomials When Multiplications are Carried Out Modulo $x^3 + x + 1$

- With multiplications modulo  $x^3 + x + 1$ , we have only the following **eight** polynomials in the set of polynomials over  $GF(2)$ :

0

1

$x$

$x + 1$

$x^2$

$x^2 + 1$

$x^2 + x$

$x^2 + x + 1$

- We will refer to this set as  $GF(2^3)$  where the power of 2 is the degree of the **modulus polynomial**.
- Our conceptualization of  $GF(2^3)$  is analogous to our conceptualization of the set  $Z_8$ . The **eight** elements of  $Z_8$  are to be thought of as integers modulo 8. So, basically,  $Z_8$  maps **all** integers to the eight in the set  $Z_8$ . Similarly,  $GF(2^3)$  maps all of the polynomials over  $GF(2)$  to the eight polynomials shown above.

- But note the crucial difference between  $GF(2^3)$  and  $Z_8$ :  $GF(2^3)$  is a field, whereas  $Z_8$  is **NOT**.

## 7.4: How Do We Know That $GF(2^3)$ is a Finite Field?

- We do know that  $GF(2^3)$  is an abelian group because of the operation of polynomial addition satisfies all of the requirements on a group operator and because polynomial addition is commutative.
- $GF(2^3)$  is also a commutative ring because polynomial multiplication distributes over polynomial addition (and because polynomial multiplication meets all the other stipulations on the ring operator: closedness, associativity, commutativity).
- $GF(2^3)$  is an integral domain because of the fact that the set contains the multiplicative identity element 1 and because if for  $a \in GF(2^3)$  and  $b \in GF(2^3)$  we have

$$a \times b = 0 \text{ mod } (x^3 + x + 1)$$

then either  $a = 0$  or  $b = 0$ . This can be proved easily as follows:

- Assume that neither  $a$  nor  $b$  is zero when  $a \times b = 0 \text{ mod } (x^3 + x + 1)$ . In that case, the following equality must also hold

$$a \times b = (x^3 + x + 1)$$

since

$$0 \equiv (x^3 + x + 1) \pmod{(x^3 + x + 1)}$$

– But the above implies that the **irreducible** polynomial  $x^3 + x + 1$  can be factorized, which by definition cannot be done.

- $GF(2^3)$  is a finite field because it is a finite set and because it contains a unique multiplicative inverse for every non-zero element.
- $GF(2^3)$  contains a unique multiplicative inverse for every non-zero element for the same reason that  $Z_7$  contains a unique multiplicative inverse for every non-zero integer in the set. (For a counterexample, recall that  $Z_8$  does not possess multiplicative inverses for 2, 4, and 6.)
- In other words, for every non-zero element  $a \in GF(2^3)$  there is always a unique element  $b \in GF(2^3)$  such that  $a \times b = 1$ .
- This follows from the fact if you multiply a non-zero element  $a$  with each of the eight elements of  $GF(2^3)$ , the result will be

**eight distinct** elements of  $GF(2^3)$ . Obviously, the results of such multiplications **must** equal 1 for exactly one of the non-zero element of  $GF(2^3)$ . So if  $a \times b = 1$ , then  $b$  must be the multiplicative inverse for  $a$ .

- The same thing happens in  $Z_7$ . If you multiply a non-zero element  $a$  of this set with each of the seven elements of  $Z_7$ , you will get **seven distinct** answers. The answer **must** therefore equal 1 for at least one such multiplication. When the answer is 1, you have your multiplicative inverse for  $a$ .
- For a counterexample, this is not what happens in  $Z_8$ . When you multiply 2 with every element of  $Z_8$ , you do not get **eight distinct** answers. (Multiplying 2 with every element of  $Z_8$  yields  $\{0, 2, 4, 6, 0, 2, 4, 6\}$  that has only **four distinct** elements).
- For a more formal proof (by contradiction) of the fact that if you multiply a non-zero element  $a$  of  $GF(2^3)$  with every element of the same set, no two answers will be the same, let's assume that this assertion is false. That is, we assume the existence of two distinct  $b$  and  $c$  in the set such that

$$a \times b \equiv a \times c \pmod{x^3 + x + 1}$$

That implies

$$a \times (b - c) \equiv 0 \text{ mod } (x^3 + x + 1)$$

That implies that either  $a$  is 0 or that  $b$  equals  $c$ . In either case, we have a contradiction.

- **The upshot is that  $GF(2^3)$  is a finite field.**

## 7.5: $GF(2^n)$ is a Finite Field for Every $n$

- None of the arguments on the previous three pages is limited by the value 3 for the power of 2. That means that  $GF(2^n)$  is a finite field for every  $n$ .
- To find all the polynomials in  $GF(2^n)$ , we obviously need an irreducible polynomial of degree  $n$ .
- AES arithmetic is based on  $GF(2^8)$ . It uses the following irreducible polynomial

$$x^8 + x^4 + x^3 + x + 1$$

- The finite field  $GF(2^8)$  used by AES obviously contains 256 distinct polynomials over  $GF(2)$ .
- In general,  $GF(p^n)$  is a finite field for any prime  $p$ . The elements of  $GF(p^n)$  are polynomials over  $GF(p)$  (which is the same as the set of residues  $Z_p$ ).

## 7.6: Representing the Individual Polynomials in $GF(2^n)$ by Binary Code Words

- Let's revisit the **eight polynomials** in  $GF(2^3)$  (when the modulus polynomial is  $x^3 + x + 1$ ):

0

1

$x$

$x + 1$

$x^2$

$x^2 + 1$

$x^2 + x$

$x^2 + x + 1$

- We now claim that there is nothing sacred about the variable  $x$  in such polynomials.
- We can think of  $x^i$  as being merely a place-holder for a bit.
- That is, we can think of the polynomials as bit strings corresponding to the coefficients that can only be 0 or 1, **each power of  $x$  representing a specific position in a bit string**.

- So the  $2^3$  polynomials of  $GF(2^3)$  can therefore be represented by the bit strings:

$0$	$\Rightarrow$	$000$
$1$	$\Rightarrow$	$001$
$x$	$\Rightarrow$	$010$
$x + 1$	$\Rightarrow$	$011$
$x^2$	$\Rightarrow$	$100$
$x^2 + 1$	$\Rightarrow$	$101$
$x^2 + x$	$\Rightarrow$	$110$
$x^2 + x + 1$	$\Rightarrow$	$111$

- If we wish, we can give a decimal representation to each of the above bit patterns. The decimal values between 0 and 7, both limits inclusive, would have to obey the addition and multiplication rules corresponding to the underlying finite field.
- Exactly the same approach can be used to come up with  $2^n$  bit patterns, each pattern consisting of  $n$  bits, for a set of integers that would constitute a finite field, provided we have available to us an irreducible polynomial of degree  $n$ .

## 7.7: Some Observations on Arithmetic Addition in $GF(2^n)$

- We know that the polynomial coefficients in  $GF(2^n)$  must obey the arithmetic rules that apply to  $GF(2)$  (which is the same as  $Z_2$ , the set of remainders modulo 2).
- And we know that the operation of addition in  $GF(2)$  is like the logical XOR operation.
- Therefore, adding the bit patterns in  $GF(2^n)$  simply amounts to taking the **bitwise XOR of the bit patterns**. For example, the following must hold in  $GF(2^8)$ :

$$\begin{array}{rclclclclcl} 5 & + & 13 & = & 0000 & 0101 & + & 0000 & 1101 & = & 0000 & 1000 & = & 8 \\ 76 & + & 22 & = & 0100 & 1100 & + & 0001 & 0110 & = & 0101 & 1010 & = & 90 \\ 7 & - & 3 & = & 0000 & 0111 & - & 0000 & 0011 & = & 0000 & 0100 & = & 4 \\ 7 & + & 3 & = & 0000 & 0111 & + & 0000 & 0011 & = & 0000 & 0100 & = & 4 \end{array}$$

- The last two examples above illustrate that **subtracting is the same as adding** in  $GF(2^8)$ . That is because each “number” is its own additive inverse in  $GF(2^8)$ . In other words, for every  $x \in GF(2^8)$ , we have  $-x = x$ . Yet another way of saying the same thing is that for every  $x \in GF(2^8)$ , we have  $x + x = 0$ .

## 7.8: Some Observations on Arithmetic Multiplication in $GF(2^n)$

- Just as it is convenient to use the simple binary arithmetic (in the form of XOR operations) for additions in  $GF(2^n)$ , could we do the same for multiplications?
- Recall, that we can of course multiply the bit patterns of  $GF(2^n)$  by going back to the modulo polynomial arithmetic and using the multiplications operations defined in  $GF(2)$  for the coefficients. (Recall that in  $GF(2)$ , multiplication is the same as logical AND.)
- But it would be **nice** if we could directly multiply the bit patterns of  $GF(2^n)$  without having to think about the underlying polynomials directly.
- It turns out that we can indeed do so, but the technique is specific to the order of the finite field being used. The **order of a finite field** refers to the number of elements in the field. So the order of  $GF(2^n)$  is  $2^n$ .

- More particularly, the bitwise operations needed for directly multiplying two bit patterns in  $GF(2^n)$  are specific to the irreducible polynomial that defines a given  $GF(2^n)$ .
- On the next slide, we will focus specifically on the  $GF(2^8)$  finite field that is used in AES and show multiplications can be carried out directly in this field by using bitwise operations.

## 7.9: Direct Bitwise Operations for Multiplications in $GF(2^8)$

- Let's consider the finite field  $GF(2^8)$  that is used in AES. This field is derived using the following irreducible polynomial of degree 8:

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

- Now let's see how we can carry out multiplications with direct bitwise operations in this  $GF(2^8)$ .
- We first take note of the following equality in  $GF(2^8)$ :

$$x^8 \text{ mod } m(x) = x^4 + x^3 + x + 1$$

The result follows immediately by a long division of  $x^8$  by  $x^8 + x^4 + x^3 + x + 1$ . Obviously, the first term of the quotient will be 1. Multiplying the divisor by the quotient yields  $x^8 + x^4 + x^3 + x + 1$ . When this is subtracted from the dividend  $x^8$ , we get  $-x^4 - x^3 - x - 1$ , which is the same as the result shown above.

- Now let's consider the general problem of multiplying a general polynomial  $f(x)$  in  $GF(2^8)$  by just  $x$ . Let's represent  $f(x)$  by

$$f(x) = b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0$$

Therefore, this  $f(x)$  stands for the bit pattern  $b_7b_6b_5b_4b_3b_2b_1b_0$ .

- Obviously,

$$f(x) \times x = b_7x^8 + b_6x^7 + b_5x^6 + b_4x^5 + b_3x^4 + b_2x^3 + b_1x^2 + b_0x$$

But now recall that we must take the modulo of this polynomial with respect to  $m(x) = x^8 + x^4 + x^3 + x + 1$ . What that yields depends on whether or not the bit  $b_7$  is set.

- If the bit  $b_7$  of  $f(x)$  is equals 0, then the right hand above is already in the set of polynomials in  $GF(2^8)$  and nothing further needs to be done. In this case, the output bit pattern is  $b_6b_5b_4b_3b_2b_1b_00$ .

- However, if  $b_7$  equals 1, we need to divide the polynomial we have for  $f(x) \times x$  by the modulus polynomial  $m(x)$  and keep just the remainder. Therefore, when  $b_7 = 1$ , we can write

$$(f(x) \times x) \text{ mod } m(x)$$

$$\begin{aligned}
&= (b_7x^8 + b_6x^7 + b_5x^6 + x_4x^5 + b_3x^4 + b_2x^3 + b_1x^2 + b_0x) \text{ mod } m(x) \\
&= (b_6x^7 + b_5x^6 + x_4x^5 + b_3x^4 + b_2x^3 + b_1x^2 + b_0x) + (x^8 \text{ mod } m(x)) \\
&= (b_6x^7 + b_5x^6 + x_4x^5 + b_3x^4 + b_2x^3 + b_1x^2 + b_0x) + (x^4 + x^3 + x + 1) \\
&= (b_6b_5b_4b_3b_2b_1b_00) \quad \otimes \quad (00011011)
\end{aligned}$$

where, in the last expression shown, we have used the fact that the addition in  $GF(2^8)$  corresponds to the logical XOR operation for the bit patterns involved.

## 7.10: Summary of How a Multiplication is Carried Out in $GF(2^8)$

- Let's say you want to multiply two bit patterns  $B_1$  and  $B_2$ , each 8 bits long.
- If  $B_2$  is the bit pattern 00000001, then obviously nothing needs to be done. The result is  $B_1$  itself.
- If  $B_2$  is the bit pattern 00000010, then we are multiplying  $B_1$  by  $x$ . Now the answer depends on the value of the most significant bit in  $B_1$ . If  $B_1$ 's MSB is 0, the result is obtained by shifting the  $B_1$  bit pattern to the left by one bit and inserting a 0 bit from the right.
- If  $B_1$ 's MSB is 1, first we again shift the  $B_1$  bit pattern to the left as above. Next, we take the XOR of the shifted pattern with the bit pattern 00011011 for the final answer.
- If  $B_2$  is the bit pattern 00000100, then we are multiplying  $B_1$  by  $x^2$ . This amounts to first multiplying  $B_1$  by  $x$ , and then multi-

plying the result again by  $x$ . So it amounts to two applications of the logic in the previous two steps.

- In general, if  $B_2$  consists of a single bit in the  $j^{th}$  position from the right (using the 0 index for the right-most position), we need  $j$  applications of the logic laid out above for multiplying with  $x$ .
- Even more generally, when  $B_2$  consists of an arbitrary bit pattern, we consider the bit pattern to be a sum of bit patterns each containing only single bit.
- For example, if  $B_2$  is 10000011, we can write

$$\begin{aligned}
 & B_1 \times 10000011 \\
 &= B_1 \times (00000001 + 00000010 + 10000000) \\
 &= (B_1 \times 00000001) + (B_1 \times 00000010) + (B_1 \times 10000000) \\
 &= (B_1 \times 00000001) \otimes (B_1 \times 00000010) \otimes (B_1 \times 10000000)
 \end{aligned}$$

Each of the three multiplications shown in the final expression involves multiplying  $B_1$  with a single power of  $x$ . That we can easily do with the logic already explained.

## 7.11: Finding Multiplicative Inverses in $GF(2^n)$

- So far we have talked about efficient bitwise operations for implementing the addition, the subtraction, and the multiplication operations for the bit patterns corresponding to the elements of  $GF(2^n)$ .
- But what about division? Can division be carried out directly on the bit patterns?
- In general, you can use the Extended Euclid Algorithm for finding the multiplicative inverse (MI) of a polynomial in  $GF(2^n)$ .
- If you have fixed the value of  $n$  for a particular  $GF(2^n)$  field (and if  $n$  is not too large), you can precompute the multiplicative inverses for all the elements of  $GF(2^n)$  and store them away. (Recall that the MI of a bit pattern  $A$  in  $GF(2^n)$  is a bit pattern  $B$  so that  $A \times B = 1$ .)
- For example, in  $GF(2^8)$ , the MI of a bit pattern  $A$  is the bit pattern  $B$  so that  $A \times B = 00000001$ .

- The table below shows the multiplicative inverses for the bit patterns of  $GF(2^3)$ . Also shown are the additive inverses. But note that every element  $x$  is its own additive inverse. Also note that the additive identity element is not expected to possess a multiplicative inverse.

	Additive Inverse	Multiplicative Inverse
000	000	-----
001	001	001
010	010	101
011	011	110
100	100	111
101	101	010
110	110	011
111	111	100

## 7.12: Using a Generator to Represent the Elements of $GF(2^n)$

- It is particularly convenient to represent the elements of a Galois Field with the help of a **generator element**.
- If  $g$  is a **generator element**, then every element of  $GF(2^n)$ , except for the 0 element, can be expressed as some power of  $g$ .
- Consider a finite field of order  $q$ . As mentioned previously in Section 7.8, the **order** of a finite field is the number of elements in the field. If  $g$  is the generator of this finite field, then the finite field can be expressed by the set

$$\{0, g^0, g^1, g^2, \dots, g^{q-2}\}$$

- How does one specify a generator?
- A generator is obtained from the irreducible polynomial that went into the creation of the finite field. If  $f(g)$  is the irreducible polynomial used, then  $g$  is that element which symbolically satisfies

the equation  $f(g) = 0$ . You do not actually solve this equation for its roots since an irreducible polynomial cannot have actual roots in the underlying number system used, **but only use this equation for the relationship it gives between the different powers of  $g$ .**

- Consider the case of  $GF(2^3)$  defined with the irreducible polynomial  $x^3 + x + 1$ . The generator  $g$  is that element which symbolically satisfies  $g^3 + g + 1 = 0$ , implying that such an element will obey

$$g^3 = -g - 1 = g + 1$$

- Now we can show that every power of  $g$  will correspond to some element of  $GF(2^3)$ .
- Shown below are the first several powers of  $g$  along with the element 0 at the very top:

$$\begin{array}{rcl}
 & & 0 \\
 & & g^0 = 1 \\
 & & g^1 = g \\
 & & g^2 = g^2 \\
 & & g^3 = g + 1 \\
 g^4 = g(g^3) = g(g + 1) & = & g^2 + g
 \end{array}$$

$$\begin{aligned}
g^5 &= g(g^4) = g(g^2 + g) = g^3 + g^2 = g^2 + g + 1 \\
g^6 &= g(g^5) = g(g^2 + g + 1) = g^3 + g^2 + g = g^2 + 1 \\
g^7 &= g(g^6) = g(g^2 + 1) = g^3 + g = 1 \\
&\qquad\qquad\qquad \vdots
\end{aligned}$$

- Note the powers  $g^0$  through  $g^6$  of the generator element, along with the element 0, correspond to the eight polynomials of  $GF(2^3)$  shown on Slide 10.
- The higher powers of  $g$  obey the relationship  $g^k = g^{k \bmod 7}$  for the example shown. The previous slide already shows that  $g^7$  is the same as  $g^0$ .
- Since every polynomial in  $GF(2^n)$  is represented by a power of  $g$ , multiplying any two polynomials in  $GF(2^n)$  becomes trivial — we just have to add the exponents of  $g$  modulo  $(2^n - 1)$ .
- So we have the conclusion that if  $g$  is the generator element of a finite field of the form  $GF(2^n)$ , then all the powers of  $g$  from  $g^0$  through  $g^{2^n-2}$ , along with the element 0, correspond to the elements of the finite field.

- That is, using the generator notation allows the multiplications of the elements of the finite field to be carried out without reference to the irreducible polynomial.

## **Acknowledgement**

Thanks go to Beau Morrison of Purdue University for catching a typographical error in one of the examples of the addition of bit patterns shown in Section 7.7.