CHAPTER 5

MONOALPHABETIC MULTILITERAL SUBSTITUTION SYSTEMS

Section I

Characteristics and Types

5-1. Characteristics of Multilateral Systems

As explained in Chapter 3, monoalphabetic unilateral systems are those in which the ciphertext unit is always one character long. Multilateral systems are those in which the ciphertext unit is more than one character in length. The ciphertext characters may be letters, numbers, or special characters.

a. Security of Multilateral Systems. By using more than one character of ciphertext for each character of plaintext, encipherment is no longer limited to the same number of different cipher units as there are plaintext units. Although there is still only one alphabet used in multilateral systems, the alphabet can have more than one ciphertext value for each plaintext value. These variant ciphertext values provide increased security. Additionally, the plaintext component of alphabets can be expanded easily to include numbers, punctuation, and common syllables as well as the basic 26 letters. When used, the variation in encipherment and the reduced spelling of numbers, punctuation, and common syllables minimize the exact weaknesses that we used in Chapter 4 to break into unilateral systems.

b. Advantages and Disadvantages. The increased security possible with variant multilateral systems is the major advantage. The major disadvantage is that by substituting more than one character of ciphertext for each plaintext value, the length of messages and resulting transmission times are increased. A second disadvantage is that more training and discipline are required to take advantage of the increased security. If training and discipline are inadequate, the security advantages are lost easily.

5-2. Types of Multilateral Systems

Multiliteral systems are further categorized by the type of substitution used. The major types are—

- Biliteral systems, which replace each plaintext value with two letters of ciphertext.
- Dinomic systems, which replace each plaintext value with two numbers of ciphertext.
- Trilateral and trinomic systems, which replace each plaintext value with three letters or numbers of ciphertext.
- Monome-dinome systems, which replace plaintext values with one number for some values and two numbers for other values.
- Biliteral with variants and dinomic with variants systems, which provide more than one ciphertext value for each plaintext value.
- Syllabary squares, which may be biliteral or dinomic, and which include syllables as well as single characters as plaintext values.

5-3. Cryptography of Multilateral Systems

The cryptography of each type of multilateral system, including some of the odd variations is illustrated in the following paragraphs. Most of these systems are coordinate matrix systems in which the plaintext values are found inside a rectangular matrix and the ciphertext values consist of the row and column coordinates of the matrix.

a. Simple Biliterals and Dinomics. The simplest multilateral systems use no variation. They typically use a small rectangular matrix large enough to contain the letters of the alphabet and any other characters the system designer wants to use as plaintext values.

(1) The plaintext values are the internals of the matrix. They may be entered alphabetically, follow a systematic sequence, or they may be random. They may be entered in rows, in columns, or by any other route.

(2) The row and column coordinates are the externals. Conventionally, the row coordinates are placed at the left outside the matrix, and the column coordinates are placed at the top. As with the internals, the coordinates may be selected randomly or produced systematically.

(3) A ciphertext value is created by finding the plaintext value inside the matrix and then combining the coordinate of the row with the coordinate of the column for that plaintext value. Either can be placed first, although placing the row coordinate before the column coordinate is more common.

(4) Five by five is a common size for a simple system (Figure 5-1). The 26 letters are fitted into the 25 positions in the matrix by combining two letters. The usual combinations are I and J or U and V. It is up to the deciphering cryptographer to determine which of the two is the correct value. There are few, if any, words in common usage in which good words can be formed using either letter of the I/J or U/V combinations. Other common sizes are 6 by 6 (which gives room for the 10 digits), 4 by 7, and 3 by 10. Many other sizes are possible.

A

	V	W	X	Y	Z
A	a	f	l	q	v
B	b	g	m	r	w
C	c	h	n	s	x
D	d	i/j	o	t	y
E	e	k	p	u	z

p: j u l i e t

c: DWEY AXDW EVDY

B

	0	1	2	3	4	5	6	7	8	9
3	m	u	r	p	h	y	s	l	a	w
6	b	c	d	e	f	g	i	j	k	n
9	o	q	t	v	x	z	.	,	?	/

p: a t t a c k .

c: 3892 9238 6168 9600

C

	O	R	A	N	G	E
V	a	b	c	d	e	f
I	t	u	v	w	x	g
O	s	6	7	8	y	h
L	r	5	0	9	z	i
E	q	4	3	2	1	j
T	p	o	n	m	l	k

p: 2 4 t a n k s

c: ENER IOVO TATE OOXX

D

	4	7	2	9	6
3	l	b	i/j	c	k
1	a	r	h	g	q
8	y	t	f	p	v
5	n	e	o	u	x
0	d	m	s	w	z

p: a t t a c k

c: 4187 7814 3963

Figure 5-1. Biliteral and dinomic matrices.

(5) Example A in Figure 5-1 is a simple 5 by 5 matrix with I and J in the same plain-text cell of the square. The coordinates and the sequence within are in alphabetic order.

(6) Example B is a simple 3 by 10 matrix with orderly coordinates and a keyword mixed sequence inscribed within. The four extra cells are used for punctuation marks.

(7) Example C is a 6 by 6 matrix with a spiral alphabetic sequence followed in the spiral with the 10 digits. The coordinates in this case are related words.

(8) Example D is a 5 by 5 matrix with numeric coordinates. The plaintext sequence is keyword mixed entered diagonally. In this case, there is deliberately no repetition between the row and column coordinates. This allows the coordinates to be read either in row-column order or in column-row order without any ambiguity, as in the sample enciphered text. This is unusual, but you should be alert to such possibilities.

b. Triliterals and Trinomics. Trilateral and trinomic systems are essentially the same as biliteral and dinomic systems. The difference is that either the row coordinates or the column coordinates consist of two characters instead of one, creating a three-for-one substitution. Such systems offer no real advantage except to provide a slightly different challenge to the cryptanalyst, and have the distinct disadvantage of tripling the length of messages. They are easily recognized, and offer no increase in security.

	L	M	N	O	P
	V	W	X	Y	Z
A	a	f	l	q	v
B	b	g	m	r	w
C	c	h	n	s	x
D	d	i/j	o	t	y
E	e	k	p	u	z

	0	1	2	3	4	5	6	7	8	9
13	m	u	r	p	h	y	s	l	a	w
26	b	c	d	e	f	g	i	j	k	n
39	o	q	t	v	x	z	.	,	?	/

p: j u l i e t
c: DMW EOY ANX DMW ELV DOY

p: a t t a c k
c: 138 392 392 138 261 268

c. Monome-Dinomes. Monome-dinomes are coordinate matrix systems constructed so that one row has no coordinate. The values from that row are enciphered with the column coordinate only. This means that some ciphertext values are two characters in length (dinomes) and others are only one (monomes). If the values used as row coordinates are also used as column coordinates, no plaintext values are placed in the monome row under those repeated column coordinates. The blanking of cells in the monome row is shown in the example below.

	1	2	3	4	5	6	7	8	9	0
-	h	e	x	a	-	-	d	c	i	m
5	l	b	f	g	j	k	n	o	p	q
6	r	s	t	u	v	w	y	z	.	,

p:	e	n	e	m	y	a	t	t	a	c	k	i	n	g
c:	2	57	2	0	67	4	63	63	4	8	56	9	57	54

Resulting message:

25720 67463 63485 69575 40000

(1) If the cells corresponding to the row coordinates in the monome row are not blanked, the deciphering cryptographer will have difficulty. Decipherment proceeds left to right, and when a 5 or a 6 is encountered in the matrix shown, it will always be a row coordinate or combine with a preceding row coordinate. It will never stand alone as a monome. If the 5 and 6 cells were not blanked, the deciphering cryptographer could not tell if a 5 or 6 were a monome or the beginning of a dinome. The cryptographer would have to rely on context to figure out which was intended, and that could lead to errors.

(2) The additional examples of monome-dinomes shown below demonstrate the various ways they can be constructed. The last example (top of page 5-5) is a monome-dinome-trinome.

	7	0	4	8	5	1	3	9	2	6
-	w	i	l	d	-	c	a	t	-	-
6	b	e	f	g	h	j	k	m	n	o
2	p	q	r	s	u	v	x	y	z	.
5	0	1	2	3	4	5	6	7	8	9

	2	4	6	8	0
-	t	e	n	o	r
1	c	b	x	a	s
3	d	f	g	h	i
5	p	m	l	k	j
7	q	u	v	w	y
9	z	.	,	;	:

	1	2	3	4	5	6	7	8	9	0
-	-	-	r	a	m	c	h	i	p	s
1	b	d	e	f	g	j	k	l	n	o
23	q	t	u	v	w	x	y	z	.	0

p: r e q u e s t h e l p

c: 3 13 231 233 13 0 232 7 13 18 9

Resulting message:

31323 12331 3023271318 90000

d. Variant Systems. Variants in a multiliteral system allow plaintext characters to be enciphered in more than one way. Variants can be external or internal.

(1) External variant systems have a choice of coordinates. Either row coordinates or column coordinates or both can have variants. Examples A and B in Figure 5-2 provide two ways to encipher every letter.

A

	L	M	N	O	P
	V	W	X	Y	Z
A	a	f	l	q	v
B	b	g	m	r	w
C	c	h	n	s	x
D	d	i/j	o	t	y
E	e	k	p	u	z

p: a t t a c k

c: AV DO DY AL CV EM

B

	0	1	2	3	4	5	6	7	8	9
13	m	u	r	p	h	y	s	l	a	w
26	b	c	d	e	f	g	i	j	k	n
49	o	q	t	v	x	z	.	,	?	/

p: a t t a c k .

c: 1892 4238 6128 9600

C

	L	M	N	O	P
	Q	R	S	T	U
AB	a	f	l	q	v
CD	b	g	m	r	w
EF	c	h	n	s	x
GH	d	i/j	o	t	y
JK	e	k	p	u	z

p: a t t a c k

c: BQGT HTAL EQKM

D

	0	1	2				
	3	4	5	6	7	8	9
1234	e	t	o	l	u	h	m
567	r	n	i	c	f	g	p
89	a	s	d	b	v	w	y
0	.	j	k	q	x	z	,

p: r e f e r e n c e

c: 6023 7710 5340 7176 3300

Figure 5-2. External variant systems.

Example C provides four ways to encipher every letter. Example D was constructed to provide the most variants for the most common letters. The letters E, T, and O can all be enciphered in eight different ways. R, N, and I can be enciphered in six different ways. A, S, D, L, U, H, and M can be enciphered in four different ways. Q, X, Z, and the comma can only be enciphered one way. When any of the systems are conscientiously used, repeated words in the text will not produce repeated ciphertext segments.

(2) Internal variant systems use larger matrices to provide variants inside the matrix. Each common plaintext letter appears more than once. Here are two examples of internal variant systems.

	3	0	2	8	6	5	1	4	7	9
7	e	e	e	e	t	t	o	n	i	s
3	e	e	e	t	t	o	r	i	a	d
9	e	e	t	t	o	r	i	a	d	u
1	e	e	t	o	r	n	a	d	u	f
6	e	t	o	r	n	a	d	u	c	m
4	t	o	r	n	a	s	u	c	m	p
8	o	r	n	a	s	l	c	y	g	w
2	r	n	i	s	l	h	y	g	v	k
5	n	i	s	l	h	f	b	v	j	x
0	i	s	l	h	f	b	p	w	q	z

	K	L	M	N	O	P	Q	R	S	T
A	l	u	c	k	y	c	h	a	r	m
B	o	b	j	e	c	t	i	o	n	s
C	g	o	l	d	r	e	c	o	r	d
D	a	f	f	e	c	t	i	o	n	s
E	r	a	p	s	e	s	s	i	o	n
F	i	n	c	e	n	d	i	a	r	y
G	t	r	i	v	i	a	q	u	i	z
H	h	e	a	v	y	m	e	t	a	l
I	m	a	s	t	e	r	w	o	r	k
J	s	i	x	t	y	s	e	v	e	n

The first example above places the letters in the matrix according to their expected frequency in plaintext. If their use is well balanced, all letters in the square will be used with about the same frequency. The second square achieves the same effect by using 10 words or phrases in the rows, which use all the letters. The first letters of the column spell out an eleventh word—logarithms.

e. Syllabary Squares. Another type of internal variant system is the syllabary square. This type includes common syllables as well as single letters. When these are used, the same square may be used for a period, changing the coordinates more frequently than the square itself.

	6	0	4	3	8	1	7	5	9	2
8	a	1	ad	al	an	and	as	at	b	2
4	c	3	ce	co	d	4	da	de	di	e
3	5	ea	ec	ed	ee	ei	el	en	ent	er
7	es	et	f	6	fi	fo	g	7	h	8
2	hi	ht	i	9	in	ing	io	ir	is	it
0	j	0	00	k	l	la	le	ll	m	ma
5	n	nd	ne	ng	ni	nt	o	on	or	ou
9	p	q	r	ra	re	ri	ro	rs	rt	s
1	se	si	st	t	ta	te	th	ti	tion	to
6	tw	ty	u	ur	v	ve	w	x	y	z

p: r ei n fo r ce m en t s

c: 94 31 56 71 94 44 09 35 13 92

p: re in f or ce m ent s

c: 98 28 74 59 44 09 39 92

The two sample encipherments of *REINFORCEMENTS* show that a syllabary square suppresses repeats in ciphertext just as single letter variant systems do. It also has the advantage of producing shorter text than single letter multilateral systems.

f. Sum Checks. It is very easy for errors to occur when messages are transmitted and received, whatever means of transmission are used. Because of this, some users introduce an error detection feature into traffic known as sum checking.

(1) In its simplest form, a sum-check digit is added to every pair of digits in numeric messages. The digit is produced by adding the pair of digits to produce the

third. If the result is larger than 9, only the second digit is used, dropping the 10's digit, for example 8 plus 9 equals 7 instead of 17. This is also known as modulo 10 arithmetic.

Ciphertext: 42 63 55 47 22 89

Ciphertext with sum check: 42<u>6</u> 63<u>9</u> 55<u>0</u> 47<u>1</u> 22<u>4</u> 89<u>7</u>

(2) Whenever the first two digits do not add up to the third, the receiving cryptographer is alerted that an error has occurred. The cryptographer then tries to figure out the correct digit from context or by assuming that two of the digits are correct and determining what the third should be.

(3) There are many variations on the simple system of sum checking described here. Sometimes the sum-check digit will be placed first or second in each resulting group of three. Sometimes a sum check will be applied to a larger group than two numbers. Sometimes a different rule of arithmetic will be used, such as adding the sum-check digit so that the resulting three always add to the same total. Sometimes a more complex system will be used that provides enough information to resolve many errors as well as detect them, particularly when computers are used in data and text transmissions.

(4) Computer produced sum checks can be used with any characters, not just numbers. Computer produced sum checks will normally be invisible to the user, as they are automatically stripped out when a message is received. They may or may not be invisible to the cryptanalyst. Recovery of computer produced sum checks is well beyond the scope of this text, but you should be alert to their existence.

Section II
Analysis of Simple Multilateral Systems

5-4. Techniques of Analysis

The first steps in solving any multilateral system are to identify the system and establish the coordinates. It makes little difference whether the system uses numbers or letters for coordinates. The techniques are the same in either case. Once the system is identified and the coordinates set up, a solution of the simpler systems is the same as with unilateral systems. Variant systems require additional steps. Each type is considered in the following paragraphs.

5-5. Identification of Simple Biliteral and Dinomic Systems

Simple biliteral and dinomic systems are very easy to recognize and solve.

a. First, the two-for-one nature of the system will usually be apparent. The message will be even in length. The majority of repeated segments will be even in length, although when an adjacent row or column coordinate is the same, a repeat may appear odd in length. The distance between repeats, counted from the first letter of one to the first letter of the next, will be even in length.

b. Second, unless the identical letters or numbers are used for row and column coordinates, there will be limitation by position. One set will appear in the row coordinate position, and the other set will appear in the column coordinate position. Even in the case where all coordinates are different and either the row or column coordinate character may be placed first, each pair will be limited to one from one set and one from the other. If you do not recognize it right away, charting contacts will make it obvious.

c. For systems with letters as coordinates, not more than half the alphabet will be used as coordinates. This severe limitation in letters used is the most obvious characteristic, since only very short unilateral messages are ever that limited. A phi index of coincidence will reflect that limitation, always appearing much higher than expected for a unilateral system.

d. Dinomic systems, since they are limited to the 10 digits anyway, are not quite as obvious. Simple systems should still show positional limitation, however.

5-6. Sample Solution of a Dinomic System

The next problem shows the steps in solution of a sample dinomic system. These steps apply equally to biliteral systems.

```
2023 2029 6224 6322 2144  4420 6362 4924 6529 2769
2043 2123 2227 4627 6521  2221 2723 6527 2349 2144
4481 8287 2423 4349 2144  4485 8089 6522 2746 2421
6365 2263 2142 2027 2324  6322 2144 4420 6362 4627
6521 2221 2723 6560 2144  4441 2047 2123 2422 6680

6666 6522 2746 4263 2069  2122 6425 2729 2924 2343
2123 4700
```

a. The most obvious thing about this cryptogram is that every pair of numbers begins with 2, 4, 6, or 8. The final pair begins with 0, but since it appears nowhere else, it is probably a filler. This suggests that we are dealing with a matrix with four rows.

b. Scanning the second digit of every pair, we see that there is some limitation in the column position, also. All digits are used except 8. The matrix appears to have nine columns, although it is possible that a column for 8 exists, but no values from it were used. Four by nine is a reasonable size for a matrix.

c. Next, we check for repeats and underline them. We also prepare a dinomic frequency count by setting up a 4 by 9 matrix and checking off each dinome that appears.

```
2023 2029 6224 6322 2144  4420 6362 4924 6529 2769
2043 2123 2227 4627 6521  2221 2723 6527 2349 2144
4481 8287 2423 4349 2144  4485 8089 6522 2746 2421
6365 2263 2142 2027 2324  6322 2144 4420 6362 4627
6521 2221 2723 6560 2144  4441 2047 2123 2422 6680

6666 6522 2746 4263 2069  2122 6425 2729 2924 2343
2123 4700
```

	1	2	3	4	5	6	7	9	0
2	15	10	10	7	1		11	4	8
4	1	2	3	10		4	2	3	

d. The two longer repeats both include patterns of repeated values. Word patterns can be constructed on repeated dinomes just as they were for repeated single letters. The word patterns for the two longer repeats are shown below.

```
-  A  B  C  D  D  E  A  -
24 63 22 21 44 44 20 63 62
A  R  T  I  L  L  E  R  Y

-  A  B  C  D  C  A  E  B
46 27 65 21 22 21 27 23 65
P  O  S  I  T  I  O  N  S
```

e. The word pattern lists in Appendix D show only one possibility for each pattern as shown. The two are consistent with each other. Using these recoveries, we can set up a matrix and place the values in it and the cryptogram.

```
e n  e    y a  r t  i l   l e  r y    a  s    o
2023 2029 6224 6322 2144  4420 6362 4924 6529 2769

e    i n  t o  p o  s i   t i  o n  s o  n    i l
2043 2123 2227 4627 6521  2221 2723 6527 2349 2144

l         a n       i l   l         s t  o p  a i
4481 8287 2423 4349 2144  4485 8089 6522 2746 2421

r s  t r  i    e o  n a   r t  i l  l e  r y  p o
6365 2263 2142 2027 2324  6322 2144 4420 6362 4627

s i  t i  o n  s    i l   l    e    i n  a t
6521 2221 2723 6560 2144  4441 2047 2123 2422 6680

     s t  o p    r  e     i t       o      a  n
6666 6522 2746 4263 2069  2122 6425 2729 2924 2343

i n
2123 4700
```

	1	2	3	4	5	6	7	9	0
2	i	t	n	a			o		e
4				l		p			
6		y	r		s				
8									

f. The plaintext words *ENEMY* and *AIRSTRIKE* are now obvious. Placing the M from ENEMY shows *COMMANDING* at the end of the message. Most of the remaining plaintext letters are easily recovered.

```
e n  e m  y a  r t  i l   l e  r y  h a  s m  o v
2023 2029 6224 6322 2144  4420 6362 4924 6529 2769

e d  i n  t o  p o  s i   t i  o n  s o  n h  i l
2043 2123 2227 4627 6521  2221 2723 6527 2349 2144

l         a n  d h  i l   l         s t  o p  a i
4481 8287 2423 4349 2144  4485 8089 6522 2746 2421

r s  t r  i k  e o  n a   r t  i l  l e  r y  p o
6365 2263 2142 2027 2324  6322 2144 4420 6362 4627

s i  t i  o n  s w  i l   l b  e g  i n  a t
6521 2221 2723 6560 2144  4441 2047 2123 2422 6680

     s t  o p  k r  e v   i t    c  o m  m a  n d
6666 6522 2746 4263 2069  2122 6425 2729 2924 2343

i n  g
2123 4700
```

	1	2	3	4	5	6	7	9	0
2	i	t	n	a	c		o	m	e
4	b	k	d	l		p	g	h	
6		y	r		s			v	w
8									

g. The letters in the second row precede all the letters in the third row alphabetically. This suggests an alphabetic structure, although the columns are clearly not in the correct order. The first row probably contains a keyword. If we rearrange the columns so the letters in the second and third rows fall in alphabetical order, we see the next structure.

	1	3	5	7	9	0	2	4	6
2	i	n	c	o	m	e	t	a	
4	b	d		g	h		k	l	p
6		r	s		v	w	y		
8									

h. The plaintext letters area keyword mixed sequence based on INCOME TAX. After placing the remaining letters, there are still 10 blank cells in the matrix. Seven of them are used in the cryptogram, and they cluster together in segments of three or four dinomes. They show the typical pattern of numbers. In particular, the four plaintext values of groups 50 and 51 of the message indicate time, and 66 is probably a 0. More likely than not, the remaining numbers fill the bottom row of the matrix in numerical order, but these recoveries cannot be confirmed without more information. If hill numbers could be compared to known numbers from an enemy map sheet, we could accept the values with more confidence. At this point, we are reasonably confident of the letter arrangement and the number 0, but the remaining numbers are only a possibility. However, if this were a current real life situation and the enemy referred to by the text is our own forces, we would certainly consider reporting the likelihood of air strikes on our artillery positions.

5-7. Analysis of Monome-Dinome Systems

The characteristics of biliteral and dinomic systems that stand out most are the divisibility by two and the positional limitation that makes it easy to determine matrix coordinates. By changing the length of the plaintext unit from character to character, monome-dinome systems avoid both of these characteristics. In their place, however, the frequency of the numbers (or occasionally, letters) used as row coordinates tends to be higher than the other coordinates. Choosing the highest frequency numbers as row coordinates gives a starting point to reconstruct a monome-dinome system. Consider the next example.

80796	78009	60720	51187	33812
07960	76059	69730	71070	99089
60905	96070	62050	09109	13866
96058	24710	81059	69740	79610
90591	19787	16833	07389	70805
00019	60509	07055	05458	57950
19196	97407	96960	72051	18733
81207	06910	70390	56545	35399
95205	00030	08204		

Numbers:	1	2	3	4	5	6	7	8	9	0
Frequency:	19	8	13	6	22	20	25	16	33	53

a. Repeats are underlined and the number frequencies are shown in the example. A dinomic system can be ruled out, because the repeats are an odd interval apart. The distance between the repeats is 153 characters, counting from the first character of one to the first character of the next. A three-for-one substitution is possible from the position of the repeats, but no patterns or positional limitations appear when divided into threes. The very high frequency of the numbers 0 and 9 in relation to the other numbers suggests that the system is monome-dinome. The most likely row coordinates are 0 and 9. Other row coordinates are possible, but at this point it is best to start with the most likely candidates only.

b. Begin by breaking the message into monomes and dinomes using only the 0 and 9 as row coordinates. Mark off the divisions in pencil, keeping in mind that some changes may be required later. Start with the first character of the message and work through in order to the end, marking off the monomes and dinomes. Whenever the first character after a division is a 0 or 9, include it with the next character. If it is any other character, leave it as a monome.

```
8/0 7/9 6/    7/8/0 0/9    6/0 7/2/0    5/1/1/8/7/    3/3/8/1/2/
0 7/9 6/0     7/6/0 5/9    6/9 7/3/0    7/1/0 7/0     9/9 0/8/9
6/0 9/0 5/    9 6/0 7/0    6/2/0 5/0    0/9 1/0 9/    1/3/8/6/6/
9 6/0 5/8/    2/4/7/1/0    8/1/0 5/9    6/9 7/4/0     7/9 6/1/0
9/0 5/9 1/    1/9 7/8/7/   1/6/8/3/3/   0 7/3/8/9     7/0 8/0 5/
0 0/0 1/9     6/0 5/0 9/   0 7/0 5/5/   0 5/4/5/8/    5/7/9 5/0
1/9 1/9 6/    9 7/4/0 7/   9 6/9 6/0    7/2/0 5/1/    1/8/7/3/3/
8/1/2/0 7/    0 6/9 1/0    7/0 3/9 0/   5/6/5/4/5/    3/5/3/9 9/
9 5/2/0 5/    0 0/0 3/0    0/8/2/0 4
```

c. With the divisions in place, we can try a word pattern on the long repeat.

```
96 07 2 05 1 1 8 7 3 3 8 1 2 07
-  A  B C  D D E F G G E D B A
R  E  C O  N N A I S S A N C E
```

d. We next set up a monome-dinome matrix with row coordinates 0 and 9 and include the recovered letters. Shown below is the partially recovered matrix and the cryptogram with all letters from *RECONNAISSANCE* placed in the plaintext and the matrix.

```
a   e   r    i a       r   e    c   o  n n a  i   s s a n c
8/0 7/9 6/   7/8/0 0/9   6/0 7/2/0   5/1/1/8/7/   3/3/8/1/2/

 e   r     e      o    r        s   e   n  e             a  r
0 7/9 6/0    7/6/0 5/9   6/9 7/3/0   7/1/0 7/0    9/9 0/8/9

        o     r   e         c  o                   n s a
6/0 9/0 5/   9 6/0 7/0   6/2/0 5/0   0/9 1/0 9/   1/3/8/6/6/

 r   o   a   c   i n        n  o    r           e     r  n
9 6/0 5/8/   2/4/7/1/0   8/1/0 5/9   6/9 7/4/0    7/9 6/1/0

    o        n     a i    n   a s s    e  s a             o
9/0 5/9 1/   1/9 7/8/7/  1/6/8/3/3/  0 7/3/8/9    7/0 8/0 5/

           r    o          e   o       o      a     i
0 0/0 1/9    6/0 5/0 9/  0 7/0 5/5/  0 5/4/5/8/   5/7/9 5/0

        r           e      r   r    e   c  o  n   n a i s s
1/9 1/9 6/   9 7/4/0 7/  9 6/9 6/0   7/2/0 5/1/   1/8/7/3/3/

a n c  e               e                          s   s
8/1/2/0 7/   0 6/9 1/0   7/0 3/9 0/  5/6/5/4/5/   3/5/3/9 9/

     c  o                  a c
9 5/2/0 5/   0 0/0 3/0   0/8/2/0 4
```

	1	2	3	4	5	6	7	8	9	0
-	n	c	s				i	a		
0					o		e			
9						r				

e. These recoveries suggest additional plaintext, particularly the message beginning *AERIAL RECONNAISSANCE REPORTS ENEMY.* Placing these new values leads to additional recoveries.

```
a  e   r    i a  l    r    e  c  o     n n a i   s s a n c
8/0 7/9 6/  7/8/0 0/9   6/0 7/2/0   5/1/1/8/7/  3/3/8/1/2/

 e   r     e   p  o   r    t  s   e    n  e   m     y  a  r
0 7/9 6/0   7/6/0 5/9   6/9 7/3/0   7/1/0 7/0   9/9 0/8/9

    m   o    r    e   d    c  o  l      u   m    n s a p p
6/0 9/0 5/  9 6/0 7/0   6/2/0 5/0   0/9 1/0 9/  1/3/8/6/6/

 r   o  a    c h i n   g   n  o   r     t  h  e     r  n  m
9 6/0 5/8/  2/4/7/1/0   8/1/0 5/9   6/9 7/4/0   7/9 6/1/0

    o   u    n  t  a i   n p a s s    e  s a  t     g   o
9/0 5/9 1/  1/9 7/8/7/  1/6/8/3/3/  0 7/3/8/9   7/0 8/0 5/

 l   f    r     o   m     e   o       o      a     i      f
0 0/0 1/9   6/0 5/0 9/  0 7/0 5/5/  0 5/4/5/8/  5/7/9 5/0

    u   r     t  h  e    r    r   e    c  o  n   n a i s s
1/9 1/9 6/  9 7/4/0 7/  9 6/9 6/0   7/2/0 5/1/  1/8/7/3/3/

a n c   e      d   u   e     b   y               s   s
8/1/2/0 7/  0 6/9 1/0   7/0 3/9 0/  5/6/5/4/5/  3/5/3/9 9/

    c   o    l    b   l    a c k
9 5/2/0 5/  0 0/0 3/0   0/8/2/0 4
```

	1	2	3	4	5	6	7	8	9	0
-	n	c	s	h		p	i	a	-	-
0	f		b	k	o	d	e	g	m	l
9	u					r	t			y

f. Several things remain to be done to complete the solution. The columns can be rearranged to recover a keyword in the top row and alphabetical progression in the next two rows. Additionally, there are two unrecovered segments of text. Both of them include a number of 5s, and the preceding text in each case suggests numbers. The solution is that there is another row in the matrix with the 5 as its coordinate. It was not used enough to select from frequency alone, but once enough text was recovered, the structure can be seen. The added row includes the numbers. The complete solution appears in the next example, with the recovery of specific numbers only tentative.

```
a   e   r    i a l   r    e  c    o   n n a i   s s a n c
8/0 7/9 6/   7/8/0 0/9    6/0 7/2/0   5/1/1/8/7/  3/3/8/1/2/

 e   r   e   p  o    r    t  s    e   n  e  m    y  a   r
0 7/9 6/0    7/6/0 5/9    6/9 7/3/0   7/1/0 7/0   9/9 0/8/9

  m   o    r   e     d    c  o    l   u   m     n s a p p
6/0 9/0 5/   9 6/0 7/0    6/2/0 5/0   0/9 1/0 9/  1/3/8/6/6/

 r   o  a    c h i n     g  n  o   r   t  h  e    r  n  m
9 6/0 5/8/   2/4/7/1/0    8/1/0 5/9   6/9 7/4/0   7/9 6/1/0

  o   u      n  t a i    n p a s s    e  s a t    g   o
9/0 5/9 1/   1/9 7/8/7/   1/6/8/3/3/  0 7/3/8/9   7/0 8/0 5/

 l  f   r      o   m     e   o   7    6   4        2  .  f
0 0/0 1/9    6/0 5/0 9/   0 7/0 5/5   0/5 4/5 8/  5 7/9 5/0

  u   r      t   h  e    r   r   e    c  o   n    n a i s s
1/9 1/9 6/   9 7/4/0 7/   9 6/9 6/0   7/2/0 5/1/  1/8/7/3/3/

a n c e      d   u   e    b   y       1   6   0    0   z
8/1/2/0 7/   0 6/9 1/0    7/0 3/9 0/  5 6/5 4/5   3/5 3/9 9

. c  o       l   b   l    a c  k
9 5/2/0 5/   0 0/0 3/0    0/8/2/0 4
```

	3	6	7	1	8	2	4	0	9	5
-	s	p	i	n	a	c	h	-	-	-
0	b	d	e	f	g	j	k	l	m	o
9	q	r	t	u	v	w	x	y	z	.
5	0	1	2	3	4	5	6	7	8	9

5-8. Application of Vowel-Consonant Relationships to Multiliterals

Vowel-consonant relationship solutions can be applied to multiliterals, too. As long as you can determine the coordinates of the matrix, you can set up a dummy matrix with any sequence of characters inside as a pseudoplain component. You then reduce the cryptogram to unilateral terms by deciphering with the dummy matrix. Next, solve the resulting unilateral cryptogram using any of the techniques learned with unilateral systems, including the use of trilateral frequency counts and the vowel and consonant lines.

5-9. Solution of Trilateral and Trinomic Systems

Trilateral and trinomic systems are solved in exactly the same way as biliterals and dinomics. The systems are identified by the tendency of messages to break into groups of three instead of groups of two. With simple triliterals and trinomics, positional limitation is even more evident than it is for biliterals and dinomics. Look for a limited set of pairs of characters as either the first pair of characters or the last pair of characters in every three, Once these are found, set up your coordinates and solve as before.

Section III

Analysis of Variant Multilateral Systems

5-10. Identification of Variant Systems

As with any coordinate system, analysis of variant multilateral systems begins with determination of the coordinates. If the product of the row and column coordinates is 50 or more, the system is almost certainly a variant system of some kind.

5-10. Analysis of External Variant Systems - Frequency Matching

External variant systems are generally easier to solve than internal variant systems. Frequency counts can usually be used to determine which coordinates combine with each other on the same row or column, whenever the text is long enough to give a good representative sample, as shown in the next problem.

```
IIUC RAPC OIPU IANU NMDR   NIRI ISIU AIII PSPR AUUN
AMDG ANPG URDU IMMA PRAU   MROU RIIM NAMO ICDN UUUA
UIOM ARAA AIII DSMI RRNO   MMPU RGUR UNDS NIIA RMMA
PSUC UONM IOAR RADU PUPG   OCIA PUMO RCMM MCDR ROIA
SORI ACNM UNRI IMII SMRA   ANNA SRNM ROMI NONR RAUC

RIPN SADG AUPR IONA DUUU   MRIA OGNR RAIR MAIA RGNI
MOPO RAMM MUII DRPS MIAR   MOAC DGUA URAC NISR NOIG
DSSI RORM MINO MURU MMAI   DOUA PGRR USXX
```

	A	C	G	I	M	N	O	R	S	U
A	1	3		3	1	2		3		3
D			3			1	1	3	3	3
I	6	1	1	5	3		2	1	1	1
M	3	1		4	4		4	2		2
N	3			4	4		4	2		1
O		1	1	1	1					1
P		1	3			1	1	3	3	4
R	6	1	2	5	2		3	2		1
S	1			1	1		1	2		
U	3	3		1		3	1	3	1	2

a. The cryptogram used 10 different letters as row coordinates and 10 different letters as column coordinates. Using these coordinates, a digraphic frequency count has been completed as shown. For example, the letter I is paired with itself five times, so the number 5 appears in the matrix at the point where the row and column of I intersect.

b. Examining the frequency count, we can see that there are good frequency pattern matches between certain rows and certain columns. For example, the I row and the R row are nearly identical. Similarly, the A column and the I column are nearly identical. Carrying this process further, we can match the row pairs, AU, DP, IR, MN, and OS. The column pairs are AI, CN, GS, MO, and RU. At this point, we have no idea in what order the coordinate pairs belong or which letter in each pair comes first or if it even matters which letter comes first. We have enough information, however, to reduce the cryptogram to unilateral terms.

c. To reduce the cryptogram to unilateral terms, we set up a matrix with the combined coordinates and write any sequence of letters within it, for example, A through Y.

	A I	C N	G S	M O	R U
AU	A	B	C	D	E
DP	F	G	H	I	J
IR	K	L	M	N	O
MN	P	Q	R	S	T
OS	U	V	W	X	Y

```
K B  K G  U J  K T  S J   P K  M O  A K  H J  E B
IIUC RAPC OIPU IANU NMDR  NIRI ISIU AIII PSPR AUUN

D H  B H  E J  N P  J E   T Y  K N  P S  L G  E A
AMDG ANPG URDU IMMA PRAU  MROU RIIM NAMO ICDN UUUA

A X  E A  A K  H P  O S   S J  M E  B H  P K  N P
UIOM ARAA AIII DSMI RRNO  MMPU RGUR UNDS NIIA RMMA

H B  D S  N E  K J  J H   V K  J S  L S  Q J  N K
PSUC UONM IOAR RADU PUPG  OCIA PUMO RCMM MCDR ROIA

X K  B S  B K  N K  X K   B P  Y S  N P  S T  K B
SORI ACNM UNRI IMII SMRA  ANNA SRNM ROMI NONR RAUC


K G  U H  E J  N P  J E   T K  W T  K O  P K  M P
RIPN SADG AUPR IONA DUUU  MRIA OGNR RAIR MAIA RGNI

S I  K S  T K  J H  P E   S B  H A  E B  P Y  S M
MOPO RAMM MUII DRPS MIAR  MOAC DGUA URAC NISR NOIG

H U  N N  P S  T O  S A   I A  H O  C
DSSI RORM MINO MURU MMAI  DOUA PGRR USXX
```

d. We see that repeats appear in the pseudotext that results from our trial decipherment. The repeats that were suppressed by the variants are now visible with the variants combined. The recovery of the plaintext is like any of the previous problems. When we recover the plaintext and enter the recovered values in the matrix in place of the trial sequence, we reach the solution shown below.

	A	C	G	M	R
	I	N	S	O	U
AU	l	n	k	g	i
DP	-	m	a	b	r
IR	e	f	d	s	c
MN	t	u	-	o	p
OS	y	z	x	v	w

```
e n   e m   y r   e p   o r     t e   d c   l e   a r   i n
K B   K G   U J   K T   S J     P K   M O   A K   H J   E B
IIUC  RAPC  OIPU  IANU  NMDR    NIRI  ISIU  AIII  PSPR  AUUN

g a   n a   i r   s t   r i     p w   e s   t o   f m   i l
D H   B H   E J   N P   J E     T Y   K N   P S   L G   E A
AMDG  ANPG  URDU  IMMA  PRAU    MROU  RIIM  NAMO  ICDN  UUUA

l v   i l   l e   a t   c o     o r   d i   n a   t e   s t
A X   E A   A K   H P   O S     S J   M E   B H   P K   N P
UIOM  ARAA  AIII  DSMI  RRNO    MMPU  RGUR  UNDS  NIIA  RMMA

a n   g o   s i   e r   r a     z e   r o   f o   u r   s e
H B   D S   N E   K J   J H     V K   J S   L S   Q J   N K
PSUC  UONM  IOAR  RADU  PUPG    OCIA  PUMO  RCMM  MCDR  ROIA

v e   n o   n e   s e   v e     n t   w o   s t   o p   e n
X K   B S   B K   N K   X K     B P   Y S   N P   S T   K B
SORI  ACNM  UNRI  IMII  SMRA    ANNA  SRNM  ROMI  NONR  RAUC


e m   y a   i r   s t   r i     p e   x p   e c   t e   d t
K G   U H   E J   N P   J E     T K   W T   K O   P K   M P
RIPN  SADG  AUPR  IONA  DUUU    MRIA  OGNR  RAIR  MAIA  RGNI

o b   e o   p e   r a   t i     o n   a l   i n   t w   o d
S I   K S   T K   J H   P E     S B   H A   E B   P Y   S M
MOPO  RAMM  MUII  DRPS  MIAR    MOAC  DGUA  URAC  NISR  NOIG

a y   s s   t o   p c   o l     b l   a c   k
H U   N N   P S   T O   S A     I A   H O   C
DSSI  RORM  MINO  MURU  MMAI    DOUA  PGRR  USXX
```

e. With the plaintext values filled into the matrix, we can see in what order the rows and columns belong. Starting with the last row of the internals, we rearrange the columns of the matrix in alphabetic order.

	M	R	G	A	C
	O	U	S	I	N
AU	g	i	k	l	n
DP	b	r	a	-	m
IR	s	c	d	e	f
MN	o	p	-	t	u
OS	v	w	x	y	z

The first row of the internals should follow alphabetically after the third row—scdef, gikln.

	M	R	G	A	C
	O	U	S	I	N
DP	b	r	a	-	m
IR	s	c	d	e	f
AU	g	i	k	l	n
MN	o	p	-	t	u
OS	v	w	x	y	z

f. All that remains is to fill in the missing letters H, J, and Q in the plaintext sequence, and to try to recognize how the coordinates were constructed. As mentioned earlier, it is common practice to couple I with J or U with V when using a 5 by 5 matrix. Since J did not appear in the plaintext, we may assume it occupies an alphabetical position within the I block. The Q clearly belongs between the P and T, leaving the H in the top row. The plaintext keyword is BRAHMS (the classical composer). With that as a clue, the letters in the coordinates are shifted to their correct positions, revealing the keywords PIANO, DRUMS, MUSIC, and ORGAN.

	M	U	S	I	C
	O	R	G	A	N
PD	b	r	a	h	m
IR	s	c	d	e	f
AU	g	i/j	k	l	n
NM	o	p	q	t	u
OS	v	w	x	y	z

5-12. Analysis of Variants - Isologs

Two or more encrypted messages with different encrypted text, but the same underlying plaintext are called isologs. When isologs are encountered, your job is much easier. Isologs are particularly useful in solving variant multilateral systems, either external or internal.

a. Isologs can be recognized by one or more of these characteristics—

- Identical message lengths.
- Similar characteristics in the text, such as repeated segments or characters occurring in the same position in each message.

- External indications, such as identical times of file or identical message numbers included in the header for each message. Normally, no two different messages from the same sender receive the same file time or message number. When you see the same time of file on the same date originating from the same unit, the messages are likely to be isologs.

b. Two messages that showed the same time of file in the message header appear in Figure 5-3.

Message 1:

XLNH GVDV NZRH DKXH AMNV RPGZ XMNK DZGP XVDH QHNB
QCFH DVRP GLFP DSAZ RHFB GKNZ DBFL DLGH RSFH QKRB
TSDP QVNK DZFP DKQP QMAC NBRL RPRK NSRV NBFL FBNP
DBLM FZGV ACRK TCTH XPTM AHNL NMRM DBFS FHRH NCRZ
XCFV NBRL FPTS DHGK NKDZ FHNV

Message 2:

GYQB EDAD QTOW ATZM OPFT GSAY OTFD ZDKW KYZY VSQD
EWOS ATGW KTGS FMKP OWFS LTQT ZDEM ARVS ERGW LDFW
OYZB LTFT ZTOS FDVW EWOH QDLR GSZS AMQS QTLM FWQY
ZDGH AWET GPZW GTQM ZRGD EPFM EYKM QTLM GSGW LBAS
OTQW ZTER GWGB QBED ADZD OSAT

Figure 5-3. Isolog example.

c. Each message shows positional limitations. Message 1 has the letters ADFGLNQRTX in the row coordinate position and BCHKLMPSVZ in the column coordinate position. Message 2 has AEFGKLOQVZ in the row coordinate position and BDHMPRSTWY in the column coordinate position. The two messages are not encrypted in the same system, but they appear to be isologs.

d. The initial step in solving these isologs is to see what values equate to each other in the two messages. Pick one of the most frequent digraphs in either message as a starting point. For example, FH occurs four times in the first message. A frequency count, while not strictly necessary, may be helpful in spotting the most common values. The digraphs that occur in the same positions in message 2 as FH in message 1 are OS, GW, GS, and another OS.

e. The next step is to find each of the digraphs in message 2 that equated to FH from message 1. The letters OS, GW, and GS in message 2 and the digraphs in the same position in message 1 are underlined in Figure 5-3.

f. We now see that RH, RP, FP, and FH in message 1 equate to GS, GW, and OS in message 2. A check of the new values in message 1 adds the additional digraph OW in message 2, completing the equations for that set. It appears that R and F are variant row coordinates and P and H are variant column coordinates in message 1. Similarly, the message 2 variants are G and O on the rows and W and S on the columns.

g. Continue the process by picking additional repeated values. Complete the equations for each, working back and forth between the two messages, just as we did for the initial digraph FH. Continue until all coordinates have been combined, or you run out of digraphs to compare. You can set up a plot to keep track of the equations as shown in the next example.

Row	Column	Message 1	Message 2	Row	Column
FR	HP	RH RP FP FH	GS GW OS OW	GO	SW
DN	BZ	DZ NZ DB NB	QD QT ZT ZD	QZ	DT
	KV	NV DV DK NK	FD FT AD AT	AF	
GQ		QK QV GK GV	ED ET LT LD	EL	
AL	CM	AM LM AC	OH GP GH OP		HP
TX	LS	XL TS	GB OY GY		BY
		GP GH QP QH	VS VW KW	KV	
		DS DL NS NL	FM AM AR		MR

h. Other combinations could have been selected than the ones shown, but these are sufficient to show all the variants in both matrices. From this point, either message can be reduced to unilateral terms and solved. Then the recovered plaintext can be applied to the other message to complete the recovery of the second matrix. Note that if the same matrix was used in both messages, the similarity should be quickly recognized and the solution accomplished more easily. The next paragraph shows the simpler technique when the same matrix is used.

5-13. Solution Using Isologous Segments

Segments of ciphertext which have the same underlying plaintext are known as isologous segments. A technique similar to the one used in isolog solution can be used any time repeated plaintext can be identified. This is likely to occur with repeated beginnings and endings to messages or with long repeated words and phrases.

a. Recognizing repeated plaintext in variant systems requires painstaking inspection of the ciphertext. Computer indexes of repeated plaintext, which show repeated text on consecutive lines along with the preceding and following text makes repeats easier to recognize. In any long plaintext repeat, some of the ciphertext digraphs or dinomes are likely to repeat. Other ciphertext digraphs or dinomes are likely to show common row or column coordinates. Pairs with neither row nor column coordinates in common will generally be in the minority. Therefore, although a lot of trial and error may be involved, the longer repeated plaintext segments can often be identified. Consider the two message beginnings shown below.

Message 1:

```
3469 8489 2469 1420 8957  7238 2311 8840 9626 6269
1429 1622 8924 ...
```

Message 2:

```
3368 6389 2468 1335 8807  7238 2316 6890 9636 6788
7338 7127 6934 ...
```

b. The similarities of the text make it quite clear that the underlying plaintext is the same in both cases, and the same matrix is used for both. Proceeding on the assumption that the plaintext and matrix are the same, it is easy to match the remaining values to determine the variants. For example, from the first dinome in each message, 3 and 4 are column variants. From the second dinome in each message, 8 and 9 are column variants. All the variants can be combined from this short example, and the remainder of the solution is routine.

5-14. Analysis of Internal Variant Systems

Internal variant systems are generally more difficult to solve than external variant systems. With no coordinates to combine, frequency counts do not provide immediate clues to variants. Similarly, isologous segments are harder to recognize. Some characters are likely to repeat in isologous segments with internal variant systems, but the partial repeats caused by common row or column coordinates are much less likely to occur. Still, given enough messages from a single system to produce repeats; given operator carelessness in encryption; or given stereotyped traffic, these systems can readily be solved, too. Once a plaintext entry is found, the remainder of a solution is not difficult. When you find isologs or isologous segments, you can equate ciphertext values just as was demonstrated in the internal variant examples. The only difference is that you do not combine coordinates through this process, but instead find all cells in the matrix that have the same plaintext value.

5-15. Analysis of Syllabary Squares

Syllabary squares are closely related to small code charts, and the solution of both types of systems is similar. The analysis of syllabary squares produces some distinct differences.

a. Isologs or isologous segments are not necessarily the same length in each case. The encipherment examples below are repeated from paragraph 5-3e.

	6	0	4	3	8	1	7	5	9	2
8	a	1	ad	al	an	and	as	at	b	2
4	c	3	ce	co	d	4	da	de	di	e
3	5	ea	ec	ed	ee	ei	el	en	ent	er
7	es	et	f	6	fi	fo	g	7	h	8
2	hi	ht	i	9	in	ing	io	ir	is	it
0	h	0	00	k	l	la	le	ll	m	ma
5	n	nd	ne	ng	ni	nt	o	on	or	ou
9	p	q	r	ra	re	ri	ro	rs	rt	s
1	se	si	st	t	ta	te	th	ti	tion	to
6	tw	ty	u	ur	v	ve	w	x	y	z

p: r ei n fo r ce m en t s
c: 94 31 56 71 94 44 09 35 13 92

p: re in f or ce m ent s
c: 98 28 74 59 44 09 39 92

b. Isologous segments can often still be recognized by the plaintext values which have no variation. In the example, there is only one way to encipher the letters M and S. When *REINFORCEMENTS* is enciphered, the ciphertext equivalents of M and S will always be the same. Other values are likely to begin with the same row coordinate, since syllables beginning with the same letter are likely to be on the same row, such as the R and the RE. Still others will have a possible variation, but the variation will not be used. The repeated CE syllable in both segments is an example of this. As a result of all these considerations, isologous segments are often recognizable and provide a point of entry to the system.

c. Solution of syllabary spelling will be further explained in Part Six, Analysis of Code Systems.