

AD-A046 887

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO
SECURE COMMERCIAL DIGITAL COMMUNICATIONS.(U)
JUL 77 P V ABENE

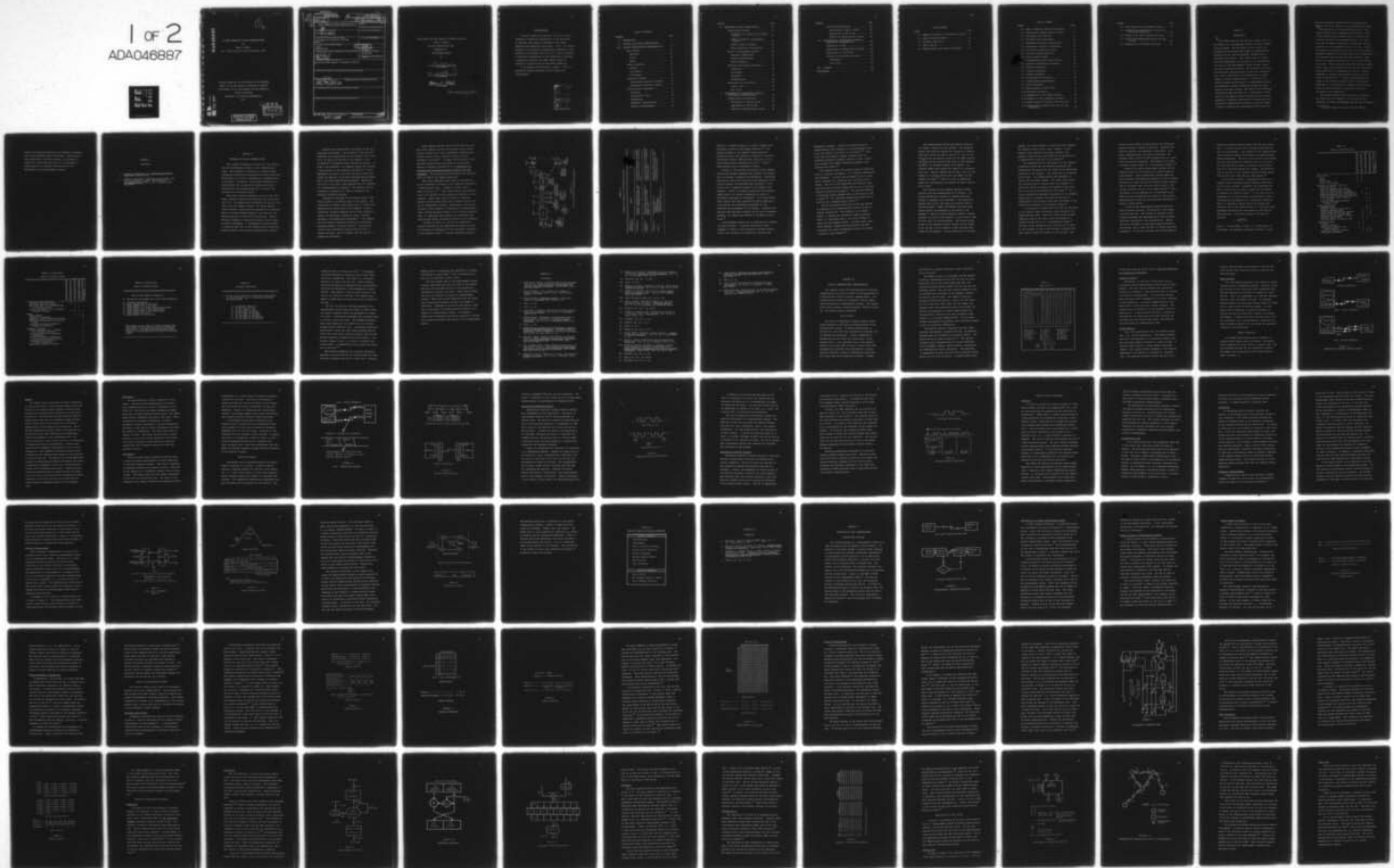
F/G 17/2

UNCLASSIFIED

AFIT-CI-77-83

NL

1 of 2
ADAO46887



77-83

① SC

AD A 0 46887

SECURE COMMERCIAL DIGITAL COMMUNICATIONS

by

Peter V. Abene

B.S., North Carolina State University, 1970

A thesis submitted to the Faculty of the Graduate School of the University of Colorado in partial fulfillment of the requirements for the degree of Master of Science

Department of Electrical Engineering

1977

AD No. _____
DDC FILE COPY

DDC
RECEIVED
NOV 28 1977
RECEIVED
B

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENT

AGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

14
AFIT

1. REPORT NUMBER
CI-77-83

2. GOVT ACCESSION NO. 3. RECIPIENT'S CATALOG NUMBER
9 Master's Thesis

4. TITLE (and Subtitle)
Secure Commercial Digital Communication

5. TYPE OF REPORT & PERIOD COVERED
Thesis

7. AUTHOR
Captain Peter V. Abene

6. PERFORMING ORG. REPORT NUMBER

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
AFIT Student at University of Colorado,
Boulder CO

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
AFIT/CI
WPAFB OH 45433

13. REPORT DATE
July 1977

12. NUMBER OF PAGES
131 pages

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)
12 141 p.

15. SECURITY CLASS. (of this report)
Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)
Approved for Public Release: Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES
APPROVED FOR PUBLIC RELEASE AFR 190-17.
JERRAL F. GUESS, Captain, USAF
Director of Information, AFIT

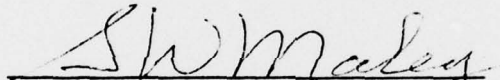
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

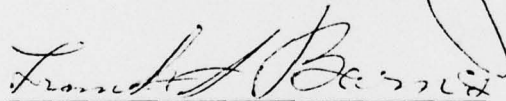
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

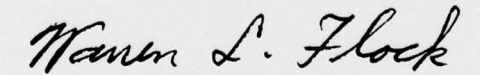
012 200

JFB

This Thesis for the Master of Science Degree by
Peter V. Abene
has been approved for the
Committee of
Telecommunications
by


S.W. Maley


F.S. Barnes


W.L. Flock

Date 15 July 1977

ACKNOWLEDGMENT

I wish to express my gratitude to Dr. S.W. Maley, Professor of Electrical Engineering of the University of Colorado, who served as chairman of the thesis committee and supervised this thesis. To Dr. F.S. Barnes, Chairman of the Electrical Engineering Department of the University of Colorado, and to Dr. W.L. Flock, Professor of Electrical Engineering of the University of Colorado; I express my gratitude for their having served as readers of the thesis and on the thesis committee.

To my patient and loving wife, Dianne; I wish to express my sincere gratitude for her support and encouragement.

ACCESSION for		
NTIS	NTIS Section	<input checked="" type="checkbox"/>
BDC	BDC Section	<input type="checkbox"/>
UNANNOUNCED <input type="checkbox"/>		
JUSTIFICATION		
BY		
DISTRIBUTION/AVAILABILITY CODES		
Dist.	AVAIL. ORG./	SPECIAL
A		

TABLE OF CONTENTS

CHAPTER	PAGE
I. INTRODUCTION	1
II. RATIONAL FOR SECURE COMMUNICATIONS	5
III. DIGITAL COMMUNICATIONS CHARACTERISTICS . .	25
Data Transfer	25
Character	28
Stream	28
Block	29
Channel Operation.	29
Simplex	31
Half-Duplex	32
Full-Duplex	32
Character Transfer	33
Asynchronous character transfer . .	36
Synchronous character transfer . .	38
Channel control Procedures	41
Protocols	41
Uncontrolled links	42
Echoplexing	43
Sequential acknowledgment	43
Parallel acknowledgment	45

CHAPTER	PAGE
IV. ENCRYPTION OF DATA COMMUNICATIONS	53
Cryptographic Systems	53
Attributes of a secure cryptographic system	55
General classes of cryptographic systems	56
General types of ciphers	57
General methods of cryptanalyst	59
Types of Cryptographic Systems	60
Character substitution.	60
Stream transformations	67
Block encryption	71
FIPS PUB 46 Encryption Algorithm	74
Background.	74
Description	75
Attributes	79
Implementation	80
Application to Data Stream	82
Link by link.	82
End to end	86
V. APPLICATION OF CRYPTOGRAPHIC SYSTEM TO ASYNCHRONOUS COMMUNICATIONS	91
Uncontrolled Link Operation	92
Description of typical system	92
Application of PUB 46 DES	93
Effects on communications system	99

CHAPTER	PAGE
Controlled Asynchronous	100
Description of typical system	100
Application of PUB 46 DES	103
Effects on communications systems	106
VI. APPLICATION OF CRYPTOGRAPHIC SYSTEM TO SYNCHRONOUS COMMUNICATIONS	109
Description of SDLC	109
Description of Communications System	114
Application of PUB 46 DES	116
Effects on Communications System	122
Throughput	122
Error rate	123
VII. SUMMARY	126
BIBLIOGRAPHY	128

LIST OF TABLES

TABLE		PAGE
2.1	Summary of Threats to Information Privacy.	9
2.2	Types of Security Risks.	16
2.3	Potential Risk Costs	19
3.1	ASCII Code Set	27
3.2	Message Header and Trailer Parameters. . .	51

LIST OF FIGURES

FIGURE		PAGE
2.1	Risk to Computer Information Security . . .	8
3.1	Communication Channel Operation Modes . . .	30
3.2	Full-Duplex Data Transfer	34
3.3	300 Baud 2-Wire Full-Duplex	35
3.4	Asynchronous Character Transmission	37
3.5	Synchronous Character Transmission.	40
3.6	Full-Duplex Sequential Control	46
3.7	Satellite Channel Operation	47
3.8	Typical Message Format	49
4.1	Cryptographic System Block Diagram.	54
4.2	Character Substitution Encryption	58
4.3	Additive Encryption	62
4.4	Playfair Encryption	63
4.5	Autokey Encryption	64
4.6	Polyalphabetic Encryption	66
4.7	Pseudorandom Encoder/Decoder	70
4.8	DeVries Encryption	73
4.9	Block Diagram of PUB 46 DES	76
4.10	Encryption Iteration.	77
4.11	Computation of the Cipher Function.	78
4.12	Examples of Data Encryption Standard. . . .	81
4.13	Block Diagram of Fairchild 9414 Chip Set. .	83
4.14	Communications Network with Link by Link Encryption	84

FIGURE	PAGE
5.1 Block Diagram of Cryptographic Device. . .	96
5.2 Asynchronous Communications System with End to End Encryption	101
6.1 Fields of the SDLC Transmission Frame. . .	111
6.2 PUB 46 DES Cryptographic Device for SDLC Protocol	115
6.3 Transmission of Message Using SDLC	117

CHAPTER I

INTRODUCTION

↙ This thesis examines the need for, applications of, and effects of a cryptographic system on digital communications systems. The discussion on the rationale for encrypting data examines the threats to communications security and the role cryptographic systems have in countering the threats. This thesis does not address the legal requirements for security as the requirements are still in the developmental stages by both federal and state legislative bodies. The discussion of digital communications characteristics outlines the parameters which are of importance to the application of cryptography to digital communications. A review of the general methods and principles of cryptography provides an examination of past methods and reviews the weaknesses of the past systems. → The Federal Data Encryption Standard provides the bases for a cryptographic system that is to be added to existing communications systems. Examples of potential applications to asynchronous and synchronous communications protocols provide an example of how to interface the cryptographic system based on

the Data Encryption Standard and the communications systems and of the effects on the communications system.

The application of cryptographic systems to voice communications is not addressed in this thesis. Many of the methods used to protect digital communications could be applied to voice communications. The voice data must be sampled, quantized, digitalized, and then encrypted at the transmission end of the voice circuit. The sampling rate should be at least twice the highest frequency component of the voice data to be transmitted. As telephone voice channels are band limited to 4000 hertz, the sampling rate must be at least 8000 samples per second. There should be around 128 levels of quantization; thus, seven digital binary bits are required or about 56000 bits of information per second. Typical bandwidth to transmit this amount of data are of the order of 50000 hertz.^{1*} Digital encryption does not appear to be currently possible within the existing bandwidth of voice channels in common carrier systems. There are analog methods of encrypting voice data discussed in the literature.²

The digital communications protocols discussed in the thesis are not intended to be the best or only protocols for which cryptographic systems may be applied.

* Footnotes appear at the end of each chapter.

Rather, the discussed protocols are intended to represent part of the possible range of protocols. These applications assume that risk/cost analysis, as discussed in Chapter II, established the need for a cryptographic system based on an economic justification of the application of the cryptographic system.

CHAPTER I

Footnotes

1. Technical User Manual II, Communications Theory,
Rixon Inc., 1973, p. 1-8.
2. Arnold M. McCalmont, "Communications Security for
Voice-Techniques, Systems, and Operations,"
Telecommunications, Vol. 7, No. 4 (April, 1973)
pp. 35-42.

CHAPTER II

RATIONAL FOR SECURE COMMUNICATIONS

This chapter develops the rational for the application of cryptographic systems to data communications links. The information contained in a computer based information system must be protected from any threat that would result in a loss of availability, confidentiality, or integrity. In general, the security objectives of a secure system are to protect the system performance, availability, and responsiveness and to restrict the information to authorized users.¹

Each threat could be countered in one of four ways. Threat avoidance removes the threatened item from potential risk. An example would be removing the payroll file from an on-line computer system. The second method, threat transfer, occurs with the purchase of an insurance policy to protect against potential loss from the risk. Threat retention is a form of self-insurance. Threat reduction attempts to reduce the potential loss through protective safeguards. A cryptographic system attempts to close the open door to the computer based information system created by the connected communications system.²

Computers are becoming more interwoven in the daily operations of business. The estimates of the degree of dependence that business has on the computer vary ranging up to "as much as 90 percent of the company's vital information"³ on computer based information systems. In order to be economical, the computer system must process a large portion of the company's information.⁴ Future dependence will grow in the next few years as more computers are employed in the daily decision making process and in the actual manufacturing process.⁵ Within the next five to ten years, most computers will be connected to telecommunications networks.⁶ The computer and the information contained in it must be protected from potential disruptive threats.

Information contained in the computer based information system is of immense value to the company. The information is valued to the extent that it reveals the operations and future plans of the company. Typical information includes summaries and raw data relating to assets, raw materials, production, sales, finances, personnel, and research and development.⁷ The computer may also contain information relating to corporation processes, patents, and trade secrets.⁸ The monetary value of this information results from the cost to replace the data, loss of business, and the loss of a competitive advantage.

These computer systems must be protected from risks that would threaten the use and accuracy of the system. Threats include fire, flood, earthquake, communications failure, power failure, computer hardware failure, intruders, and misuse.⁹ A company should evaluate these and other threats using such guides as the National Bureau of Standards publication 31 on Guidelines for Automatic Data Processing Physical Security and Risk Management. This evaluation should address all potential risks in view of the company operations.

This thesis addresses those risks to the security of the information contained in the computer connected to a communications link. A summary of these risks is shown in Figure 2.1.¹⁰ This figure depicts the potential risk to the security of the information contained in the computer. These risks cover the range of potential hazards including those which can be reduced through the use of cryptographic systems. Specifically, the threats relating to the transfer of information on communications links, through switching centers, and to remote terminals. Table 2.1 summarizes these threats and illustrates how these threats effect information security of the computer.

As shown in Table 2.1, the risk to information security incurred by the communications system can lead to disclosure or corruption of the information contained in the computer system.¹¹ "The most dangerous external

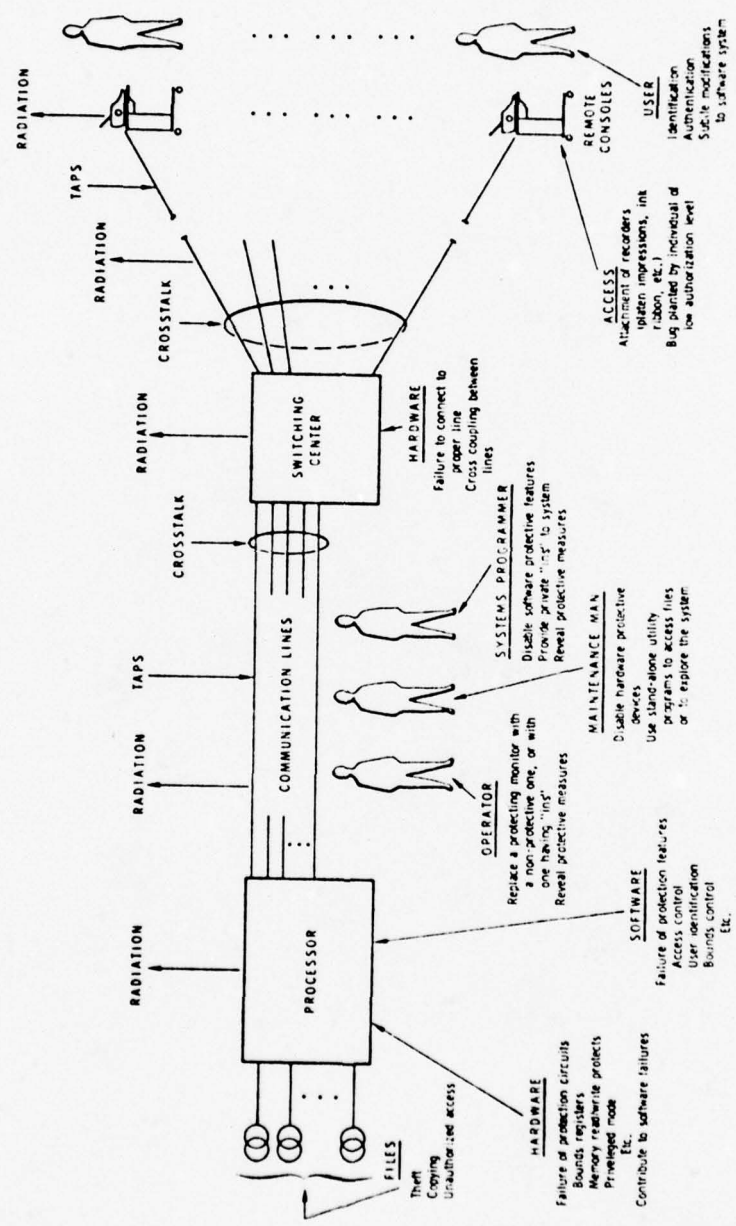


FIGURE 2.1
RISK TO COMPUTER INFORMATION SECURITY

TABLE 2.1
 Summary of Threats to Information Privacy

Nature of Infiltration	Means	Effects
Accidental	Computer malfunctioning; user errors; undebugged programs	Privileged information dumped at wrong terminals, printouts, etc.
Deliberate Passive	Wiretapping, electromagnetic pickup, examining carbon papers, etc.	User's interest in information revealed; content of communications revealed
Deliberate Active	Entering files by: "Browsing" "Masquerading" "Between lines" "Piggy-back" penetration	Specific information revealed or modified as a result of infiltrator's actions

threat to a computer system is a clever, computer wise individual sitting at some remote terminal."¹² The security problem of the computer system increases dramatically once the computer is connected to external environment through communications links with dial-up access specifically increasing the risk.¹³ Each of these threats will be discussed in detail.

Threats to the security and privacy of the communication links between computers and terminals are divided into two classes, accidental and deliberate. Accidental disclosures result from errors by human operators of the various computer and communications equipment or from errors made in computer programs by the computer programmers. Failure of hardware associated with computer, communication, or terminal equipment can result in accidental disclosure of information. To a large extent, these threats have been countered by better administrative controls and programming procedures for the operators and programmers of the systems. Computer hardware has been made more reliable and the use of redundant hardware has reduced the chances of accidental disclosures.¹⁴

The deliberate threats can be separated into passive and active threats. A passive infiltration threat attempts to listen to the information transfer process without the intention of disrupting or changing the

information exchange. "Passive infiltration may be accomplished by wire tapping or by electromagnetic pick-up of the traffic at any point in the system."¹⁵ With other than the simplest systems, computers and the connected peripheral equipment radiate in such a manner as to require very complex detection equipment, thus reducing this threat.¹⁶

Wire tapping offers the easiest method of gathering data in a passive environment. It is neither complicated nor expensive. A circuit can be bugged in one of three ways: physical tap, inductive tap, or drop-in transmitter. The use of line amplifiers and inductive taps make detection difficult except with the use of the most sensitive detection equipment.¹⁷ The sidelobe patterns of microwave antenna systems offer an easy point to tap a circuit.¹⁸ The literature extensively covers ways to actually tap a communications circuit.¹⁹

Active infiltration uses some of the same methods of attaching to the communications circuit as used in passive infiltration. The intention is to modify, insert, or disrupt the information exchange process. Table 2.1 illustrates some of the actual methods of entry into the computer system. The data codes, modulation methods, communications protocols, and data structures are highly standardized and do not present a barrier to the intruder.²⁰

The communications system and computer system do not present a barrier to the intruder. The telephone company even helps by tagging the circuits at junction boxes with origin and type of circuit.²¹ Dial up circuits provide the easiest access for the remote authorized user and the intruder from local or long distance. The caller is afforded a degree of anonymity. An unlisted number only reduces the chance of an unauthorized call. Another computer can be used to try all the numbers of a central office until the number for the computer is found.²² There is no way to physically secure the communication link except for short links in secure areas.

The computer and the computer operating system software have been demonstrated to be insecure. All serious penetration attempts of computer operating systems of computers have succeeded. Even operating systems that provide some degree of security such as IBM VS2/release 2, TENEX, and MULTICS have been penetrated without the knowledge of the owners of the systems.²³ The only current method to secure a system requires all users of the system at a given time to have the same level of trustworthiness. Each person having access to the computer has been cleared to have access to all the data in the computer at the time they have access to the computer. As the sensitivity of the data

changes, the access granted to authorized users changes; the computer access "color" level, changes. If the sensitive data requires such access controls, the computer containing the data should not be connected to a communications link that is not secured.²⁴

The responsibility for the security of the computer information system and the connected communications system rests ultimately with the owner of the information contained in the computer. The assets must be protected by the owner. The communications link equipment is rented from the phone company. The renter must assume the responsibility of insuring that the equipment is physically secure to prevent wire tapping on the equipment or wiring on the property of the renter.²⁵ In a survey conducted by Honeywell Corporation of its computer system users, 82 percent felt that the majority of responsibility for security rests with the owners of the information contained in the computer system.²⁶

A cryptographic system serves to close the open door to the computer based information system caused by the communications links. Since the links can not be made physically secure, the data on the link should be secured through the use of a cryptographic system.²⁷ The data on the encrypted circuit will not be of value to the intruder as long as the key used to encrypt the data remains secret and unknown to the intruder. The

intruder cannot modify the data because the encryption process provides a method of detection. Modified data would not decrypt into meaningful information. With proper communications link control procedures, the intruder cannot record and then playback data at a later time that has been encrypted. The message could carry a message sequence number or a time dependent element.²⁸

Currently available cryptographic systems for the commercial user do not provide truly secure communications systems. These cryptographic systems generally perform simple transformations requiring as few as twenty characters of clear text and the matching twenty characters of encrypted text to recover the key used. Once the key is found, the data is no longer secure.²⁹ More information on this weakness will be discussed later. These low level systems may be a block to truly secure communications as they may give the false impression of having a secure system.

A truly secure cryptographic system should have these following features. The security of the system rests with the key. The intruder is assumed to have complete knowledge and understanding of the encryption algorithm, application, and implementation. Second, the time to decipher an intercepted message should be sufficiently long to make the data of little value when deciphered. If the data has value for one day when the

encryption algorithm should protect the data for several days. A long key period and a nonlinear algorithm will meet this requirement. The securing of the communications links should not degrade the computer and communications system performance. The cost of the encryption unit should allow for high production quantities and low unit cost. The algorithm should allow a relatively high volume of data to be protected between key changes. The distribution of the keys to the various locations should require little effort and allow easy and rapid change.³⁰

The decision to apply a cryptographic system to a computer based information system should be made based on a risk/cost analysis. Management must determine how much it would cost to counter a threat and the amount of reduction in potential loss.³¹ As all risks are not applicable to a given situation, each potential risk is evaluated and the probability of occurrence determined. Table 2.2 provides a typical matrix to evaluate the potential risk of the listed threats.³² From this table, management determines the risk potential and then the loss potential. Using this potential loss and the equation:

$$E = \frac{10^{(P+D-3)}}{4}$$

where P is from Table 2.2 and D is from Table 2.3, the manager can determine the amount of money, E, that

TABLE 2.2
Types of Security Risks

	Inability to Process	Loss of Entire File	Loss of Single Records	Modification of Records	Unauthorized Reading or Copying
Acts of God					
Fire	2	2			
Flood	2	2			
Act of war	2	2			
Other catastrophe	2	2			
Hardware and program failures					
Computer outage	5				
Pile unit damages disk track			3		
Tape unit damages part of tape			3		
Disk, or other volume, unreadable		3			
Hardware/software error damages file		2	4	5	
Data transmission error not detected			4	6	
Card (or other input) chewed up by machine			6	5	
Error on application program damages record			4	5	
Human carelessness					
Keypunch error			4	7	
Terminal operator input error			5	7	
Computer operator error		3	4	5	
Wrong volume mounted and updated		3		3	
Wrong version of program used		3		3	
Accident during program testing		3	4	4	
Mislaid tape or disk		3			2
Physical damage to tape or disk		3	3		
Malicious damage					
Looting	2	2			
Violent sabotage	2	2			
Nonviolent sabotage (e.g., tape erasure)	2	2	3	3	
Malicious computer operator		2	3	3	
Malicious programmer		3	3	3	

TABLE 2.2 (continued)
Types of Security Risks

	<u>Inability to Process</u>	<u>Loss of Entire File</u>	<u>Loss of Single Records</u>	<u>Modification of Records</u>	<u>Unauthorized Reading or copying</u>
Malicious tape librarian	2				
Malicious terminal operator	2		3	3	
Malicious user (e.g., punches holes in returnable card)			3	3	
Playful malignancy (e.g., misusing terminal for fun)	2		3	3	3
Crime					
Embezzlement			3	4	4
Industrial espionage					3
Employees selling commercial secrets					3
Employees selling data for mailing lists					3
Data-bank information used for bribery or extortion					3
Invasion of privacy					
Casual curiosity (e.g., finding out employee salaries)					4
Looking up data of a competing corporation					4
Obtaining personal information for political or legal reasons					3
Nondeliberate revealing of private information					4
Malicious invasion of privacy					

TABLE 2.2 (continued)
Types of Security Risks

Key to the numbers in Table 2.2

P: Rating for the probability of an event occurring:

- 0: Virtually impossible
- 1: Might happen once in 400 years
- 2: Might happen once in 40 years
- 3: Might happen in 4 years (1000 working days)
- 4: Might happen once in 100 working days
- 5: Might happen once in 10 working days
- 6: Might happen once a day
- 7: Might happen 10 times a day

*The numbers in the table are merely examples and would have to be evaluated anew for any specific application. One data processing manager rated several of the probabilities an order of magnitude higher.

TABLE 2.3
Potential Risk Costs

D: Rating for the amount of damage the event causes
in lost business, cost of correcting the data,
and other costs:

- 0: Negligible (about \$1)
 - 1: On the order of \$10
 - 2: On the order of \$100
 - 3: On the order of \$1,000
 - 4: On the order of \$10,000
 - 5: On the order of \$100,000
 - 6: On the order of \$1,000,000
 - 7: On the order of \$10,000,000
-

should be spent to counter the risk.³³ For example, the manager decides to reduce the risk of data being stolen by a competitor. From Table 2.2, the threat potential is found to be four (4). After reviewing operation procedures, he determines that the potential loss from the disclosure of the information is about \$10,000. Using Table 2.3 and the above equation, the yearly risk exposure is \$25,000. The company could expect to spend about this much each year to avoid the risk.

The cost of currently available equipment for a cryptographic system ranges from \$1000 to \$5000. There are several companies making the equipment for commercial users. International Telephone and Telegraph makes an offline device for \$2700. The DATOTEK Corporation makes three models with the model DC-110 for online systems costing \$3000 per unit. Ground/Data Corporation manufactures a model for time sharing systems costing \$1300 (this model uses a Read Only Memory, ROM, for key storage making key changes difficult and costly). The European company, Crypto AG, makes an automated unit costing \$5200. A communication link requires two units, one at each end.³⁴

Manufacturing companies are currently developing equipment using the FIPS Pub 46 algorithm that will provide much increased security at a lower cost. Motorola

Company plans to incorporate the algorithm in a single card module for about \$500.³⁵ This algorithm and its uses will be examined in detail later.

The total system cost of the cryptographic system involves cost other than just the costs of the devices. The total cost includes the cost of key generation, distribution, storage, and issue. These costs include the costs of personnel to manage the cryptographic systems. There will be cost associated with the training of operations personnel and with the changes in procedures. Depending on the actual application, there may be additional cost associated with changes in the computer or communications systems. For example, a higher data transmission rate may be required to include the overhead associated with the use of the cryptographic system.

CHAPTER II

Footnotes

1. Harrison R. Burris, "Computer Network Cryptographic Engineering," AFIPS Conference Proceedings 1976 National Computer Conference, AFIPS PRESS, 1976, Vol. 45, p. 91.
2. David Firmbery, "Your Computer in Jeopardy," Computer Decisions, Vol. 8, No. 7 (July 1976), p. 29.
3. Peter Hamilton, Computer Security, Associated Business Programmers Ltd., 1972, p. 27.
4. Ibid., p. 28.
5. Ibid., p. 27.
6. Paul Armer, "Computer Technology and Surveillance," Computers and People, Vol. 24, No. 9 (September 1975), p. 9.
7. Belden Menkus, "Management's Responsibilities for Safeguarding Information," Journal of Systems Management, Vol. 27, No. 12 (December 1976), p.33.
8. Ibid., p. 34.
9. Guidelines for Automatic Data Processing, Physical Security and Risk Management, National Bureau of Standards, Federal Information Processing Standards Publication 31, June 1974, p. 11.
10. Willis H. Ware, "Security and Privacy in Computer Systems," AFIPS Conference Proceedings 1967 Spring Joint Computer Conference, AFIPS Press, 1967, Vol. 30, p. 280.
11. H.E. Petersen and R. Turn, "Systems Implications of Information Privacy," AFIPS Conference Proceedings 1967 Spring Joint Computer Conference, AFIPS PRESS, 1967, Vol. 30, p. 28.
12. Heather M. David, "Computers, Privacy, and Security," Computer Decisions, Vol. 7, No. 2 (February 1975), p. 15.

13. Robert H. Milligan, "Management Guide to Computer Protection," Journal of System Management, Vol. 27, No. 11 (November 1976), p. 15.
14. Petersen, Op. cit., p. 291.
15. Ibid., p. 291.
16. Heather M. David, "Computers, Privacy, and Security," Computer Decisions, Vol. 6, No. 5 (May 1974), p.46.
17. Stephen W. Seibholz and Louis D. Wilson, User's Guide to Computer Crime, Chilton Book Company, 1974, pp. 42-49.
18. David (February 1975), Op. cit.,p. 28.
19. John F. Mason, "Designers Compete for the Snug Electronic Bug in a Rug," Electronic Design Vol. 21, No. 20 (September 27, 1973) pp. 22-30.
20. Seibholz, Op. cit., pp. 42-49.
21. Richard G. Cannind, ed., "Integrity and Security of Personnel Data," EDP Analyzer, Vol. 14, No. 4 (April 1976), p.4.
22. Seibholz, Op. cit., p. 42.
23. Cannind, Op. cit., p. 9.
24. Ibid., p. 10.
25. Seibholz, Op. cit., p. 51.
26. Jerome Sobel, "Planning a Secure System," Journal of Systems Management, Vol. 27, No. 7 (July 1976), p. 15.
27. David J. Sykes, "Protecting Data by Encryption," Datamation, Vol. 22, No. 8 (August 1976), p. 81.
28. Frank R. Heinrick and David J. Kaufman, "Centralized Approach to Computer Network Security," AFIPS Conference Proceedings 1976 National Computer Conference, AFIPS Press, 1976, Vo. 45, p. 89.
29. Seibholz, Op. cit., p. 51.
30. Sykes, Op. cit., pp. 84-85.
31. Milligan, Op. cit., p. 15.

32. James Martin, Security, Accuracy, and Privacy in Computer Systems, Prentice Hall Inc., 1973, pp. 12-15.
33. Ibid., p. 16.
34. Paul Franson, "Scrambled Data Baffles Thieves," Electronics, Vol. 45, No. 2 (January 17, 1972), pp. 87-88.
35. David Williams, "FC&I Chip Set to Use IBM Encryption Algorithm," Electronics News, Vol. 22, No. 1121 (February 28, 1977), p. 63.

CHAPTER III

DIGITAL COMMUNICATIONS CHARACTERISTICS

This chapter covers the characteristics of digital communications that have an effect on the application of a cryptographic system to digital communications. The characteristics provide the foundation for the subsequent discussions of cryptographic systems. The chapter covers data transfer, channel operation, character transfer, and channel control procedures.

Data Transfer

The units of information transfer assume various forms depending on the type and characteristics of the communications system. In spoken communication, a syllable or a word could be considered the smallest unit of information transferred. Digital communication systems use the bit, short for binary digit, as the smallest unit. A bit represents one of two possible states; one or zero, mark or space, long or short. The concatenation of several bits representing a digital character provides a means of representing ideas that are larger than that allowed by two states. Another

term used for a digital character is byte, typically six to nine bits.

The number of bits in a character and the meaning of the bit combinations varies with the code set used. Morse Code, one of the early binary code sets, was invented by Samuel Morse in about 1837 for use with the telegraph system. The code uses varying numbers of bits to represent the characters of the alphabet, numbers, and special control codes. The number of bits per character varies from one bit, a short, for the letter "E" up to five bits, five longs, for the number zero. Numerous other code systems have been used to meet the specific requirements of a given system; Baudot for teletypewriters, Hollerith for data processing cards, fielddata for military communications. Each of these systems represented the character with a different number of bits and different combinations.

The American National Standards Institute (ANSI) developed and adopted a standard code set, the American Standard Code for Information Interchange (ASCII). The ASCII code set is shown in Table 3.1.¹ The code set provides binary representations of various characters of the alphabet, numbers, special symbols, punctuation, and communication control characters. Each character is represented by seven data bits and an optional eighth parity bit for error detection. A complete description

TABLE 3.1
ASCII Code Set

Low Order Bits				High Order Bits 6,5,4	000	001	010	011	100	101	110	111
3	2	1	0	Hex 1	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	.	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	\$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(8	H	X	h	x
1	0	0	1	9	HT	EM)	9	I	Y	i	y
1	0	1	0	A	LF	SUB	*	:	J	Z	j	z
1	0	1	1	B	VT	ESC	+	;	K	[k	{
1	1	0	0	C	FF	FS	.	<	L	\	l	
1	1	0	1	D	CR	GS	-	=	M]	m	}
1	1	1	0	E	SO	RS	.	>	N	^	n	~
1	1	1	1	F	SI	US	/	?	O	_	o	DEL

The following table gives expansions of the multiple letter mnemonic functions in the chart.

NUL	Null	VT	Vertical Tab	SYN	SYNchronous idle
SOH	Start Of Heading	FF	Form Feed	ETB	End Transmission Block
STX	Start of Text	CR	Carriage Return	CAN	CANcel
ETX	End of Text	SO	Shift Out	EM	End of Medium
EOT	End Of Transmission	SI	Shift In	SUB	SUBstitute
ENQ	ENQuiry	DLE	Data Link Escape	ESC	ESCape
ACK	ACKnowledge	DC1	Peripheral control	FS	file Separator
BEL	BELL	DC2		GS	Group Separator
BS	Back Space	DC3		RS	Record Separator
HT	Horizontal Tab	DC4		US	Unit Separator
LF	Line Feed	NAK	Negative Acknowledge	DEL	DELete (rubout)
		SP	Space		



of the ASCII code set can be found in ANSI X3.4-1968 Code for Information Interchange.

Character Transfer

Information, in the form of characters, may be transferred across communications links by one of three methods. The first method transfers the information character by character. After each character is created, the transfer is accomplished. The time delay between each character occurs independent of any other character or time constraint. A typical example would be a keyboard used to enter data into a remote computer. The operator keys in the data at a rate dependent upon his typing skill. As soon as each character is entered on the keyboard, the associated electronics converts the keystroke into the digital code, ASCII, and transmits the code across the communications link.

Stream Transfer

Using the second method, data is constantly being sent, e.g., stream transmission. This method requires the data to be generated at a relatively constant rate. The time delay between characters is about the same. A typical system could be a remote weather monitoring station. The various weather parameters; wind velocity, temperature, and humidity, are sampled at a constant rate. The sampling equipment converts the data into a

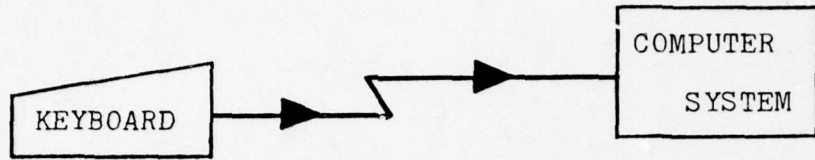
digital code and sends the information with the time delay between data characters being the same and the data continuous.

Block Transfer

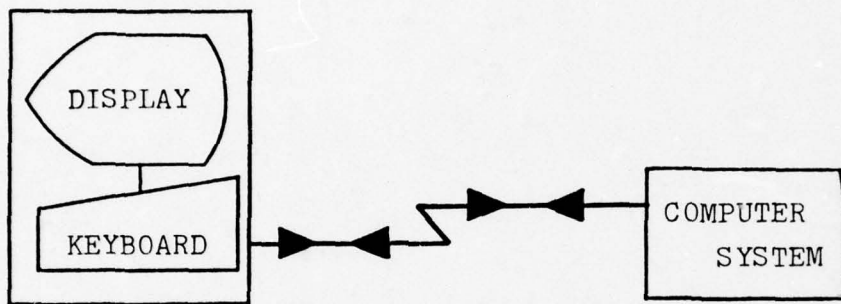
The third method groups data into blocks of several characters before transmission begins. The block length may be fixed in size with each block transmitted after the last character of the block is readied for transmission. Thus, the transmission of a specific character must wait until the block is filled. An alternate method transmits the data block without regard to length after a fixed period of time has elapsed. A third method uses an operator or a system indication to signify the end of a block and a request for transfer. For example, a terminal may store the data until a control character, e.g., Carriage Return, enters the terminal signalling the end of a data block and requesting transfer to the distant end.

Channel Operation

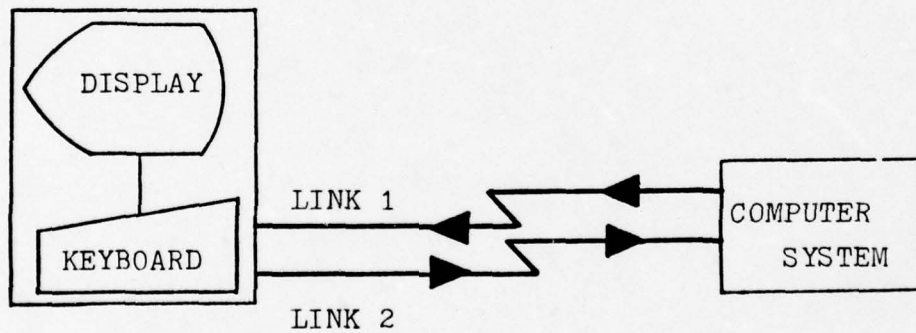
To transfer data from one point to another, a communications channel must be provided. The channel may be realized by a wire, radio, or optical link. The link provides the path for the information transfer. The channel may be operated in one of three ways as shown in Figure 3.1.



SIMPLEX OPERATION



HALF - DUPLEX OPERATION



FULL - DUPLEX OPERATION

FIGURE 3.1
COMMUNICATION CHANNEL OPERATION MODES

Simplex

The simplex mode of operation transmits information in only one direction. Since the data moves in only one direction, the channel cannot provide a method for the receiving end to send control information to the transmitting end. The receiving end must always be ready to accept data at the rate the transmitter sends the data. Error control must be done either by out of channel communications or by forward error correction. Out of channel communications might consist of an operator at the receiving end, after reviewing the received data, calling the operator at the transmitting end. The receiving operator either accepts the data as transmitted or requests the data be retransmitted. Forward error correcting adds redundant information bits to the data transmitted. This redundant information allows errors introduced by the communications channel to be corrected at the receiving end. The addition of the redundant bits reduces the communications channel efficiency; the amount of reduction is dependent on the error characteristics and the error correcting system used. Either the error correcting mechanism must be applied before the encryption of the data with both data and redundant bits being encrypted, or the data must be encrypted and then the error correcting mechanism applied to the encrypted data.

Half-Duplex

The second method of channel operation is half-duplex. Each end of the communications link takes a turn transmitting data or control information to the other end. One end of the channel assumes the master state and the other assumes the slave state. The master sends control information and data to the slave. After the master has finished transmitting, it sends a sequence of control characters to the slave requesting the slave to send data or control information the slave has ready for transmission. The master then waits to receive the data. The master may also send a control sequence to the slave and then assume the slave state. This mode of operation allows each end to send and receive data and to request correction of any data received in error.

Full-Duplex

The full-duplex mode of operation transmits data in both directions simultaneously on either physically or logically separate channels. This mode of operation is sometimes treated as two simplex circuits. Each end of the channel has a transmitter (master) and a receiver (slave). The master sends data and control information to the slave at the receiving end. The logic in the terminal or the computer generates the necessary control

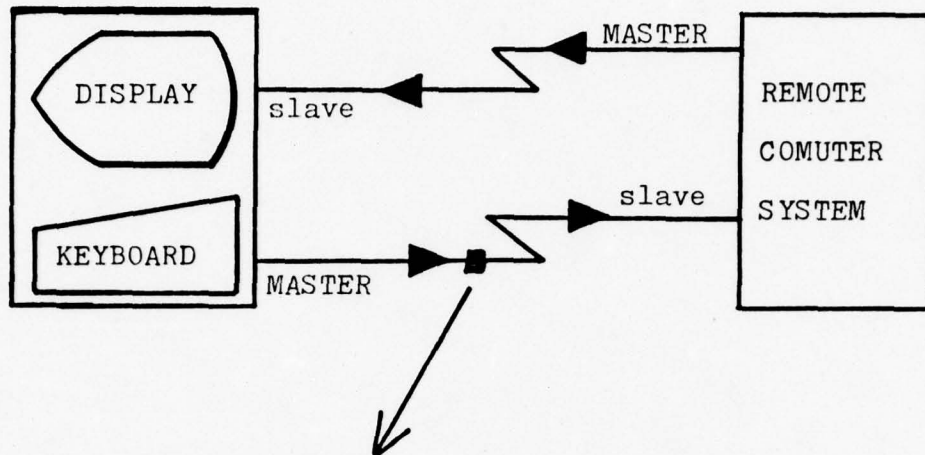
information, e.g., error control, based on information received by the slave. The control information is merged with data and sent by the master to the distant end slave where the control information and data are separated. Figure 3.2 illustrates this relationship showing a full-duplex channel which allows simultaneous data transfer and error control in both directions.

A full-duplex channel may be realized on one physical circuit through the use of appropriate multiplex equipment. The multiplexer divides the available bandwidth into two (or more) parts. Each part of the bandwidth acts as a separate logical channel. A typical allocation of frequencies is shown in Figure 3.3 for a 300 band modem/multiplexer using a standard voice grade telephone line. Using this arrangement, the computer and terminal respond as though they were connected by two separate circuits.

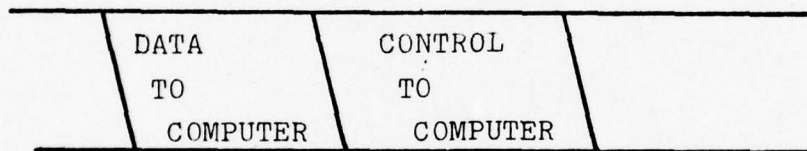
Character Transfer

Information can be transferred between two points, either in parallel or in series. Parallel transfer requires a separate channel for each bit of the character, i.e., ASCII would require at least seven channels. Serial transfer requires one channel to accomplish the transfer. The transmitter separates the characters into bits and sends them sequentially to the receiver. The

FULL - DUPLEX OPERATION



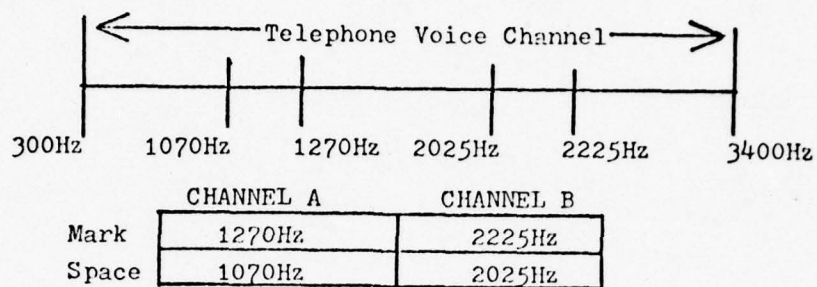
Example of Information Transmitted



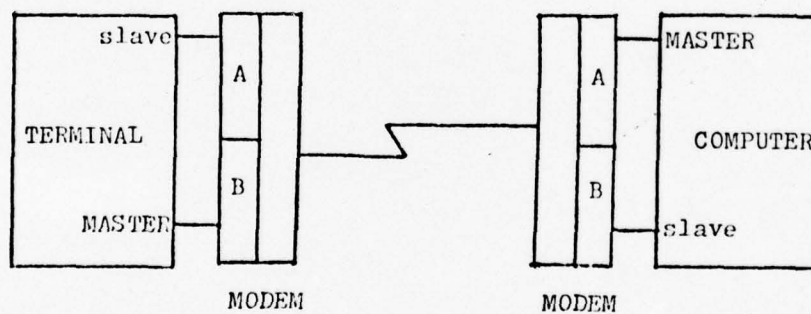
Acknowledgment for Data Block from
Computer "Master" to Terminal "slave"
sent by Terminal "Master."

FIGURE 3.2

FULL - DUPLEX DATA TRANSFER



Channel Division for Frequency Shift Keying (FSK)



Typical Configuration

FIGURE 3.3

300 BAUD 2-WIRE FULL-DUPLEX

receiver reassembles the bits into the characters. The serial transmission of the characters may be accomplished asynchronously or synchronously as discussed below.

Asynchronous Character Transfer

Asynchronous character transfer transmits characters independent of time constraints, The amount of time between character varies from microseconds to several hours. The ability to detect the beginning and end of the serialized character is independent of time and is based on the addition of at least two bits to the character. A single bit added to the beginning of the character indicates the start of a character for transfer and one, one and one half, or two bits added to the end of the serialized character, indicates the stopping of the character transfer.

Figure 3.4 illustrates the structure of a character on an asynchronous channel. Assume the steady state of the channel is a one, illustrated by a high mark, then the start bit would be a zero, low, for one bit time followed by the seven bit of data. The transmissions of at least a single one bit following the last data bit represents the stop bit. Although the timing between characters is asynchronous, the timing between bits of the character is critical. Typical tolerances of one percent to five percent are required between bits.

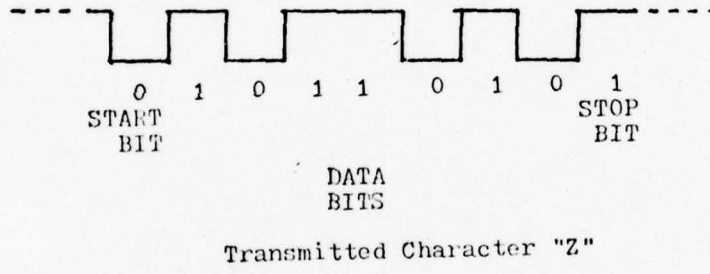
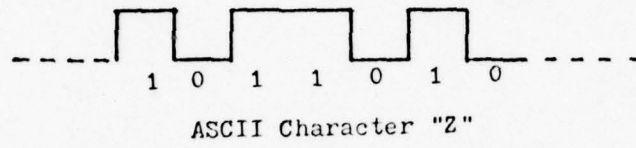


FIGURE 3.4
ASYNCHRONOUS CHARACTER TRANSMISSION

In addition to providing the indication of the start of a character, the start bit allows time for mechanical devices such as a teletypewriter to engage to receive the character. The stop bit allows time for the mechanisms to operate on the data, e.g., print, and to settle down to receive the next character.

The use of asynchronous transmission reduces the potential efficiency of a communications channel. The addition of the start and stop bits creates overhead that does not carry information; rather, only control information. For slow speed devices, this overhead is of small concern. It should be noted that there is at least a 20 percent overhead of ASCII code and as much as 27 percent if two stop bits are used. The use of synchronous transmission reduces this overhead and increases the efficiency of the channel.

Synchronous Character Transfer

Synchronous character transfer requires a relatively constant timing between characters and between bits within a character. The use of start and stop bits is not required to indicate the beginning and end of a character. Rather, the dependence on timely receipt of the character provides a delimiter for each character. Each character does not have the overhead of start and stop bits thereby increasing the potential efficiency of the communications channel. The use of synchronous

transmission still requires the ability to distinguish one character from another. To accomplish this task, the equipment sends a special sequence of bits called a synchronization (SYNC) character.

In ASCII, the SYNC character has the structure as shown in Figure 3.5. The originator sends one or more SYNC characters to the receiver. The receiver hunts for the SYNC character by examining each group of eight bits for a match. If a match is not found the next received bit is concatenated to the character and the oldest bit dropped. The receiver then attempts another match. Once a match is found, the receiver assumes each subsequent set of eight bits to be a character. Data characters will be received correctly as long as the timing between characters and bits remain constant within a few percent.

Typically synchronous transmission is used with systems requiring higher data rates. Systems use data rates from 1200 to several million bits per second. The reduced character overhead, no start and stop bits, increases the efficiency depending on the number and frequency of SYNC characters used to establish and maintain channel synchronization.

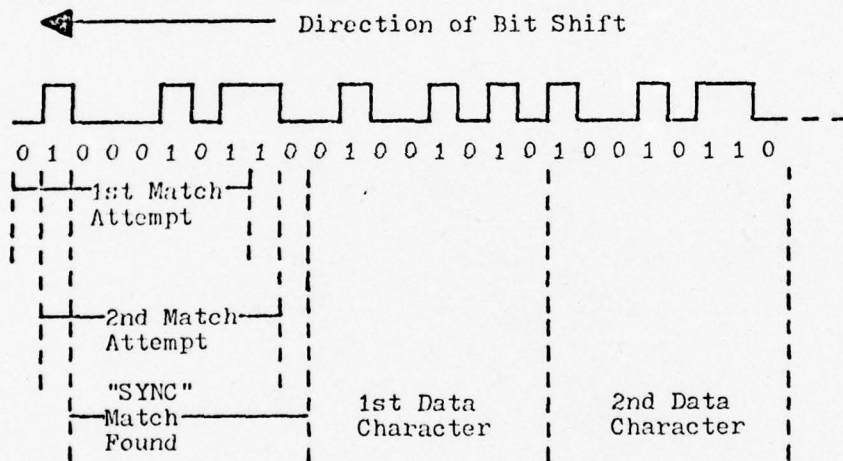
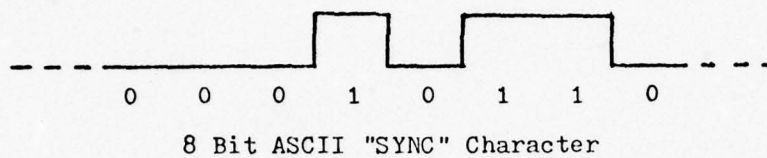


FIGURE 3.5

SYNCHRONOUS CHARACTER TRANSMISSION

Channel Control Procedures

Protocols

In order to provide the orderly transfer of information between the ends of a communication link, a set of rules governing the operation of the links must be established.² These rules, or simply protocols, must as a minimum provide for transmission initiation, transmission control, identification of sender and receiver, error checking and recovery, and transmission termination.³ These protocols may be very simple for a transmitter directly connected to the receiver as on a dedicated link, e.g., a local terminal connected to a computer. The use of a dial-up link requires the use of a more extensive protocol. The most complete protocols involve communications links connected through networks and switching centers. In each case the required elements of communication protocols are provided either implicitly as in a dedicated link, or explicitly as required for dial-up and network operations.

The number and variety of protocols ranges widely depending on the characteristics of the communication links. Typically, each digital communications system develops an optimized set of protocol for the communications links used. Some protocols have become more widely acknowledged as standards through acceptance by

formal standards organizations such as the American National Standards Institute (ANSI). Acceptance by a wide number of users creates defacto standards such as IBM Binary Synchronous Communications.

The application of cryptographic systems to some of the typical standard protocols will be discussed in chapters five and six. Any application of a cryptographic system must not block or hinder the communication protocols for the link. The further discussion of error and retransmission control procedures in this chapter separates the control procedures into four classes; uncontrolled, echoplexed, sequential acknowledgment, and parallel acknowledgment.

Uncontrolled Links

On an uncontrolled link, the transmitter sends data to the distant end, which must always be ready to receive. The receiving end cannot stop the transmission of data nor can it request a retransmission to remove errors. Typically, uncontrolled communications links are used on links having a low probability of introducing errors, or transmitting data relatively insensitive to errors. Out of channel, i.e., by telephone conversation between human operators, procedures provide a method of requesting acknowledgment for received data, or retransmission of data missed or subjected to errors.

Uncontrolled links are being replaced as the introduction of lower cost electronic control equipment becomes available. A simplex channel is typically used.

Echoplexing

An echoplexed control procedure involves the echoing, by the distant end of each character entered at the transmitter.⁴ The echoed character provides the sender an immediate check on the potential correctness of the character received by the distant end. (It should be noted that the error could have occurred on the return path. While the originator would suspect an error, the distant end would have received the character correctly.) If the echoed character is not the same as the transmitted character, the originator could backspace and send the corrected character. Thus, a form of error control would be established. Typically this mode of operation is used with a time-sharing system or terminal connected to the computer system without an intervening digital communications network. Two simplex channels or a half duplexed channel provides the links for echoplex channel operations.

Sequential Acknowledgment

A communications channel using sequential acknowledgment provides for error control and retransmission within the channel by using special sequences of

characters or bits. The originator sends a group of data to the destination and then waits for a reply. The reply acknowledges (ACK) receipt of the data without detected errors and acceptance of the data. If the data had errors introduced by the communication channel detected by the destination, a negative acknowledgment (NAK) is sent requesting a retransmission of the previously sent data. If the destination was unable to accept the data, a NAK is sent. In either case, the originator must retransmit the data group if a NAK is received. To solicit information from the distant end the originator sends a sequence of control character and then becomes the destination for data from the distant end. If a reply is not received from the distant within a previously defined period of time, the end expecting the reply times out and notifies the operator of a potential failure of the communication process (link, equipment, or control process.) In summary, a sequential controlled link has only one data group outstanding and only one type of ACK or NAK expected for a half-duplex circuit.

A full-duplex circuit can be operated in the same manner or can be viewed as two communication paths. In this manner the full-duplex channel logically appears to be both a destination and an originator. ACK and NAK messages from the destination merge with data from the originator of the same end and are sent to the far end.

The receiving end separates the data from the ACK/NAK messages giving the data to the logical originator. In any case the logical originator at each end will have only one data group waiting for a ACK or NAK from the other end's logical destination. Figure 3.6 illustrates this information transfer process and shows how data on ACK/NAK messages are merged and later separated.

Parallel Acknowledgment

While sequential acknowledgment allows only one outstanding data block, parallel acknowledgment allows up to a predetermined number. Typically, up to eight unaccounted blocks of data may be sent from the originator to the destination before the originator stops transmitting data to wait for an acknowledgment. The destination may acknowledge for a message as soon as it is received or may wait until some further action on the data is completed, i.e., writing the data to a tape or disk journal. The communications link may be of such length that the path delay time required to send a message and receive an acknowledgment makes parallel operation more efficient.

An example of this delay on a communications link is shown in Figure 3.7. The communications link trans-
verses a path from the ground through a satellite in synchronous orbit (22,300 miles above the earth) to the

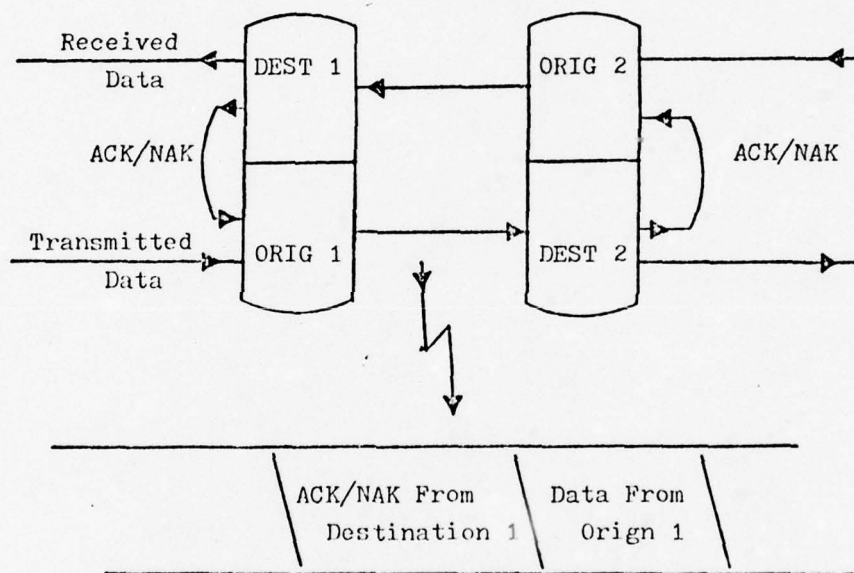
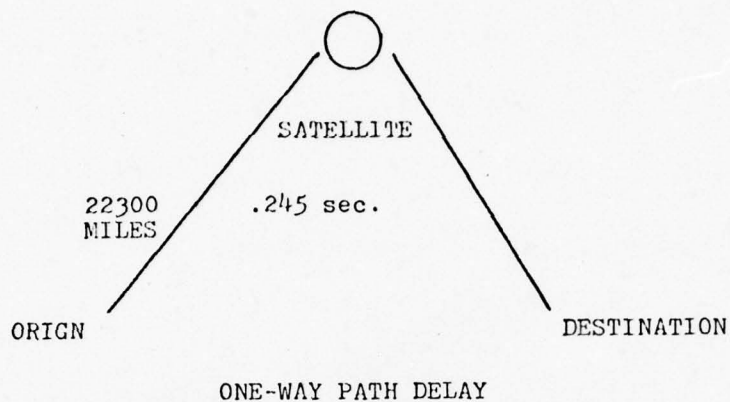


FIGURE 3.6
FULL - DUPLEX SEQUENTIAL
CONTROL



	TIME (seconds)									
ACTION	0.0	.25	.50	.75	1.0	1.25	1.50	1.75	2.0	2.25
ORIGN TX	BLK1	BLK2	BLK3	BLK4						
DEST REC			BLK1	BLK2	BLK3	BLK4				
DEST TX ACK				ACK1	ACK2	ACK3	ACK4			
ORIGN REC ACK						ACK1	ACK2	ACK3	ACK4	

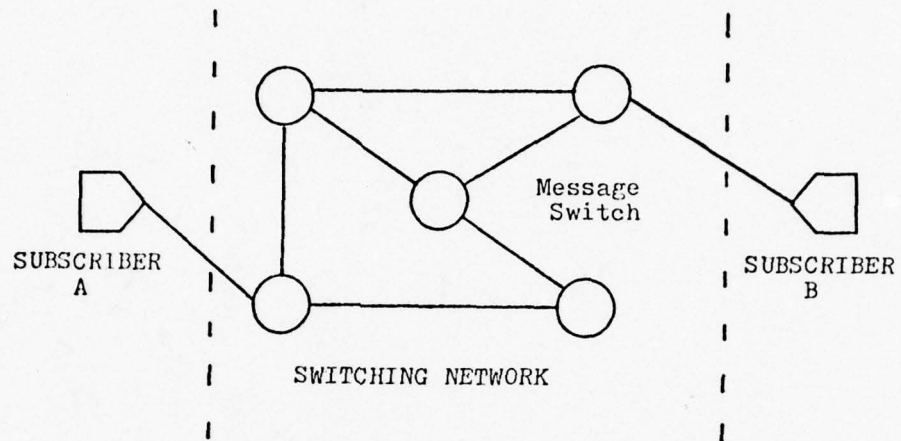
↳ Earliest Point For ACK to be Received

NOTE:
 Transmission at 9600BPS
 Data Block Size 290 Characters
 Data and ACK Piggy-Back in Same Data Block

FIGURE 3.7
 SATELLITE CHANNEL OPERATION

receiving ground station. The total path length is about 45,600 miles resulting in a one way path delay of .49 seconds ($2 \times 45000 / 186000$). As seen in Figure 3.7, a channel operating at 9600 bits per second can send two typical blocks of data during the half of a second one-way delay. Since the return path from the destination is the same length, the originator could send an additional two data blocks before the acknowledgment for the first blocks could be possibly received. Therefore the communications control procedure used in this example must use parallel acknowledgments allowing at least six outstanding data blocks to achieve full utilization of the communications channel (assuming the acknowledgments are merged with data block).

The previous discussions in this chapter have viewed the communications channel as being comprised of a single link between the originator and destination. Typical digital communications systems may be comprised of many actual links interconnected by switching centers. Each source and destination, called subscribers, are connected to the network by a communications channel. Data enters and exits the network across these links. Figure 3.8 illustrates a simplified network connecting two subscribers. In addition to the data, the originator includes control information with the data block. The data and the added information are called messages.



Typical Digital Communications Network

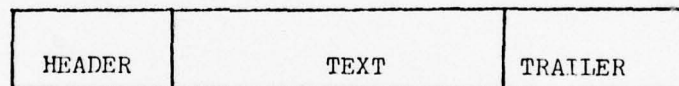


FIGURE 3.8
TYPICAL MESSAGE FORMAT

Each message represents a transaction to the digital communications network. Figure 3.8 shows the basic format of a message; header, text, and trailer. The header must at least indicate the subscriber to receive the message and the originating subscriber. Table 3.2 outlines additional parameters that may be included in the message header and trailer. All the information shown is not required in all networks. The information in the header is used by the communication network to accurately deliver the message.

TABLE 3.2
MESSAGE HEADER AND TRAILER PARAMETERS

HEADER PARAMETERS
TO Address(es)
FROM address
Speed of Service Requested
Security Level Requested
Length of Message
Message Sequence Number
Data and Time
Type of Message
TRAILER PARAMETERS
Error control
Next Message Sequence Number
End of Message Indicator

CHAPTER III

Footnotes

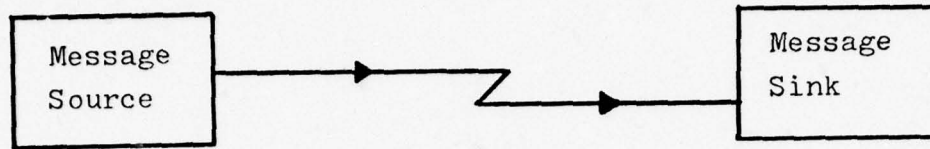
1. Bob Brehm, "Using a Keyboard ROM," Byte, Vol. 2, No. 5 (May 1977), p. 82.
2. Donald W. Davies and Derek L.A. Barber, Communications Networks for Computers, John Wiley & Son, 1973, p.382.
3. Albercht J. Neumann, Brian G. Lucas, Justin C. Walker, and Dennis W. Fife, A Technical Guide to Computer-Communications Interface Standards, National Bureau of Standards, 1974, p. 66.
4. Davies, Op. cit., p. 412.

CHAPTER IV

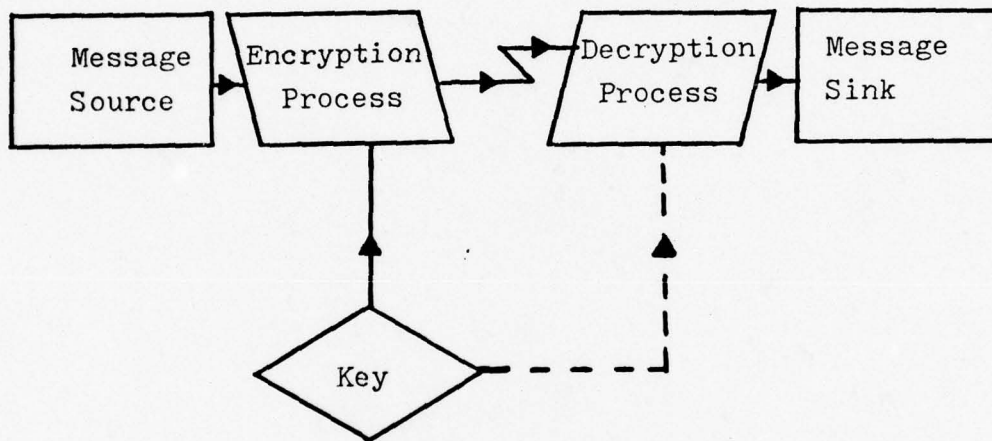
ENCRYPTION OF DATA COMMUNICATIONS

Cryptographic Systems

The intended purpose of a cryptographic system is to hide information from disclosure to third parties. All methods of encryption attempt to replace known, meaningful information with unknown, meaningless information.¹ A cryptographic system comprises a set of operations, called the encryption algorithm, to convert the information into a protected state, encrypted form. The inverse of this algorithm, the decryption process, provides a one to one relationship between the encrypted text and the original text. Figure 4.1 provides a block diagram of the cryptographic system.² The four key elements of the system are the text, the encryption process, decryption process and the key. In order for the decryption process to recover the original text, the same key used in the encryption process must be used in the decryption process. The text to be encrypted is defined as plaintext, and the encrypted text is defined as ciphertext.



Unencrypted Communications Link



Encrypted Communications Link

FIGURE 4.1
CRYPTOGRAPHIC SYSTEM BLOCK DIAGRAM

Attributes of a Secure Cryptographic System

In 1883, Auguste Kerckhoffs, a naturalized Frenchman, established two basic principles for a cryptographic system. First, the key used to encrypt the information must "withstand the operational strain of heavy use."³ The intruders are assumed to have knowledge of the general cryptographic system. Thus, the security of the information rests with the key.⁴ The second principle is that only a cryptanalyst knows the true security of the cryptographic system.⁵ A system is secure only until the secrecy is removed by the cryptanalyst.

A cryptographic system can be evaluated as to its usefulness based on five tests outlined by Shannon.⁶ First, the degree of secrecy provided by the system is a function of the amount of work required to break the system and reveal the hidden information. Next, the size of the key should be as small as possible. The key must be conveyed to the various locations to be used in the cryptographic system. Third, the complexity of the encrypting and decrypting process must be as simple as possible to avoid errors and lost time. The fourth requirement states that errors introduced into the encryption or decryption process or in the encrypted information should have as low an error extension as possible. Ideally an error in one character should affect only that character. Fifth, the encrypted

information should be as nearly the same size, length, as the unencrypted information. If the unencrypted information is 20 characters, the encrypted text should also be 20 characters.

General Classes of Cryptographic Systems

In general there are two methods of securing data, codes and ciphers. A code conveys meanings planned and published in advance. These codes must be available to both sender and receiver. Typically a code replaces single words and commonly used phrases with prearranged, meaningless sequences of characters usually of fixed length.⁷ The person (or system) using the code finds the word or phrase to be encoded in the code book and locates the corresponding code sequence. To decode, the code sequence is located in the code book and the resulting meaningful text found. The use of codes is typically limited to predefined words and phrases.

The second method, cipher, replaces the characters of words (or numerals of numbers) with another character or symbol. Since the cipher is alphabetic anything a language can describe can be represented in the cipher and any new idea representable in the alphabet can be used with the cipher.⁸ This thesis deals with the use of ciphers rather than codes as the use of a cipher is more adaptable to automated digital communications.

General Types of Ciphers

A cipher may be achieved in one of three ways; substitution, transposition, or addition; or by a combination. Substitution replaces the information character with a different character on a one-for-one basis.⁹ Figure 4.2 illustrates a typical substitution cipher. In this example the letter D replaces each A appearing in the original text. The example shown is based on the Caesar cipher to be explained below.

The second method, transposition, rearranges the characters without changing them.¹⁰ For example, the word "encryption" might be transposed to "iycernpotn." This form of cipher is distinguished from the other two in that more than one character is operated on at one time, i.e., a group or block of characters comprising a word or phrase. Transposition need not be limited to single words; whole paragraphs could be transposed provided the encryption algorithm could span that number of characters.

The third method, addition, uses appropriate algebraic transformation to operate on the data and key to produce the encrypted text.¹¹ Figure 4.3 shows two types of addition that may be performed on a text stream. In the first example; B added, modulo-26, to U; yields the encrypted character, V. (The addition assumes A = 0 through Z = 25, and any sum above 25 is

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

Character Substitution Table

Message..... BUY THREE HUNDRED SHARES OF PIKE MINING
Encrypted..... EXB WKUHH KXQGUNG VKDUHV RI SLNH PLQLQJ

Example of Encryption Process

FIGURE 4.2
CHARACTER SUBSTITUTION ENCRYPTION
(CAESAR CIPHER)

treated modulo 26, e.g., 29, modulo-26=3). In the second method the letters are treated as five bit (binary) numbers and modulo-2 addition is performed. The resulting sum is converted back to a character. Other forms of algebra could be performed on the data by the encryption algorithm provided the algebra is reversible. The encrypted text must be decrypted to recover the exact same data as before encryption.

General Methods of Cryptanalyst

A Cryptanalyst, the intruder, is a person who uses any method other than having the key, to decrypt information previously encrypted in a system he knows or has guessed. Although the assumption that the cryptanalyst knows the cryptographic system is pessimistic, the user of the system must accept that the intruder will eventually determine the system used; the secrecy must lie in the key.¹² The most common method the cryptanalyst employs to attack a cryptographic system uses statistical analysis of intercepted, encrypted information based on knowledge of the language characteristics.¹³ Such statistical analysis uses letter or word frequencies plus the inherent redundancy in natural languages to break the cipher.¹⁴

To prevent the cryptanalyst from succeeding, the cryptographic system must diffuse and confuse the original text. These two methods will frustrate the

statistical analysts of intercepted, encrypted text.¹⁵ The principle of diffusion spreads the known characteristics of the language such as to lose the identifiable traits among the whole of the text. This greatly increases the amount of encrypted text required to recover the needed statistics to decrypt the text. The second principle, confusion, hides the known statistics in such a way as to greatly increase the problem of recovering them by making the relationship between the encrypted text and the key very involved.

Types of Cryptographic Systems

This section covers various types of cryptographic systems used in past communications. These systems have been grouped into three classes; character substitution, stream transformation, and block encryption. The systems presented are intended to be representative of the various methods used to secure data with the intention of pointing to the weaknesses of these methods.

Character Substitution

Encryption algorithms that operate on each character of text as a unit are separated into two general classes, monoalphabetic and polyalphabetic. A monoalphabetic substitution uses one key alphabet in the encryption process and the polyalphabetic substitution uses two or more key alphabets.

Monoalphabetic encryption algorithms are among the oldest to be used. In general they can be divided into three groups; simple substitution (Caesar), Matrix (Playfair), and Auto-key (self-keying). Figure 4.1 is an example of the Caesar type. This type of encryption derives its name from Julius Caesar who used a simple form in his communications to Cicero.¹⁶ The playfair encryption algorithm is illustrated in Figure 4.4. This system was devised by the ancient Greek writer, Polybius.¹⁷ The playfair system has the advantage of converting the alphabet of 26 characters into a system of a smaller number of symbols.¹⁷ In the example shown in Figure 4.4, the letter A is converted to number 52. These numbers can easily be transmitted in a communications system. The third type is called auto-key invented by Blaise de Vigenère in 1586. The autokey uses a key word and then the message itself to provide the long key stream used to encrypt the message.¹⁸ In the example shown in Figure 4.5, the key word QKXJS is used beginning the encryption process. The first letter of the message, B, is add modulo-26 to the first letter of the key word, Q, resulting in the letter R. This process continues until all the letters of the key word are used. Then, the message becomes the key, i.e., B becomes the next key letter. There are many variations and combinations on these three methods.

Message..... B O U G H T T H R E E
 Encryption Key..... U X T K E L H D A Y O
 Encrypted Message... V L N O L E A K R C U

Note: B added, modulo-26, to U equals V, etc.
 Where $a=0, b=1, c=2, \dots, z=25$.

Modulo-26 Addition

Message Characters.....	S	E	L	L
Binary Substitution.....	10010	00010	01011	01011
Key Characters.....	U	X	A	K
Binary Substitution.....	10100	10111	00000	01010
Encrypted Binary.....	00110	10101	01011	00001
Encrypted Characters.....	G	V	L	B

Note: Modulo-2 Addition: $0+0=0$
 $1+0=1$
 $0+1=1$
 $1+1=0$

Where $A=00000, B=00001, C=00010, \dots, Z=11001$, and
 $2=11010, \dots, 7=11111$.

Modulo-2 Addition

FIGURE 4.3
 ADDITIVE ENCRYPTION

		Second Number				
		1	2	3	4	5
First Number	1	G	E	V	T	O
	2	H	R	L	X	Z
	3	S	I	J	Y	U
	4	B	N	K	F	Q
	5	W	A	C	P	M

Note: I and J Both Equal 32

Sample Matrix

Message..... S E L L T E N

Encrypted Message.. 31 12 23 23 14 12 42

Sample Message

FIGURE 4.4

PLAYFAIR ENCRYPTION

Initial Key....QKXJS

Message.....Bought Ten Shares

Message.....	B	O	U	G	H	T	T	E	N	S	H	A	R	E	S
Key Stream.....	Q	K	X	J	S	B	O	U	G	H	T	E	N	S	
	KEY						MESSAGE								
Encrypted message....	R	Y	R	P	Z	V	H	Y	T	Z	A	T	V	R	K

FIGURE 4.5
AUTOKEY ENCRYPTION

The major weakness of these monoalphabetic encryption algorithms lies in their inability to diffuse or confuse the characteristics of the language used. The cryptanalyst uses the letter frequency of the encrypted text to determine probable clear text characters. For example, in English the letter "E" is most frequent. Another characteristic is letter pairing. In English the pairs TH, QU, ST, and RE occur with a relatively high frequency; while the pairs AO and OI occur relatively infrequent. Word characteristics, such as the frequent use of THE and single letter words A and I, provide the third method used by the cryptanalyst.¹⁹

Polyalphabetic substitution uses more than one key alphabet to encrypt the text. Figure 4.6 shows a typical system using 26 alphabets. In the example shown the alphabets are listed in rotated alphabetic order; a stronger system would use 26 random order alphabets. The interception of the Nth letter of the key word or phrase (on the vertical axis) and the Nth letter of the clear text (on the horizontal axis) provides the encrypted character.²⁰ As pointed out by Shannon, if the table is random and a random key phrase at least as long as the message is used, then in theory the encrypted text is perfect and cannot be broken.²¹ When used by hand this system was subject to error and was not extensively used after its invention in the 1800's.²²

Message Text

ABCDEFGHIJKLMN OPQRSTUVWXYZ

A	abcdefghijklmnopqrstuvwxy
B	bcdefghijklmnopqrstuvwxyza
C	cdefghijklmnopqrstuvwxyza
D	defghijklmnopqrstuvwxyza
E	efghijklmnopqrstuvwxyza
F	fghijklmnopqrstuvwxyza
G	ghijklmnopqrstuvwxyza
H	hijklmnopqrstuvwxyza
I	ijklmnopqrstuvwxyza
J	jklmnopqrstuvwxyza
K	klmnopqrstuvwxyza
L	lmnopqrstuvwxyza
M	mnopqrstuvwxyza
N	nopqrstuvwxyza
O	opqrstuvwxyza
P	pqrstuvwxyza
Q	qrstuvwxyza
R	rstuvwxyza
S	stuvwxyza
T	tuvwxyza
U	vwxyza
V	wxyza
W	wxyza
X	xyza
Y	xyza
Z	xyza

Key
Stream

Sample Table

Message.....	S E L L	A L L	S H A R E S
Key.....	O X U Y	E A Z	L V H R K F
Encrypted Message..	G B F J	E L K	Z C H I O X

Sample Message

FIGURE 4.6

POLYALPHABETIC ENCRYPTION

Stream Transformations

Stream Transformations are intended to operate (encrypt) a continuous stream of information as found on a digital communications link. One of the earliest attempts to develop equipment that would encrypt digital communications automatically was done by Gilbert S. Vernam of American Telephone and Telegraph Company in 1918.²³ Vernam proposed a form of teletypewriter equipment that would automatically change the input key stream to an encrypted form using a long papertape as an additive key. The major advantage of the equipment allowed the encryption process to be accomplished automatically just prior to transmission and decryption just after reception. These automatic operations increased the speed of encryption/decryption and reduced the chance of human error. In operation, the key tape (a long pre-punched tape of random Baudot characters) is added modulo-2 (exclusive ored) to the text to be transmitted. At the receiving end, the similar key tape is again added modulo-2 recovering the original text. The system uses modulo-2 addition since the Baudot code used with teletypewriters consists of five binary bits for each character.

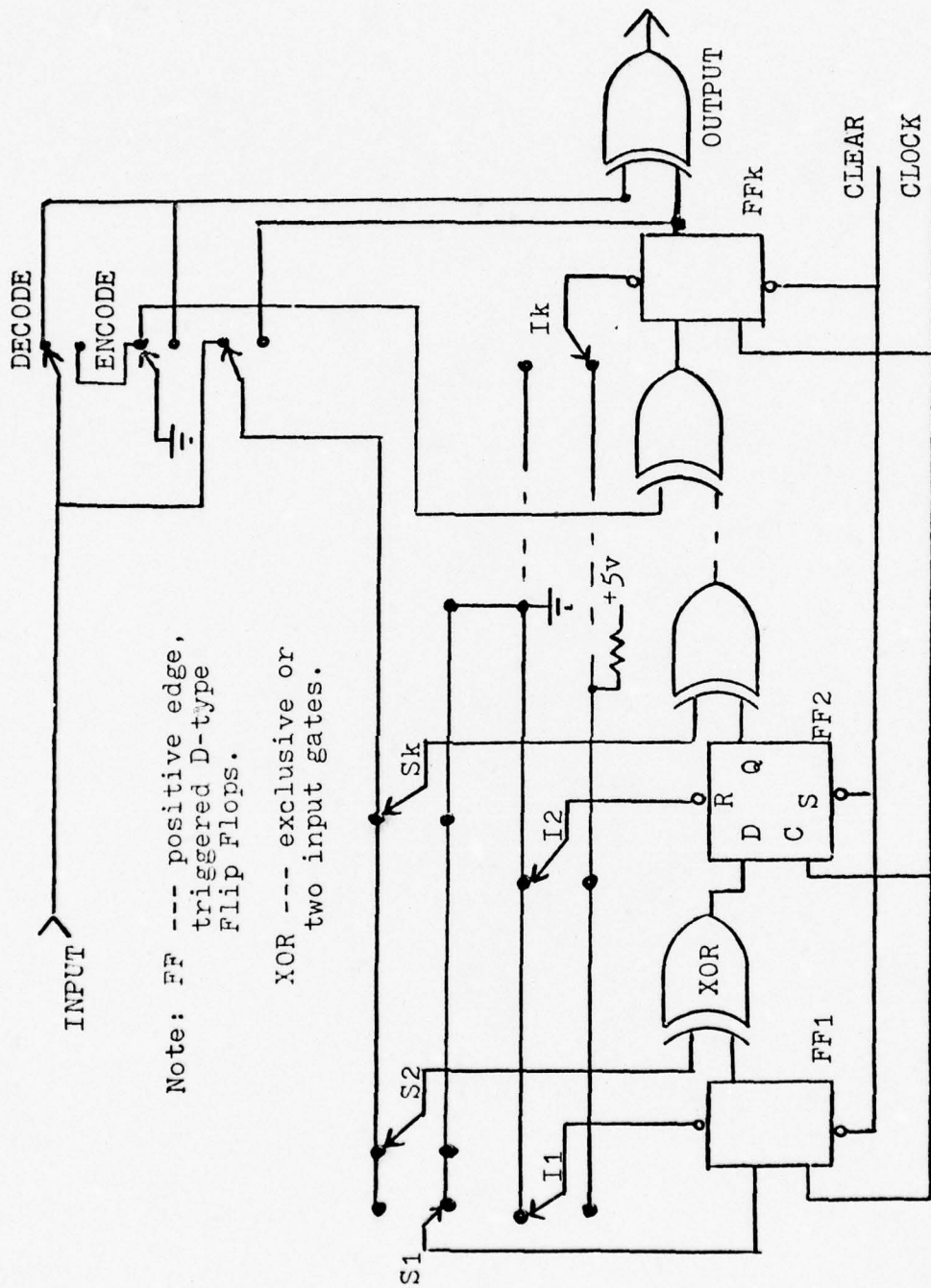
The major weakness of the single tape Vernam system lies in its use of a 32 by 32 polyalphabet for the key tape. If the key repeats as in the originally designed

system, the cryptanalyst can use the previously discussed methods to break the encryption process and recover the data.²⁴ A system using a key tape only once that is composed of random Baudot characters and is at least as long as the message can make the system theoretical secure.²⁵ However, the number of keys required and their length makes the key system administratively unsupportable.

In an attempt to increase the effective key tape length, Lyman F. Morehouse of AT&T suggested using two tapes, one of length 1000 and one of length 999 characters.²⁶ The first tape, U, and the second tape, V, are both added to the data producing the encrypted text. After stepping through the U tape completely, the V tape steps once. In this manner an apparent key length of V times U is created, i.e., 999,000 characters long. Bryant Tuckerman of IBM has recently shown that the effective key length is not V times U but rather $(V+U)$ to $5(V+U)$ depending on the type of text intercepted. Using computer programs developed by Tuckerman, the structures of the tapes can be recovered with as few as $100(U+V)$ characters of encrypted text even if the unencrypted data is unknown.²⁷

The second form of stream transformation involves the use of pseudorandom binary stream implemented with shift registers as used in most currently available

commercial equipment. This form of encryption operates on the input data producing an apparently random stream of characters. However the binary key stream will repeat after $2^n - 1$ bits, where n equals the number of stages in the shift register. If n equals sixteen, then the key stream will begin to repeat after 32768 bits.²⁸ Figure 4.7 shows a typical linear shift register circuit used to encrypt and decrypt data.²⁹ The operations on the data by the shift register are linear since an modulo-2 addition is performed and no multiplication is performed. The two key variables are the settings of the switches, S_1 to S_n providing the feedback paths and the settings of switch I_1 to I_n provides the initial condition setup. The encryption process uses these initial conditions plus the previously sent data to encrypt the next bit, i.e., future encryption depends on past encrypted data. Errors introduced in the encrypted data stream are extended to the subsequent data. This error extension results in more than one bit in error for a single bit error. Due to the effects of error extension, the data to be encrypted should be divided into blocks and each block headed by a group of known data to insure synchronization. Without the destination being synchronized with the originator, a complete loss of data is possible resulting in a barrier to communications rather than securing the communications link.³⁰



Note: FF --- positive edge, triggered D-type Flip Flops.
 XOR --- exclusive or two input gates.

FIGURE 4.7

PSEUDORANDOM ENCODER/DECODER

The need for cryptographic synchronization creates the opening for the cryptanalyst to break the encryption process.³¹ "Only a limited amount of information--on the order of $2n$ (n = the number of shift-register stages) bits of enciphered text and the corresponding message--is enough to break the code."³² The synchronization pattern or standardized parts of the message could provide all the known text required to break the cipher. The solution process involves the use of a set of N linear equations where N is the number of shift register stages. Using well known rules of matrix operation, the N equations are solved resulting in the key used to encrypt the data.³³ These solution algorithms can be easily programmed on a computer giving rapid solutions to the matrix.

The addition of nonlinear stages involving the use of multiplication to the shift-registers can increase the protection by several orders of magnitude when used in conjunction with a mixing transformation.³⁴ A typical transformation is the deVries cipher which will be discussed below.

Block Encryption

Block encryption algorithms differ from character substitution and stream transformation in that the block encryption algorithm operates on more than one character at a time. The basic procedures used require multiple

steps. First, the data is segmented into blocks of a predetermined length. Then using a series of character substitutions or transformations, the plain text is converted into the cipher text. The number and type of operations vary from system to system. The substitution, the transformation, or both may involve the use of a key. After all encryption steps are completed, the communications system transfers the encrypted data to the destination. There the inverse algorithm is applied to recover the plain text. The block size of the plain text and the cipher text are usually the same length.³⁵ Any errors introduced by the communications system to the encrypted block will affect only that block if there is no inter-block dependency.

An example of a simple block encryption system is the deVries cipher. The process consists of a two step substitution using binary groups to represent the characters. Figure 4.8 shows a typical set of groups that exhibit the necessary comma free (not requiring separators to distinguish each character from the adjoining one) properties.³⁶ The binary groups are concatenated together and then separated into groups of five bits. These new groups determine the letter from the lower list to be transmitted. This system has the advantage of diffusing the language characteristics and making the cryptanalyst's job more difficult.

E 000
 T 001 I 1000 F 11000 Y 111000 K 1111001
 A 0100 S 1001 C 11001 P 111001 X 1111010
 O 0101 H 1010 M 11010 W 111010 J 1111011
 N 0110 D 10110 U 110110 B 111011 Q 1111100
 R 0111 L 10111 G 110111 V 1111000 Z 1111101

Comma Free Substitution List

A 00000 B 00001 C 00010 D 00011 E 00100
 F 00101 G 00110 H 00111 I 01000 J 01001
 K 01010 L 01011 M 01100 N 01101 O 01110
 P 01111 Q 10000 R 10001 S 10010 T 10011
 U 10100 V 10101 W 10110 X 10111 Y 11001
 Z 11001 2 11010 3 11011 4 11100 5 11101
 6 11110 7 11111

Five Bit Group Substitution

Message.....	S	E	L	L	O	I	L
Comma Free...	1001	000	10111	10111	0101	1000	10111
Five Bit.....	10010	00101	11101	11010	11000	10111	
Encrypted....	S	F	5	2	Y	X	

Sample Message

FIGURE 4.8

DEVRIES ENCRYPTION

The second example of a block encryption system is the Lucifer device developed by IBM. This system uses repeated substitutions and transformations on a block of sixteen, eight bit characters for a total block length of 128 bits using a key of the same length.³⁷ The Lucifer system algorithm provided the bases for the FIPS PUB 46 Data Encryption Standard to be discussed next.

FIPS PUB 46 Encryption Algorithm

Background

On January 15, 1977 the Secretary of Commerce approved the adoption of a data encryption standard published in the Federal Information Processing Standards (FIPS) Publication (PUB) 46, Data Encryption Standard (hereafter referred as PUB 46 DES.) The standard is based on the Lucifer system developed by IBM. IBM has granted royalty free use of the patent associated with their algorithm. The development of the algorithm was based on work done by Horst Feistel, William A. Notz, and J. Lynn Smith.³⁸ In the original form, the block size was 128 bits with a 128 bit key. As adopted, the algorithm uses a block of 64 bits and 64 bit key consisting of 56 key bits and eight parity bits.³⁹

Description

The algorithm uses a recirculating block product cipher with all bits of the block being encrypted at once. The block size of 64 bits approximates eight ASCII characters (eight, eight bit, bytes.) The algorithm alternately performs linear permutations, reordering of the bits, and nonlinear substitution. These operations repeat sixteen times insuring a thorough mixing of the data.

Figure 4.9 depicts the block diagram of the algorithm operation.⁴⁰ After an initial permutation, the data is subjected to sixteen iterations of the substitution and permutation operators. Each iteration uses the rightmost 32 bits of the block and one of sixteen, 48 bit selections from the key as shown in Figure 4.10.⁴¹ The rightmost 32 bits form the leftmost 32 bits of the next iteration. To form the rightmost 32 bits for the next iteration, the rightmost 32 bits of this iteration are subjected to the operations depicted in Figure 4.11.⁴² The resulting bits are then exclusive-ored with the leftmost 32 bits of this iteration. All sixteen iterations follow this procedure except the last. After the exclusive-or operation the rightmost and leftmost 32 bits are exchanged and then the inverse of the initial permutation is applied.

Decryption is performed using the exact same sequence except that the sixteen sets of 48 bit keys are selected in

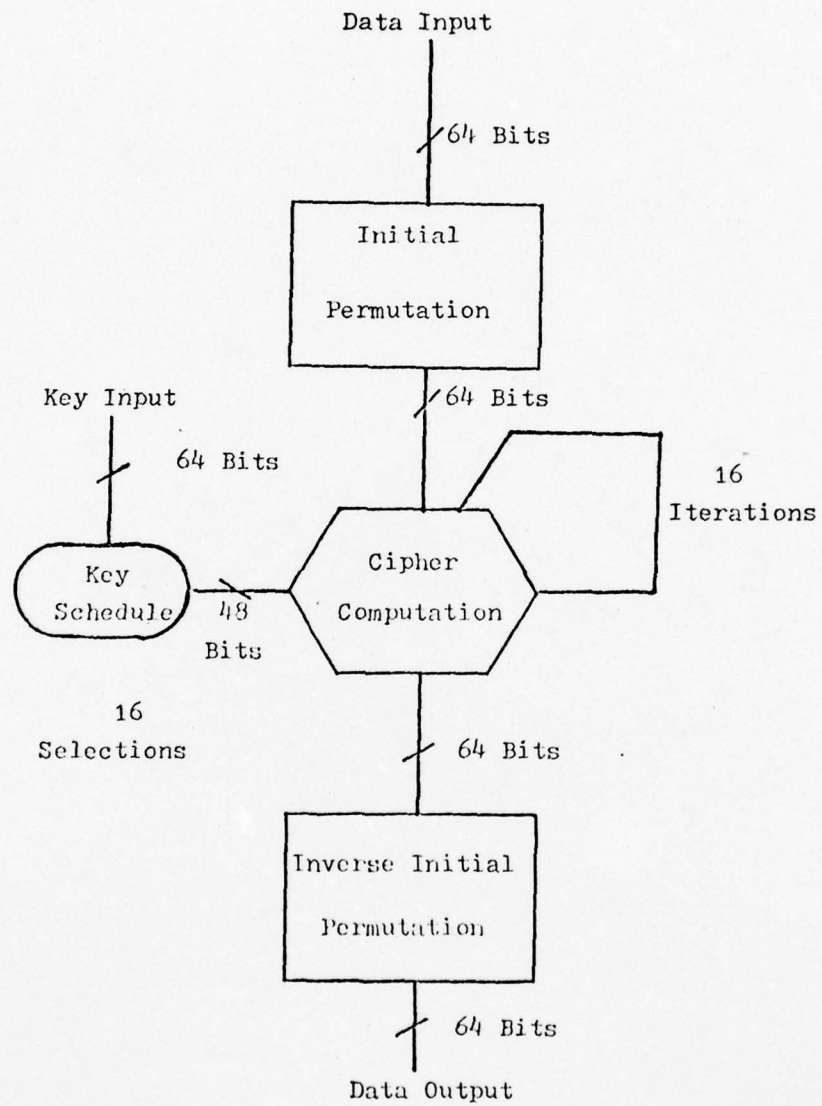


FIGURE 4.9
BLOCK DIAGRAM OF PUB 46 DES

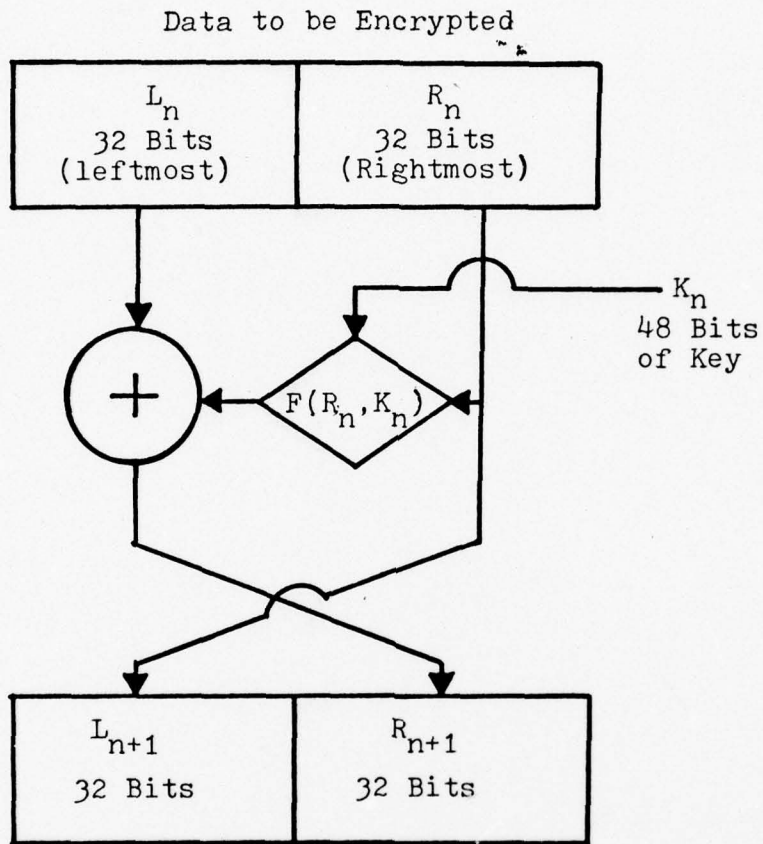


FIGURE 4.10
ENCRYPTION ITERATION

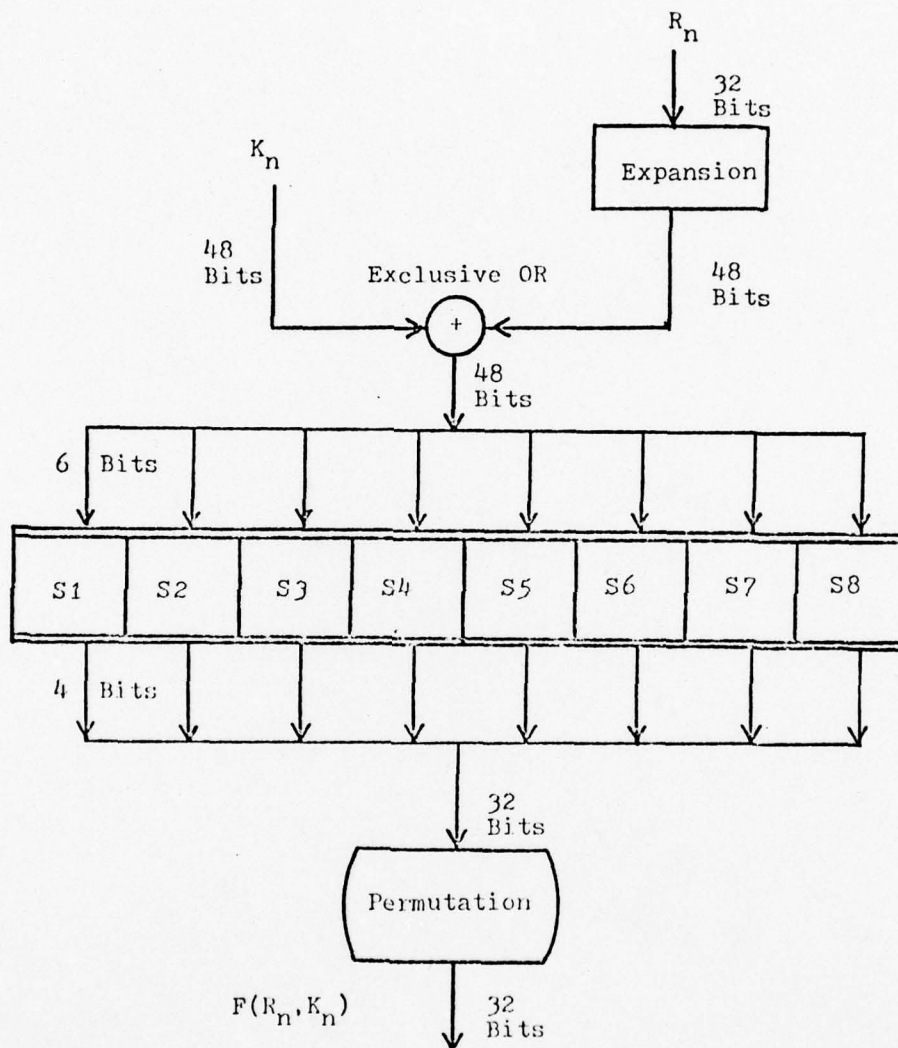


FIGURE 4.11

COMPUTATION OF THE CIPHER FUNCTION

reverse order. This allows the same equipment to be used to encrypt and decrypt a block. A detailed description of the substitution, data permutation, and key permutation is contained in FIPS PUB 46.

Attributes

The basic question asked of the algorithm is how secure is it? The exact amount of security is a function of the amount of work required to break the code. To date, there have not been any reported cases of a properly implemented system being broken. The National Bureau of Standards asked the National Security Agency, NSA, to investigate the security of the algorithm. The Agency reported the algorithm was very effective.⁴³ "In its security function, NSA creates and supervises the cryptography of all U.S. government agencies."⁴⁴ In short, NSA is the expert agency on cryptographic systems in the United States. "With a 56 bit key ($2^{56} = 7.2 \times 10^{16}$) if each trial took one microsecond (which is not really possible today), it would take over one thousand years to find the key" by trial and error methods.⁴⁵ Since the algorithm has been designed to be highly resistant to statistical attack, only governments would have the necessary money and resources to break the cipher.⁴⁶

Due to the high degree of mixing in the algorithm, small changes in the plain text, key, or cipher text produce large changes in the decrypted text or cipher

text. Figure 4.12 illustrates these effects.⁴⁷ As seen in the examples A16 and A18, a single bit change in the key greatly changes the resulting cipher text. Examples A19 and A20 exhibit similar action for a single bit change in the plain text. Due to the high degree of mixing, a single bit error in the cipher text could be expected to have an effect on all eight characters of the 64 bit block.⁴⁸ In summary, the algorithm performs five distinctly separate transformations on the data; keyed transposition, key addition, keyed nonlinear transformation, permutation, and convolution.⁴⁹ These steps insure a complete confusion and thorough diffusion of the data.

Implementation

The algorithm is intended to be implemented using hardware rather than software technology. Possible hardware consists of Large Scale Integration (LSI) or by using Medium Scale Integration (MSI) such as the 7400 series Transistor Transistor Logic (TTL) circuits.⁵⁰ Implementations using Microprocessors with the algorithm and control program in Read Only Memory (ROM) are also within the standard.⁵¹

The algorithm has been implemented in various ways. Most of the early implementations have been in software, primarily for testing the security of the algorithm. The amount of time to encrypt a 64 bit block varies from

EXP	KEY	DATA	CIPHER (ENCRYPTED)	DECRYPTION OF CIPHER
A 9	49E026469EBA7304	0573EC52D6837492	B02B087303484D84	0573EC52D6837492
A10	C1C1010101010101	0000000000000000	8CA64DE9C1B123A7	0000000000000000
A11	7F7F7F7F7F7F7F7F	0000000000000000	5EFA76E8A5A9EB37	0000000000000000
A12	7F7F7F7F7F7F7F7F	1111111111111111	CEDA5902D980D525	1111111111111111
A13	C1C1010101010101	AAAAAAAAAAAAAAAAAA	3AE716954DC04E25	AAAAAAAAAAAAAAAAAA
A14	C1C1010101010101	AAAAAAAAAAAAAAAAAB	17D8E9C374D14494	AAAAAAAAAAAAAAAAAB
A15	C1C1010101010101	5555555555555555	B109FD803EB2D05E	5555555555555555
A18	C1C1010101010102	5555555555555555	451F0C33F24FB8DC	5555555555555555
A19	C1C1010101010104	5555555555555555	CAB6E849E0AB0C32	5555555555555555
A20	C1C1010101010104	5555555555555554	7D34A65A0E2B62CE	5555555555555554

Note: Hexadecimal numbers used.

FIGURE 4.12

EXAMPLES OF DATA ENCRYPTION STANDARD

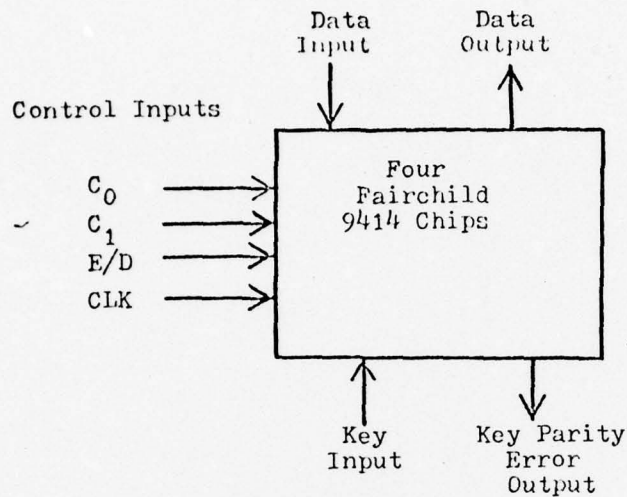
one hundred microseconds for large computers and 50,000 microseconds for minicomputers.⁵² About 160 TTL integrated circuits are required to implement the algorithm, hold the key, and provide interface logic to the communications system.⁵³ The Motoral Corporation plans to offer a single card system using the 6800 microprocessor. The unit would sell for about \$500 in single units. The Fairchild Camera and Instrument Corporation is designing a LSI implementation of the algorithm using four specially designed circuits. Figure 4.13 is a block diagram of the LSI implementation. Initial cost should be about \$200 with the price dropping to about \$20 as demand increases.⁵⁴

Application to Data Stream

In general, two methods can be used in the application of a cryptographic system to digital communications systems, link by link and end to end. Link by link application encrypts the data only on the communications links and not within the data switching centers. End to end application encrypts the data prior to entry into the communications system and decrypts the data only on exit from the communications system.

Link by Link

As shown in Figure 4.14, each link of the communications system requires two encryption devices. A message



Definition of Control Inputs:

<u>C₀</u>	<u>C₁</u>	
0	0	Input Key -- Read in Key, 8 Clock periods.
0	1	Input/Output Data -- Read in data or write out data, 8 Clock periods.
1	0	Encrypt/Decrypt Data -- Process stored data, 16 Clock periods.
1	1	Not Used
<u>E/D</u>		
0		Set Encrypt Mode
1		Set Decrypt Mode
CLK		---- 200 Nanosecond Clock Period

FIGURE 4.13

BLOCK DIAGRAM OF FAIRCHILD 9414 CHIP SET

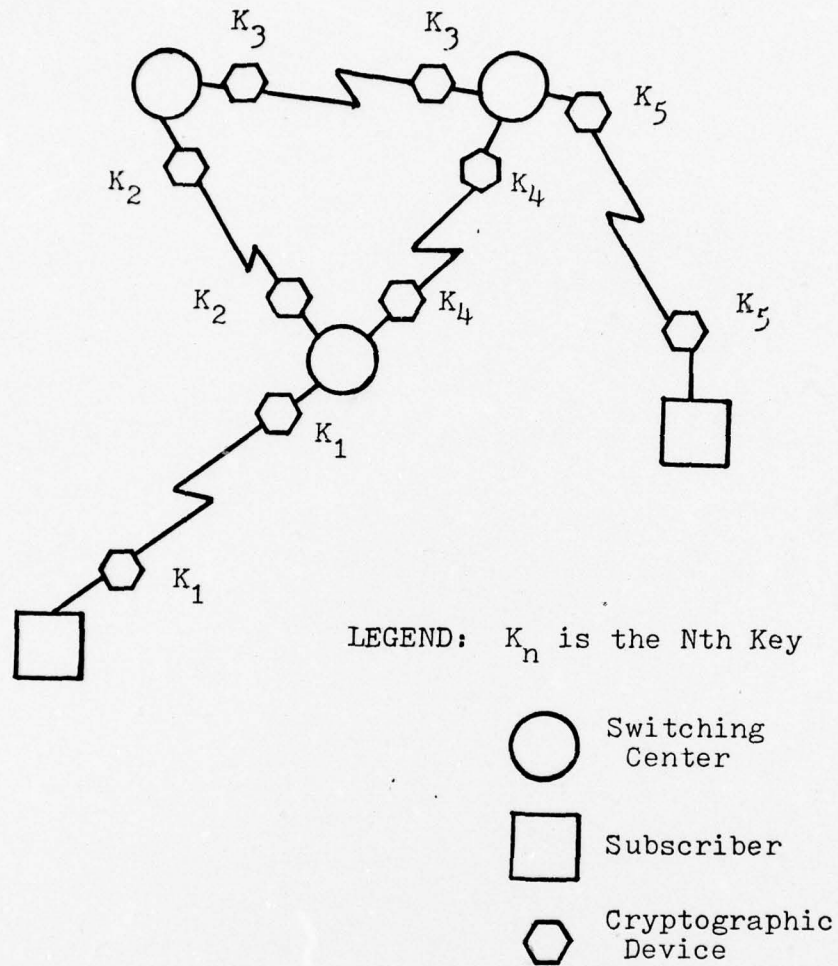


FIGURE 4.14

COMMUNICATIONS NETWORK WITH LINK BY LINK ENCRYPTION

is encrypted at the originating end with a key, K_1 . This key, K_1 , must then be available at the switching center. In general, each link requires a pair of encryption devices and a separate key. The separate keys are required to reduce the amount of traffic sent with any one key. If the message crosses four links and the same key is used on each link, then the amount of information in the key is four times the traffic volume. The number of keys and devices required for link by link encryption results in a potentially more expensive secure system than compared with end to end.

Since each link is encrypted and then decrypted, the link control and message header information is available in the clear for switching centers to route the messages. In effect, the use of link by link encryption is transparent to the communications system except as discussed in the next chapter on asynchronous communications protocols using block encryption.

Link encryption provides adequate protection against wiretapping. It does not protect against interception while in the switching center nor against potential misroutes, sending to the wrong addressee.⁵⁵ Link by link encryption can not currently be used with common carrier systems such as TWX and TELEX. These systems currently make no provisions for cryptographic systems at the switching centers.

End to End

With end to end encryption, only the originator and the addressees possess the key used to encrypt or decrypt the data. Since only the subscribers encrypt and decrypt the data, the number of cryptographic devices is reduced to the number of subscribers of the communications system. This reduction in the number of devices and keys results in a potential savings as compared with link by link encryption.

Using end to end encryption, requires that any link control or message header information required for message switching remain unencrypted. To accomplish this, the cryptographic device must be able to distinguish between the message header and the text. Therefore, end to end encryption devices could cost more, individually, than the link by link devices.

End to end encryption fully protects the message against wiretapping, misroutes, and interception within the switching center. Both public and private communications systems may be used.⁵⁶ Since only the addressees will have the decryption key, all potential addressees must be identified and provided with keys. The use of multiple keys allows several levels of security; several communities of interest can exist on the secured communications network.

AD-A046 887

AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO
SECURE COMMERCIAL DIGITAL COMMUNICATIONS.(U)

F/G 17/2

UNCLASSIFIED

JUL 77 P V ABENE
AFIT-CI-77-83

NL

2 OF 2
ADA046887



END
DATE
FILMED
12-77
DDC

The next two chapters will examine the application of FIPS PUB 46 Data Encryption Standard to digital communications systems using various protocols, both asynchronous and synchronous.

CHAPTER IV

Footnotes

1. Horst Feistel, "Cryptography and Computer Privacy," Scientific American, Vol. 228, No. 5 (May 1973), p. 16.
2. H.E. Petersen and R. Turn, "Systems Implications of Information Privacy," AFIPS Conference Proceedings 1967 Spring Joint Computer Conference, ARIPS PRESS, 1967, Vol. 30, p. 293.
3. R.O. Skatrud, "A Consideration of the Application of Cryptographic Techniques to Data Processing," AFIPS Proceedings of the 1969 Fall Joint Computer Conference, AFIPS Press, 1969, Vol. 35, p. 111.
4. Clyde E. Shannon, "Communications Theory of Secrecy Systems," The Bell System Technical Journal, Vol. 28, (1949), p. 662.
5. Skatrud, Op. cit., p. 111.
6. Shannon, Op. cit., pp. 669-670.
7. Feistel, Op. cit., p. 15.
8. Ibid., p. 15.
9. Petersen, Op. cit., p. 293.
10. Ibid., p. 293.
11. Ibid., p. 293.
12. Shannon, Op. cit., p. 662.
13. Ibid., pp. 708-710.
14. Ibid., pp. 657.
15. Ibid., pp. 708-710.

16. David Kahn, The Code Breakers, The New American Library Inc., 1967, p. 77.
17. Ibid., p. 76.
18. Ibid., p. 98.
19. Ibid., pp. 81-83.
20. Ibid., pp. 98-100.
21. Shannon, Op. cit., p. 682.
22. Kahn, Op. cit., p. 99.
23. Ibid., pp. 194-199.
24. Ibid., p. 198.
25. Shannon, Op. cit., p. 682.
26. Kahn, Op. cit., p. 198.
27. M.B. Girdansky, "Cryptology, The Computer, and Data Privacy," Computers and Automation, Vol. 21, No. 4 (April 1972), p. 16.
28. Terry Twigg, "Need to Keep Digital Data Secure?" Electronic Design, Vol. 20, No. 23 (November 9, 1972), p. 68.
29. Ibid., p. 71.
30. Ibid., p. 71.
31. Ibid., p. 71.
32. C.H. Meyer and W.L. Tuchman, "Pseudorandom Codes can be Cracked," Electronic Design, Vol. 20, No.23 (November 9, 1972), p. 74.
33. Ibid., pp. 75-76.
34. Philip R. Geffe, "How to Protect with Ciphers that are Really Hard to Break," Electronics, Vol. 46, No. 1 (January 4, 1973), p. 101.
35. Girdansky, Op. cit., p. 16-17.
36. Geffe, Op. cit., pp. 100-101.

37. Girdansky, Op. cit., p. 18.
38. Ibid., p. 12.
39. David J. Sykes, "Protecting Data by Encryption," Datamation, Vol. 22, No. 8 (August 1976), p. 82.
40. Ibid., p. 82.
41. Ibid., p. 82.
42. Ibid., p. 83.
43. Richard G. Cannind, ed., "Integrity and Security of Personnel Data," EDP Analyzer, Vol. 14, No. 4 (April 1976), p. 12.
44. Kahn, Op. cit., p. 381.
45. Sykes, Op. cit., p. 83.
46. Ibid., p. 83.
47. Herbert S. Bright and Richard L. Enison, "Cryptography Using Modular Software Elements," AFIPS Conference Proceedings 1976 National Computer Conference, AFIPS Press, 1976, Vol. 45, p. 117.
48. Horst Feistel, William A. Notz, and J. Lynn Smith, "Some Cryptographic Techniques for Machine-to-Machine Data Communications," Proceedings of the IEEE, Vol. 63, No. 11 (November 1975), pp. 1546-1548.
49. Ibid., p. 1548.
50. Data Encryption Standard, Federal Information Processing Standards Publication 46 (January 15, 1977), p. 2.
51. Ibid., p. 2.
52. Sykes, Op. cit., p. 83.
53. Feistel, Notz, and Smith, Op. cit., p. 1553.
54. Sykes, Op. cit., p. 83.
55. Ibid., p. 81.
56. Ibid., p. 81.

CHAPTER V

APPLICATION OF CRYPTOGRAPHIC SYSTEM TO ASYNCHRONOUS COMMUNICATIONS

This chapter will examine the application of FIPS PUB 46 Data Encryption Standard to two asynchronous character oriented, control procedures. The first procedure is a typical uncontrolled link between a terminal and a remote computer. The second control procedure uses the ANSI data link control procedure for two-way alternate, switched point-to-point operation.¹ The approach for the suggested design of the cryptographic device outlined in this chapter and the next assumes that the communications link between the modem and the terminal, or computer, is opened and the cryptographic device inserted with the least impact on the existing system.

This chapter examines the actions and operations of the cryptographic system involved in the transfer of encrypted information through the communication system using the block encryption/decryption algorithm of PUB 46 Data Encryption Standard. The first section of the chapter covers the user of a link by link cryptographic system and the second section covers the use of an end-

to-end cryptographic system. Each section will first discuss the pertinent aspects of the communications and cryptographic systems. The effects on operation, information throughput, and error rate will then be discussed.

Uncontrolled Link Operation

Description of Typical System

Uncontrolled link operations using asynchronous characters are typical of many applications intended to provide the simplest possible communications between a remote terminal and a computer. Examples of such systems include time sharing systems and remote data entry terminals. The terminal in this example is connected to the remote computer by a dedicated, two-wire, voice grade telephone link. The modems operate the physical link as two logical simplex channels using Frequency-Shift Keying (FSK) as shown in Figure 3.3. The characters are structured with one start bit, seven data bits, a parity bit, and one stop bit (a total of ten bits). The signalling rate of 300 bits per second results in thirty characters per second as the maximum transfer rate.

Each of the logical, simplex channels provides an uncontrolled communications path. One path transfers characters from the terminal to the remote computer. The other path provides communications in the reverse

direction. Since the links are uncontrolled, error control and transmission control information are provided out of channel by voice communications between the terminal and computer operators.

Data is sent from either the terminal's keyboard or papertape reader and is received on either the terminal's printer or papertape punch. A message from the keyboard is typically a short sequence of characters of varying length. The rate of entry of the characters depends on the typing skill of the operator. The papertape reader provides the means to enter long prepared messages. These messages are read and transmitted at a fixed rate of thirty characters per second. Data arrives at the terminal at a fixed rate of thirty characters per second from the computer. The message is either printed on paper or punched on papertape. Although the terminal generates and receives a parity bit for each character, the terminal takes no action based on a parity error.

Application of PUB 46 DES

This section will examine the implementation of a cryptographic system based on the Data Encryption Standard to the uncontrolled communications link described above. As an uncontrolled link does not require control information to remain unencrypted or to be decrypted for error control and message switching, the basic design of the

cryptographic device will equally apply to various forms of controlled communications links using link by link encryption. The major consideration in this design requires that the timely flow of information is not unduly delayed by the encryption process. While this section deals with the application of encryption to an uncontrolled, asynchronous communications link, the functional design of the device is applicable to link by link encryption systems needing only to be tailored by software changes to the microprocessors to meet the specific system requirements.

To be able to maintain a timely transfer of information using block encryption, the encryption device must add any necessary "null" characters to fill out a block. The PUB 46 algorithm operates on 64 bits representing eight characters in this example. The eighth initial characters transferred out of the terminal, or computer, will be delayed to comprise the initial block to be encrypted. After this initial delay, data transferred through the encryption device is transparent to the communications system as long as the data flow is continuous and in multiples of eight characters. If the data appears at an irregular rate slower than thirty characters per second or if the data is not in multiples of eight characters, the encryption device must fill an uncompleted block with "nulls" to insure that the data

contained in the unfilled block is not delayed. The decryption device must remove the "nulls" and deliver the data to the receiver (computer or terminal). The ASCII code set provides a NULL character (seven zero bits) that could be used to fill a block before encryption. Any unused combination of bits not needed to carry information could also be used.

In this example the NULL character is used to fill a block. The encryption device continuously examines the data line from the terminal (or computer). When a character is detected, the device stores the character. If eight characters have been received, the device encrypts the block and transmits the encrypted block to the distant end. After a fixed period of time (equal to one or more character times), the encryption device would fill out a partial block with NULL characters, encrypt the block, and then transmit it. The receiving decryption device examines each decrypted block and moves any NULL characters found. The remaining characters are transferred to computer (or terminal).

Figure 5.1 is a block diagram of a suggested arrangement of hardware for an encryption/decryption device. This arrangement is intended to allow the terminal or computer to be unplugged from the modem and the cryptographic device to be inserted. Data clocking must be

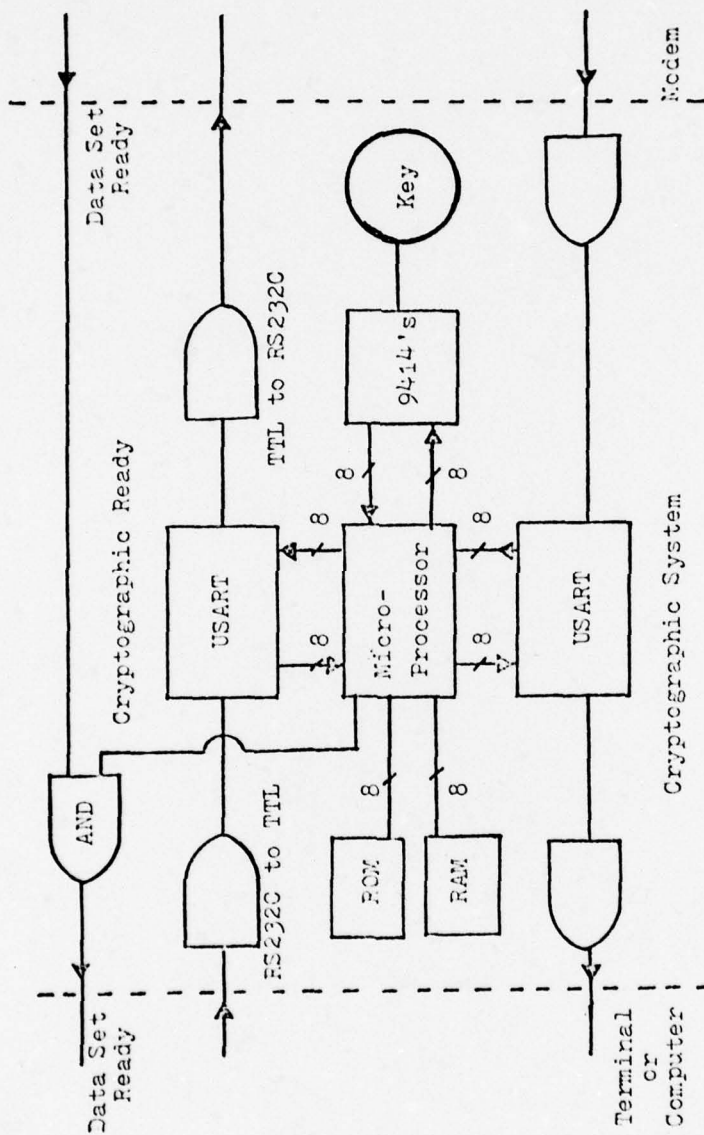


FIGURE 5.1
 BLOCK DIAGRAM OF CRYPTOGRAPHIC DEVICE
 FOR PUB 46 DES

provided to the cryptographic device from either the modem or the terminal (or computer). The cryptographic device develops its own clocking for the encryption process. The "data set ready" signal from the modem is "ANDed" with the cryptographic device ready. This new "data set ready" is supplied to the terminal or computer indicating that the communications and cryptographic equipment are ready.

Data received from the terminal, computer, or modem is converted from RS232C levels of +6, -6 volts to TTL levels of +5, 0 volts. A Universal Synchronous Asynchronous Receiver/Transmitter (USART) converts the received serial data to parallel characters removing the start and stop bits. The microcomputer stores the received character. After eight data characters have been received or after the microcomputer times out the block and adds the necessary NULL characters, the microcomputer transfers the block to the encryption element. After the encryption process is completed, the block is transferred back to the microcomputer. The microcomputer then transfers the data back to the USART for transmission to the modem, computer, or terminal. The output of the USART is converted from TTL levels back to RS232C levels.

The decryption process is nearly the same as outlined above with two exceptions. First, the cryptographic device is set for decryption. Second, after decryption,

all NULL characters are removed.

The key used to encrypt the data can be placed directly into the cryptographic device or into the microcomputer. The key could be inserted through the use of 16 hexadecimal switches or from a magnetic strip card reader such as used with bank credit cards. Each user of the terminal arranges with the computer operator to have the appropriate key placed in the cryptographic device at the computer end of the communications link.

After the control program for the microprocessor has been verified to be correct and error free, the program should be placed in read only memory to comply with PUB 46 DES requirements. The general design of the cryptographic system now is hardware/firmware rather than software.

The circuits to do the encryption and decryption could be the Fairchild 9414 LSI chip set. The set is composed of four, forty pin, chips each performing a part of the encryption/decryption process. With a 200 nanosecond clock period, the 9414 requires only 6.4 microseconds to process one block of data. Thus, the cryptographic system should be able to handle data rates in excess of 9600 bits per second on a half-duplex or full-duplex circuit. The block diagram of the 9414 set is shown in Figure 4.13.

Effects on Communications System

There are two major potential effects on the communications system caused by the addition of the cryptographic system. The first relates to information throughput. As outlined in the above section, the channel is continuously busy, although not always sending information. One advantage of this arrangement is that a person wire tapping the circuit could not detect the beginning and end of messages. On the other hand, the continuous transfer of characters could be a problem in a Time Division Multiplexed system if the system is designed to use the delay time between human typed characters. The logic of the proposed system could be modified to transmit a block containing "nulls" only if there was at least one data character (non "null") in the block. This would reduce the potential overhead, "null" characters. In the example outlined above, the added "null" characters do not present a problem since the link is dedicated to terminal/computer operation.

The second major effect is the impact on error conditions introduced by the communications system. As discussed in chapter four, a single bit in error in a data block after encryption will result in many bits in error in the decrypted block. Thus a single bit error could result in all the characters in the block being in error. For an uncontrolled operation, this error

extension does not change the operation except that more data may be received in error requiring the operators to call and request more retransmissions. Error recovery in the communications link would require at most one block to overcome the disruption of the link. Disruption caused by the use of the wrong cryptographic key is a potential. Such disruptions would require human operation intervention to change to the correct key.

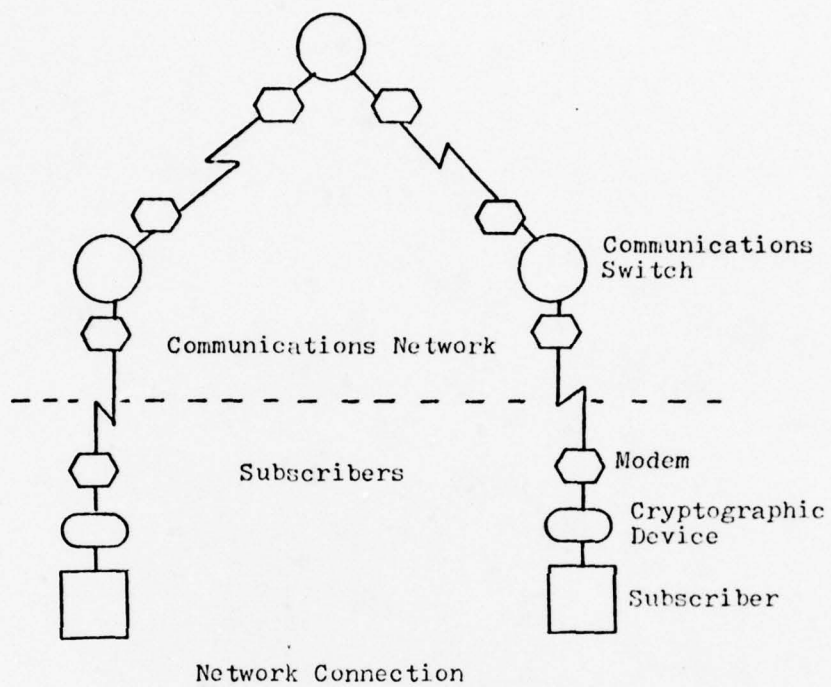
Controlled Asynchronous

Description of Typical System

In this example, the data to be secured passes through a digital communications network using two-way alternate, switched point-to-point communications. The exact details of this procedure are described in ANSI X3.28-1976.² The discussion in this section assumes the connection to the communications network has been completed and both the subscriber and the network are ready to transfer information. The cryptographic system in this example operates in the end-to-end mode; thus, the information necessary for message switching and error control must be transmitted in an unencrypted form.

Figure 5.2 is a block diagram of the system.

The message serves as the method of information transfer. As discussed in chapter three, the message is composed of three parts; the header, the text, and



S O H	Message Header	S T X	Meaasge Text	E T X	B C C
-------------	-------------------	-------------	-----------------	-------------	-------------

Message Format

FIGURE 5.2
 ASYNCHRONOUS COMMUNICATIONS SYSTEM WITH
 END TO END ENCRYPTION

the trailer. The header contains the information required to identify, control, and route the message as it passes through the communications network. In this example the trailer and the individual character parity bits provide the error detection. The message header begins with the ASCII character, SOH, and ends with STX. The text of the message is contained between the STX and the ETX characters. The message trailer consists of the ETX character and the Block Check Character, BCC. The BCC is the exclusive-or of the characters, without parity bit, of the message. All characters after the SOH including the ETX are used in computing the BCC.³ The encryption/decryption process will operate on the text characters of the message excluding the STX and ETX. The message is sent as one continuous transmission. Figure 5.2 contains a diagram of the message structure.

In addition to messages containing text information, there are control messages sent between the subscriber and the network. These control messages are used to accept the message from the master, ACK, or to request a retransmission of the message, NAK. If the master does not receive a reply after sending a message to the network, another control message, ENQ, is sent. The data messages are sent by the master and control messages are sent by either the slave or the master. To reverse roles the master sends an EOT character and assumes the slave

state. On receiving the EOT, the slave assumes the master state and send any data messages ready for transmission.⁴ The control messages must be transmitted unencrypted.

Each character sent in the communications system consists of seven bits plus an eighth parity bit. The control characters have the bit configurations shown in columns 0 and 1 of Table 3.1. The BCC character is seven bits plus an eighth parity bit. The text of the message consists of only the 64 characters contained in columns 2, 3, 4, and 5 of Table 3.1.

Application of PUB 46 DES

When added to the communications system, the cryptographic system must not change those portions of a message used to control the transfer of information. The cryptographic system must be able to identify the text of the message from all other communications on the link. The resulting encrypted text must comply with the characteristics of the communications characters. In this example the encrypted characters must appear to be from the set of 64 allowed text characters and have the proper parity bit. Since the characters of the text are changed, a new BCC must be computed for the entire message.

The basic design of the cryptographic system is

like that in Figure 5.1 with the difference being in the software of the microprocessor. The microprocessor examines the incoming characters from the subscriber looking for the message text. After finding the STX character, the microprocessor divides the text into ten character groups. If an ETX character occurs before a ten character group is completed, the group is filled with the "SPACE" characters to yield ten characters. After encryption is completed, the microprocessor expands the 64 bit encrypted block to eleven valid characters and computes the BCC character. When the last group has been sent, the microprocessor sends an ETX and the new BCC. The new BCC is computed based on the unencrypted message header and trailer, and the encrypted text.

To simplify the following discussions, the bits of a character are numbered as follows: P 7 6 5 4 3 2 1 . The P bit is the parity bit and bits 1-7 are the information bits.

The ten characters of a group must be reduced from 80 bits to 64 bits before encryption. As discussed above there are only 64 different characters (from Table 3.1, columns 2,3,4 and 5); thus, only six bits are required to represent a character. The structure of the ASCII code allows bit seven to be dropped resulting in the desired six bit characters. The parity bit and bit

seven of each of the ten characters in the group to be encrypted are dropped. The microprocessor concatenates the ten, six bit characters together resulting in a new 60 bit block. Four bits, zeros, are concatenated to the beginning in the 64 bit block for encryption.

After encryption, the microprocessor separates the 64 bit encrypted block into eleven ASCII characters. The 64 bit block is segmented into ten, six bit characters and one, four bit segment. Two bits are concatenated to the end of the four bit segment to give a six bit character. To convert the six bit characters back to valid ASCII characters the microprocessor sets bit seven to a one and computes the proper parity, P bit. These eleven ASCII characters are used to compute the new, partial BCC.

The decryption process is the reverse of the above procedure. The received characters are segmented into eleven character groups, bits seven and P are removed, bits two and one of the eleventh character are removed, and the resulting 64 bit block decrypted. The microprocessor converts the decrypted 64 bit block by dropping the last four bits and separates the remaining 60 bits into ten ASCII characters. The new BCC is then computed.

Effects on Communications Systems

The throughput on the communications system could be reduced by the addition of a cryptographic system. As discussed above, ten text characters are converted to eleven encrypted characters. This is a potential reduction in system throughput of ten percent and a corresponding increase in overhead. To accommodate this decrease in throughput, either existing unused channel capacity must be available or a channel with higher capacity must be acquired. Most commercial, digital communications networks charge based on the number of words or characters. Thus, the increase in the number of text words after encryption by ten percent would result in a high cost for the communications service.

One possible method of reducing the overhead, and increasing the throughput, would be to use a larger grouping size. A grouping of 128 unencrypted characters would result in 128 encrypted characters without any additional overhead. The encryption process for 128 characters would be divided into twelve blocks of 64 bits. Increasing the group size may not be acceptable to communications system operations due to the increased amount of buffer space in the microprocessor, increased delay time to encrypt a block, and increased overhead for messages shorter than 128 text characters.

The encryption process will change the effect of undetected errors on the message. As the message passes through the communications network, each receiving switch or subscriber checks the character parity and BCC bits for errors. If an error is found the receiver requests a retransmission. Using character parity and a BCC for error detection, the number of undetected errors is reduced by one hundred to one thousand.⁵ Thus, with typical error rates of one bit in error for each ten thousand sent, the undetected error rate is of the order of one bit in 10^7 to 10^8 characters sent.⁶ A single bit error within an encrypted group of eleven characters can be expected to change at least half the bits in the decrypted group of ten characters and to change all the characters in the block. Therefore a single undetected bit error in the encrypted text can reasonably be expected to change ten characters rather than one.

As described, the cryptographic system does not check for and act on errors in the received data. A change to the design could be made. Any errors detected by changes to character parity or the resulting BCC for the encrypted data could be forced into the decrypted data. Thus, after detecting the error, the subscriber would request a retransmission from the communications network by sending a NAK control message.

CHAPTER V

Footnotes

1. American National Standard Procedures for the Use of the Communication Control Characters of American National Standard Code for Information Interchange in Specified Data Communication Links, American National Standards Institute, 1976, X328-1976, p.32.
2. Ibid., p. 31.
3. Ibid., p. 28.
4. Ibid., p. 31.
5. James Martin, Security, Accuracy, and Privacy in Computer Systems, Prentice Hall Inc., 1973, p. 99.
6. Ibid., p. 99.

CHAPTER VI

APPLICATIONS OF CRYPTOGRAPHIC SYSTEM
TO SYNCHRONOUS COMMUNICATIONS

This chapter will discuss the application of a cryptographic system to the Synchronous Data Link Control (SDLC) procedure. This procedure requires the timing between characters and bits of a character to remain fairly constant. The cryptographic process must not disrupt this timing requirement. As in the previous chapter, the cryptographic system is assumed to have been inserted between the subscriber communications equipment and the modem allowing an existing system to be retrofited with the cryptographic system.

This chapter discusses the SDLC protocol, the communication system, and the cryptographic system. This discussion attempts to cover the major aspects that are impacted by the addition of the cryptographic system. The last section of this chapter covers the effects of the cryptographic device on the communications system.

Description of SDLC

The Synchronous Data Link Control, SDLC, procedure developed by IBM, provides the synchronous communications

protocol for either half-duplex or full-duplex link operation. The SDLC procedure evolved, in part, from the work done by the American National Standards Institute on the American National Standard for Advanced Data Communications Control Procedures (ADCCP). In addition to IBM, the Honeywell Information Systems Incorporated (HIS) has implemented a protocol based on ADCCP.¹ At least two manufacturers, Standard Microsystems and Signetics, have developed an LSI chip to perform most of the operations of the communications interface using SDLC or ADCCP. The LSI chips replace as many as 50 TTL chips.²

SDLC is based on a bit oriented protocol (as opposed to character oriented) for synchronous communications over either half-duplex or full-duplex links.³ The protocol provides for the transfer of data and control information across the communications link. The data may consist of any number of bits with any representation. The text is transparent to the communications process. The only constraint on the data requires that if the data sent is not a multiple of eight bits, that the data be filled with either ones or zeros to make it a multiple of eight bits.

The unit of information exchange using SDLC is a Frame. The basic structure of the Frame is shown in Figure 6.1. The Frame begins and ends with an eight

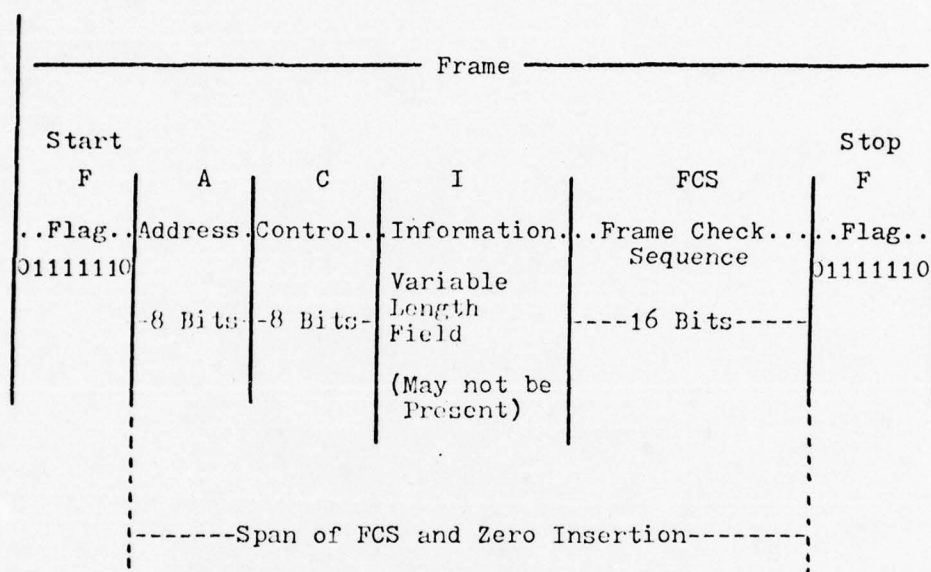


FIGURE 6.1

FIELDS OF THE SDLC TRANSMISSION FRAME

bit sequence called a Flag, 01111110. This sequence may not appear anywhere else in the Frame. To prevent the Flag sequence from falsely appearing in the Address, Control, Information, or Frame Check Sequence (FCS) fields, the transmitting communications controller inserts a zero bit after any sequence of five contiguous one bits. The receiving communications controller scans the incoming bit stream and removes any zero bit detected after five contiguous one bits in all fields except the Flag field. The adding of zero bits is called "bit stuffing."

The second field shown in Figure 6.1 is the Address field. This field is used to indicate the subscriber on a polled link to receive the frame. The Address field, treated as an eight bit unit, may address a single or a group of subscribers.⁴ A physical subscriber could be composed of several logical subscribers each with a separate address.

The third field, the Control field, provides the instructions to the transmitting and receiving controllers necessary to insure a smooth, orderly flow of information in a timely and error free manner. The Control field comprises several subfields used to effect the control and transfer of information. The structure of the Control field operates to allow acknowledgments

for up to seven frames. The specific definition and operation of the Control field is contained in IBM publication GA 27-3093, IBM Synchronous Data Link Control General Information.

The Information field, if present, carries the data to be transferred. There are no prior constraints on content or structure of the information field except as noted above. In general the length of the Information field is such as to allow a timely transfer of information on the link. Many small frames, as compared with single large frames, offer better throughput in communications links subjected to errors. Only the frame containing the error need be retransmitted.

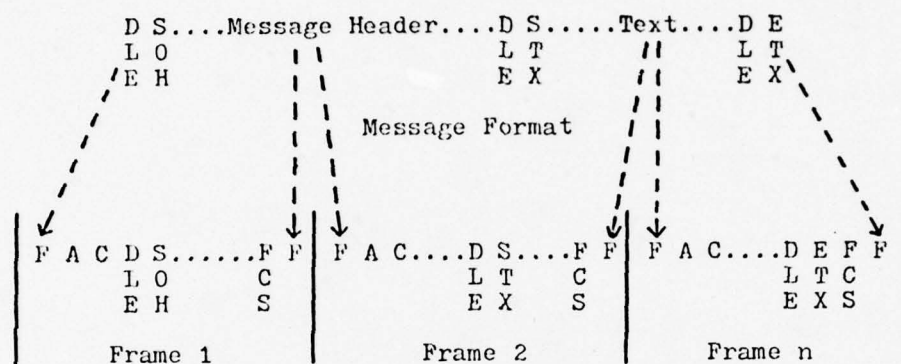
The Frame Check Sequence, FCS, provides the redundant information required for error detection. The FCS consists of sixteen bits developed as the remainder by dividing the Address, Control, and Information fields by the polynomial, $x^{16} + x^{12} + x^5 + 1$. This operation is sometimes called "cyclic redundancy checking." The transmitting controller computes the FCS prior to "bit stuffing" and the FCS is "bit stuffed" to remove any potential sequences of one bits longer than five contiguous one bits.⁵ The FCS is computed and checked for each link in the communications network. If an error is detected by the receiving controller, a retransmission of the frame is requested using the appropriate Control

field in the reply Frame. .

Description of Communications System

The communications network in the example has the same block structure as contained in Figure 5.1. The network receives a message consisting of one or more frames from a subscriber. Based on information contained in the header, the message is sent to one or more other subscribers. To accomplish this routing function the communications network must be able to access the message header in an unencrypted form. The message trailer needs only to consist of an end of message indicator since the error control is contained in the Frame Check Sequence of each Frame.

Figure 6.2 illustrates the basic message format and the method of segmenting the message into the Information fields for Frames. The message header may be less than one Frame, exactly one Frame, or be more than one Frame long. Part of the data would be added to the message header to complete a Frame. A total message may consist of many Frames or be completely contained in one Frame. The start of the message header containing the routing information for the communications switches is indicated by the two ASCII characters, DLE and SOH. The two ASCII characters, DLE and STX, indicate the start of message text; that part of the message



Message Segmented into SDLC Frames

Legend: ASCII Message Control Characters

SOH--Start Of Header

STX--Start Of Text

ETX--End Of Text

DLE--Data Link Escape

SDLC Fields

F----Flage Sequence

A----Address

C----Control

FCS--Frame Check Sequence

FIGURE 6.2

TRANSMISSION OF MESSAGE USING SDLC

to be encrypted. The end of text is indicated by the two ASCII characters, DLE and ETX. The DLE character is required to insure that control characters; SOH, STX, and ETX; are not confused with the encrypted text. The use of the DLE character will be further discussed below.

Application of PUB 46 DES

In this example the encryption device will operate in the end to end mode. The message header and trailer and the SDLC Flag, Address, Control, and Frame Check Sequence must remain unencrypted. The cryptographic device is to be added to the system in such a manner as to not disrupt the flow of information and the communications system operations. Since the device is inserted into the communications path after the communications controller of the subscriber, the cryptographic device must perform some of the same basic functions of the controller. Specifically, the cryptographic device must check for the start and end of a Frame indicated by the Flag sequence, for the Abort sequence, for the Information field, and for the Frame Check Sequence. If the Information field contains text, the device must either encrypt or decrypt the text and then update the Frame Check Sequence to reflect the change. The following is a more detailed discussion of the operation of the device. Figure 6.3 is a block diagram of the device.

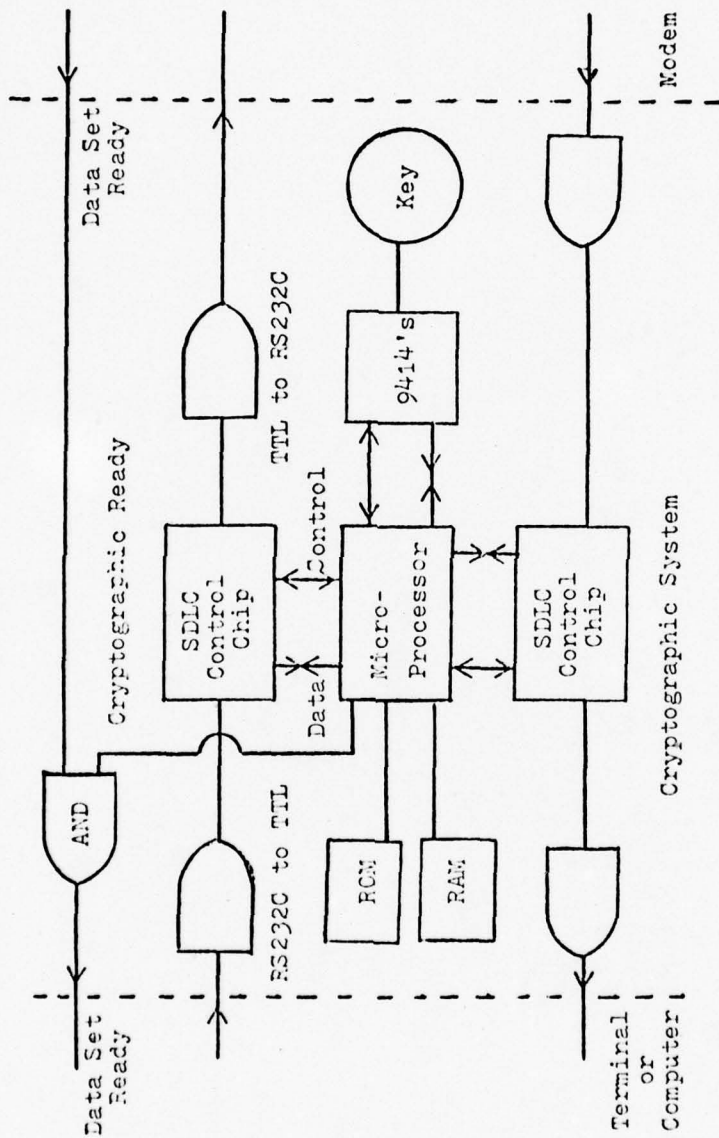


FIGURE 6.3

PUB 46 DES CRYPTOGRAPHIC DEVICE FOR
SDLC PROTOCOL

The discussion will follow a message out of the subscriber communications controller through the encryption process in the cryptographic device to the modem for transmission to the communications network.

The cryptographic device monitors the communications link looking for the beginning of a Frame. The LSI chip for the SDLC protocol hunts for the Flag sequence. Once it finds the sequence, the character synchronization is established. Each eight bit group is examined looking for the Flag sequence. If the eight bit sequence is not a Flag, then the SDLC chip assumes the beginning of the Address field. From this point until another Flag is found, the SDLC chip extracts a zero bit that was preceded by five contiguous one bits and provides the data stream to the microprocessor. The SDLC chip detects and transfers the Address, Control and Information fields and passes them to the microprocessor. In addition, the SDLC chip is computing the Frame Check Sequence from the received data. On detecting the ending Flag, the SDLC chip will indicate to the microprocessor whether the Frame passed or failed the error detection process. The SDLC chip also hunts for the Abort sequence, eight contiguous one bits, indicating that the Frame is to be closed and that the link is to stop operation. The indication of the Abort sequence is provided to the microprocessor.

After being notified by the SDLC chip of the start of a Frame, the microprocessor examines the next eight bits, the Address field, insuring that address is for the connected subscriber. If the address does not check, the microprocessor merely passed the Frame back to the SDLC for subsequent transmission. (In this example if the Address field does not check, the subscriber has made an error since this is not a polled network.)

The next eight bits after the Address field form the Control field. The microprocessor examines the Control field, but does not change it, looking for the indication of an Information field. If there is not an Information field indicated, the microprocessor returns the entire Frame as received to the SDLC chip for subsequent transmission.

Since the speed of the encryption/decryption element is 6.4 microseconds, assuming the Fairchild 9414 chip set is used; the microprocessor needs only to delay the start of transmission of a Frame containing an Information field on the order of fifteen to eighteen character times to allow for the gathering of 64 bits of text for the encryption. Thus, all Frames that contain Information fields will be delayed even though all delayed Frames will not be encrypted. This delay of the entire Frame is required to maintain synchronization on the link.

The received Information field is examined for an indication of the start of the text. In this example, the data in the Information field is transmitted as eight bit ASCII characters. Thus, the first eight bits of the Information field form the first character with each subsequent character being formed from each subsequent eight bit group. Each character received is examined for the ASCII DLE character followed by the STX character indicating the start of text. The microprocessor sets an indicator to signify that the encryption process has begun. All subsequent Frames containing an Information field are encrypted until the indicator is reset by the End of Text sequence for the message. The microprocessor gathers the data into blocks of 64 bits for encryption (or decryption) and transfers the data to the cryptographic device. After the cryptographic device has completed the operation, the encrypted data is transferred back to the microprocessor. The encryption process continues until the microprocessor finds the end of text indicator, the ASCII DLE character followed by the ETX character. If the ETX character is found before a block of 64 bit is created, a "null" sequence must be added to fill the block to 64 bits (see chapter five for a full discussion of the "null" sequence). If the Frame ends before a 64 bit block is filled, "nulls" are added prior to encryption.

Since the encrypted data can be any pattern, including the ETX pattern; microprocessor must create an unambiguous indicator of the end of text for the decryption device. This indicator is created by the using of the Data Link Escape, DLE, character of the ASCII code set. After the data has been encrypted, the microprocessor examines each of the eight bit characters in the 64 bit encrypted block looking for an occurrence of the DLE character. If a DLE character is found, another DLE character is inserted, making a block of nine characters. After the last block of text has been encrypted and transmitted, the microprocessor inserts a DLE and then a ETX character in the data stream transferred to the SDLC chip.

The microprocessor in the receiving decryption device checks for these two sequences of characters; DLE, DLE, and DLE, ETX. On finding the DLE, DLE sequence; the microprocessor extracts one of the DLE characters before the block is decrypted. On finding the DLE, ETX sequence; the microprocessor has received the end of text indication. Throughout the decryption process, the microprocessor checks the decrypted data for "null" characters and removes them on any occurrence. The resulting decrypted data is transferred to the SDLC chip for transmission to the subscriber.

After the microprocessor has finished a Frame, it signals the SDLC chip to complete computation and to transmit the new Frame Check sequence. Thus any change made to the Information field by the encryption or decryption process will be reflected in the Frame Check Sequence used by the rest of the communications network and the subscribers.

Effects on Communications System

Throughput

The throughput of the communications system could be impacted by the addition of the cryptographic system in two ways. First, the delay of the Frame to allow for the encryption of the information could cause the system to time out and consider the link to be nonoperational. The delay of the Frame requires the microprocessor to be able to buffer at least as many characters as are delayed. The characters delayed could be a complete Frame containing only control information. The microprocessor must be able to properly process the delayed Frame without disruption of the communications process.

The second effect on throughput is caused by the addition of the DLE and "null" characters to the encrypted data. These added characters serve to increase the system overhead as they do not contain information but rather control information. Although the SDLC

protocol makes no restriction on the length of the Information field, the communications system must be able to accept the increased Frame size caused by the added characters after the encryption of the data. The buffer space in the various communication switches must be defined to allow for Frames with the increased length.

Error Rate

Since a new Frame Check Sequence is computed after the encryption process, the error detection capabilities of the SDLC protocol are not affected. If the cryptographic device detects an error in a Frame received from the subscriber or from the communications system, the device must be able to pass the indication of the error to the subscriber and to have the subscriber take the appropriate action to effect a retransmission of the Frame.

Any indicated error will potentially have a disruptive effect on the communications process and must be handled at a higher level than the SDLC procedures. A single undetected bit error could change all eight characters in the data block after decryption. An undetected error could change the DLE-ETX sequence causing the meaning to be lost. Thus, the decryption device would attempt to decrypt Frames that had not been encrypted. It should be noted that the undetected

error rate using a sixteen bit Frame Check Sequence is of the order of one bit in error for 10^8 characters sent.⁶

CHAPTER VI

Footnotes

1. Stuart B. Cooper, "Hardware Considerations for High Level Data Link Control Communications," Computer Design, Vol. 14, No. 3 (March, 1975), p. 81.
2. "Digital Gets SDLC IC for Multi-Protocol Use," Electronic Engineering Times, No. 106 (June 27, 1977), p. 4.
3. R.A. Donnam and J.P. Kersey, "Synchronous Data Link Control: A Perspective," IBM Systems Journal, Vol. 13, No. 2 (March, 1974), p. 149.
4. Ibid., p. 153.
5. Ibid., p. 157.
6. James Martin, Security, Accuracy, and Privacy in Computer Systems, Prentice-Hall Inc., 1973, p. 109.

CHAPTER VII

SUMMARY

This thesis has addressed the rationale for secure commercial digital communications. Using the information presented in this thesis, the communications manager can estimate the potential risks to the communications system. Knowing the risks, the potential losses, and the expenses of securing the communications system, the manager can determine the measures needed to protect the company communications from an intruder. The communications manager must use an economic evaluation to insure the cost of the cryptographic system is justified based on the potential loss.

The discussions on cryptographic systems have shown the weaknesses of past cryptographic systems and of most currently available commercial systems. The primary weakness of these systems is their lack of sufficient confusion and diffusion of the data during the encryption process. This weakness reduces the amount of work required to break the encryption process. Decreased cost and increased computing power of computers has reduced the amount of time to break these systems.

The Data Encryption Standard in FIPS PUB 46 has been shown to offer greatly increased security for digital communications. The encryption algorithm used in PUB 46 DES has been shown to be adaptable to a variety of communications protocols. This thesis has shown by examples that the algorithm could be adapted to existing asynchronous and synchronous communications protocol. The cryptographic equipment can be added to the communications system with minimum impact on the system throughput and error control. These examples were intended to provide the bases for the design of a cryptographic system to be tailored by the communications engineer. Future work relating to the use of secure communications should address the integration of the cryptographic system into the communications system rather than as an add-on to the existing communications system. Integrating the two systems should reduce the cost of implementation and increases the efficiency of operation.

BIBLIOGRAPHY

- Armer, Paul, "Computer Technology and Surveillance," Computers and People, Vol. 24, No. 9 (Sept., 1975), pp. 8-11.
- American National Standards Institute, American National Standard Procedures for the Communication Control Characters of the American National Standard Code for Information Interchange in Specified Data Communication Links, New York, American National Standards Institute, Vol. X3.28-1976, 71 pp.
- Brehm, Bob, "Using a Keyboard ROM," Byte, Vol. 2, No. 5 (May, 1977), p. 82.
- Bright, Herbert S. and Richard L. Enison, "Cryptography Using Modular Software Elements," AFIPS Conference Proceedings 1976 National Computer Conference, Vol. 45 (1976), pp. 113-123.
- Burris, Harrison R., "Computer Network Cryptographic Engineering," AFIPS Conference Proceedings 1976 National Computer Conference, Vol. 45 (1976), pp. 91-96.
- Cannind, Richard G., "Integrity and Security of Personal Data," EDP Analyzer, Vol. 14, No. 4 (April, 1976), pp. 1-14.
- Cooper, Stuart B., "Hardware Considerations for High Level Data Link Control Communications," Computer Design, Vol. 14, No. 3 (March, 1975), pp. 81-87.
- David, Heather M., "Computers, Privacy, and Security," Computer Decisions, Vol. 7, No. 2 (February, 1975), pp. 28-30.
- _____, "Computers, Privacy, and Security," Computer Decisions, Vol. 6, No. 5 (May, 1974), pp. 46-48.
- Davies, Donald W. and Derek L.A. Barber, Communications Networks for Computers, London, John Wiley and Sons, 1973, 575 pp.

- Donnam, R.A. and J.P. Kersey, "Synchronous Data Link Control: A Perspective," IBM Systems Journal, Vol. 13, No. 2 (March, 1974), pp. 140-162.
- Electronic Engineering Times, "Digital Gets SDLC IC for Multi-Protocol Use," Electronic Engineering Times, No. 106 (June 27, 1977), p. 4.
- Feistel, Horst, "Cryptograph and Computer Privacy," Scientific American, Vol. 228, No. 5 (May, 1973) pp. 15-23.
- Feistel, Horst, William A. Notz, and J. Smith, "Some Cryptographic Techniques for Machine-to-Machine Data Communications," Proceedings of the IEEE, Vol. 63, No. 11 (November, 1975), pp. 1545-1554.
- Firmbery, David, "Your Computer in Jeopardy," Computer Decisions, Vol. 8, No. 7 (July, 1976), pp. 28-30.
- Franson, Paul, "Scrambled Data Baffles Thieves," Electronics, Vol. 45, No. 2 (January 17, 1972), pp. 87-88.
- Geffe, Philip R., "How to Protect with Ciphers that are Really Hard to Break," Electronics, Vol. 46, No. 1 (January 4, 1973), pp. 99-101.
- Girsdansky, M.B., "Cryptology, the Computer and Data Privacy," Computers and Automation, Vol. 21, No. 4 (April, 1972), pp. 12-19.
- Hamilton, Peter, Computer Security, London, Associated Business Programmes Ltd., 1972, 122 pp.
- Heinrick, Frank R. and David J. Kaufman, "Centralized Approach to Computer Network Security," AFIPS Conference Proceedings 1976 National Computer Conference, Vol. 45 (1976), pp. 85-90.
- Kahn, David, The Code Breakers, New York, New American Library, Inc., 1967, 476 pp.
- Seibholz, Stephen W. and Louis D. Vitson, User's Guide to Computer Crime, Radnor, Pa., Chilton Book Company, 1974, 204 pp.
- Lobel, Jerome, "Planning a Secure System," Journal of Systems Management, Vol. 27, No. 7 (July, 1976), pp. 14-19.

- McCalmont, Arnold M., "Communications Security for Voice - Techniques, Systems and Operations," Telecommunications, Vol. 7, No. 4 (April, 1973), pp. 35-42.
- Mason, John F., "Designers Compete for Snug Electronic Bug in a Rub," Electronic Design, Vol. 21, No. 20 (September 27, 1973), pp. 22-30.
- Martin, James, Security, Accuracy, and Privacy in Computer Systems, Englewood Cliffs, New Jersey, Prentice-Hall, Inc., 1973, 626 pp.
- Menkus, Belden, "Management's Responsibilities for Safeguarding Information," Journal of Systems Management, Vol. 27, No. 12 (December, 1976), pp. 32-38.
- Meyer, C.H. and W.L. Tuchman, "Pseudorandom Codes Can be Cracked," Electronic Design, Vol. 20, No. 23 (November 9, 1972), pp. 74-76.
- Milligan, Robert H., "Management Guide to Computer Protection," Journal of System Management, Vol. 27, No. 11 (November 1976), pp. 14-18.
- National Bureau of Standards, Guidelines for Automatic Data Processing, Physical Security, and Risk Management, Federal Information Processing Standards Publication 31, (June, 1974), 92 pp.
- National Bureau of Standards, Data Encryption Standard, Federal Information Processing Standards Publication 46, (January 15, 1977), 18 pp.
- Neumann, Albercht J., Brian G. Lucas, Justin C. Walker, and Dennis W. Fife, A Technical Guide to Computer-Communications Interface Standards, National Bureau of Standards, 1974, 104 pp.
- Petersen, H.E. and R. Turn, "Systems Implications of Information Privacy," AFIPS Conference Proceedings 1967 Spring Joint Computer Conference, Vol. 30 (1967), pp. 291-300.
- Rixon, Inc., Technical User Manual II, Communications Theory, USA, Rixon Inc., 1973, 52 pp.
- Shannon, Clyde E, "Communications Theory of Secrecy Systems," The Bell System Technical Journal, Vol. 28 (1949), pp. 656-715.

Skatrud, R.O., "A Consideration of the Application of Cryptographic Techniques to Data Processing," AFIPS Proceedings of the 1969 Fall Joint Computer Conference, Vol. 35 (1969), pp. 111-117.

Sykes, David J., "Protecting Data by Encryption," Datamation, Vol. 22, No. 8 (August, 1976), pp.81-85.

Twigg, Terry, "Need to Keep Digital Data Secure?," Electronic Design, Vol. 20, No. 23 (November 9, 1972), pp. 68-71.

Ware, Willis H., "Security and Privacy in Computer Systems," AFIPS Conference Proceedings 1967 Spring Joint Computer Conference, Vol. 30 (1967), pp. 279-284.

Williams, David, "FC&I Chip Set to use IBM Encryption Algorithm," Electronic News, Vol. 22, No. 1121 (February 28, 1977), p. 63.

Abene, Peter V.

Secure Commercial Digital Communications

Captain, United States Air Force

1977, 131 Pages

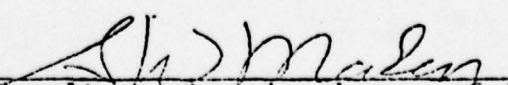
Master of Science, Telecommunications

University of Colorado

With the ever increasing dependence of businesses on digital communications, the potential for the loss of information and privacy to third parties increases. This thesis addresses this potential loss during the communications process and develops the rationale for secure digital communications links. After discussing pertinent aspects of digital communications, a look at past cryptographic methods provides an understanding of their weaknesses. Using the Federal Information Processing Standards Publication 46, Data Encryption Standard, the thesis provides examples of methods to retrofit existing asynchronous and synchronous digital communications systems with a cryptographic system. Some of the effects of the cryptographic system on the communications system operations, throughput, and error control are discussed.

This abstract is approved as to form and content.

Signed


Faculty member in charge of thesis