

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

2

NCS TIB 83-7



NATIONAL COMMUNICATIONS SYSTEM

**TECHNICAL INFORMATION BULLETIN
83-7**

AD A137592

**MIXED MODE
FOR GROUP 4 FACSIMILE
SYSTEMS**

NOVEMBER 1983

DTIC
ELECTE
S FEB 6 1984 **I**

D

**APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED**

DTIC FILE COPY

84 02 06 048

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NCS-TIB-83-7	2. GOVT ACCESSION NO. AD-A137592	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Mixed Mode for Group 4 Facsimile Systems	5. TYPE OF REPORT & PERIOD COVERED Final Report	
	6. PERFORMING ORG. REPORT NUMBER	
7. AUTHOR(s) Alan Deutermann	8. CONTRACT OR GRANT NUMBER(s) DCA100-82-C-0047	
9. PERFORMING ORGANIZATION NAME AND ADDRESS Delta Information Systems 310 Cottman Street Jenkintown, PA 19046	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
11. CONTROLLING OFFICE NAME AND ADDRESS National Communications System Attn: NCS-TS Washington, D.C. 20305	12. REPORT DATE November 7, 1983	
	13. NUMBER OF PAGES 122	
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	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		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Mixed Mode, Facsimile, Group 4, Segmentation		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The purpose of this effort is to investigate the more advanced Mixed Mode coding technique which will be one service of Group 4 facsimile, In a mixed mode system the information printed on a page is divided into two parts - symbols (letters, numerals, punctuation, etc.) and graphics (logos, signatures, sketches, etc.). This study examines possible techniques for segmenting a document into graphic and symbol areas, and assemble a code that represents the entire document. Four segmentation techniques were selected for analysis as follows: SYMBOL REMOVAL/SCAN LINE; SYMBOL REMOVAL/LINE OF SYMBOLS; EXTENDED TELETEx; SYMBOL		

REMOVAL/HYBRID. These techniques were designed to differ from each other as much as possible, so as to display a wide variety of characteristics. For each technique, many minor modifications would be possible, but it is not expected that these modifications would alter the conclusions drawn from the study.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A/1	



NCS TECHNICAL INFORMATION BULLETIN 83-7

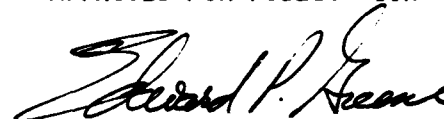
MIXED MODE FOR GROUP 4
FACSIMILE SYSTEMS

NOVEMBER 1983

PROJECT OFFICER

DENNIS BODSON
Senior Electronics Engineer
Office of NCS Technology
and Standards

APPROVED FOR PUBLICATION:



EDWARD P. GREENE
Acting Assistant Manager
Office of NCS Technology
and Standards

FOREWORD

Among the responsibilities assigned to the Office of the Manager, National Communications System, is the management of the Federal Telecommunication Standards Program. Under this program, the NCS, with the assistance of the Federal Telecommunication Standards Committee identifies, develops, and coordinates proposed Federal Standards which either contribute to the interoperability of functionally similar Federal telecommunication systems or to the achievement of a compatible and efficient interface between computer and telecommunication systems. In developing and coordinating these standards, a considerable amount of effort is expended in initiating and pursuing joint standards development efforts with appropriate technical committees of the Electronic Industries Association, the American National Standards Institute, the International Organization for Standardization, and the International Telegraph and Telephone Consultative Committee of the International Telecommunication Union. This Technical Information Bulletin presents an overview of an effort which is contributing to the development of compatible Federal, national, and international standards in the area of digital facsimile standards. It has been prepared to inform interested Federal activities of the progress of these efforts. Any comments, inputs or statements of requirements which could assist in the advancement of this work are welcome and should be addressed to:

Office of the Manager
National Communications Systems
ATTN: NCS-TS
Washington, DC 20305
(202) 692-2124



DELTA INFORMATION SYSTEMS, INC.

310 COTTMAN STREET JENKINTOWN, PA 19046
(215) 572-5640

**MIXED MODE
FOR GROUP 4 FACSIMILE SYSTEMS**

November 7, 1983

Final Report

Submitted to:

**NATIONAL COMMUNICATIONS SYSTEM
Office of Technology and Standards
Washington, D.C. 20305**

Contracting Agency:

DEFENSE COMMUNICATIONS AGENCY

Contract Number

DCA100-83-C-0047

Submitted by:

DELTA INFORMATION SYSTEMS, INC.

Table of Contents

1.0 Introduction	1-1
2.0 Develop Candidate Mixed Mode Algorithm	2-1
2.1 Symbol Removal/Scan Line	2-1
2.3 Symbol Removal/Line of Symbols	2-2
2.3 Extended Teletex-CR/LF option	2-3
2.4 Symbol Removal/Hybrid	2-4
3.0 Task 2 - Measurement of Compression	3-1
3.1 Methodology for Measuring Comp.	3-1
3.2 Assumptions	3-1
3.3 Calculating Compression	3-3
3.4 Scanned Document - CCITT No. 1	3-6
3.5 Computer Generated Documents	3-25
4.0 Task 3 - Analysis of Results	4-1
4.1 Compression	4-1
4.2 Complexity of Implementation	4-3
4.3 Commonality	4-4
5.0 Conclusion & Recommendations	5-1

Appendix A: CCITT Draft Recommendation S.a

Appendix B: Combined Symbol Matching Facsimile Data Compression System

1.0 Introduction

This document summarizes work performed by Delta Information Systems, Inc. for the Office of Technology and Standards of the National Communications System, an organization of the U. S. Government, under contract Number DCA100-83-C-0047. The Office of Technology and Standards, headed by National Communications System Assistant Manager Marshall L. Cain, is responsible for the management of the Federal Telecommunications Standards Program, which develops telecommunication standards whose use is mandatory by all Federal agencies.

Consideration is now being given to possible CCITT standards for Group 4 Facsimile which refers to the transmission of an A4 sized page over data networks containing error control. It is likely that the basic coding technique for Group 4 transmissions will be some advanced form of the Modified READ code, which is the optional compression algorithm for Group 3. The purpose of this study is to investigate the more advanced Mixed Mode coding technique which will be one service of Group 4 as shown in Figure 1-1. In a mixed mode system the information printed on a page is divided into two parts - symbols (letters, numerals, punctuation, etc.) and graphics (logos, signatures, sketches, etc.) The purpose of this study was to examine possible techniques for segmenting a document into graphic and symbol areas, and assemble a code that represents the entire document.

CLASS		1	2	3
SERVICE				
	FACSIMILE	SEND/RECEIVE	SEND/RECEIVE	SEND/RECEIVE
SERVICE	TELETEX	-	RECEIVE	GENERATE/ TRANSMIT/ RECEIVE
	MIXED MODE	-	RECEIVE	GENERATE/ TRANSMIT/ RECEIVE
TRANSMIT RESOLUTION	STANDARD	200	200 and 300	200 and 300
	OPTIONAL	240,300,400	240,400	240,400
PEL CONVERSION		NOT REQUIRED	YES	YES
PAGE MEMORY		NOT REQUIRED	YES	YES

CLASSES OF GROUP 4 TERMINALS

FIGURE 1-1

Parameters to be considered include compression, commonality with facsimile and TELETEX ^{1/} transmissions, and complexity of implementation.

Four segmentation techniques were selected for analysis. The techniques were designed to differ from each other as much as possible, so as to display a wide variety of characteristics. For each technique, many minor modifications would be possible, but it is not expected that these modifications would alter the conclusions drawn from the study.

The segmentation techniques analyzed are:

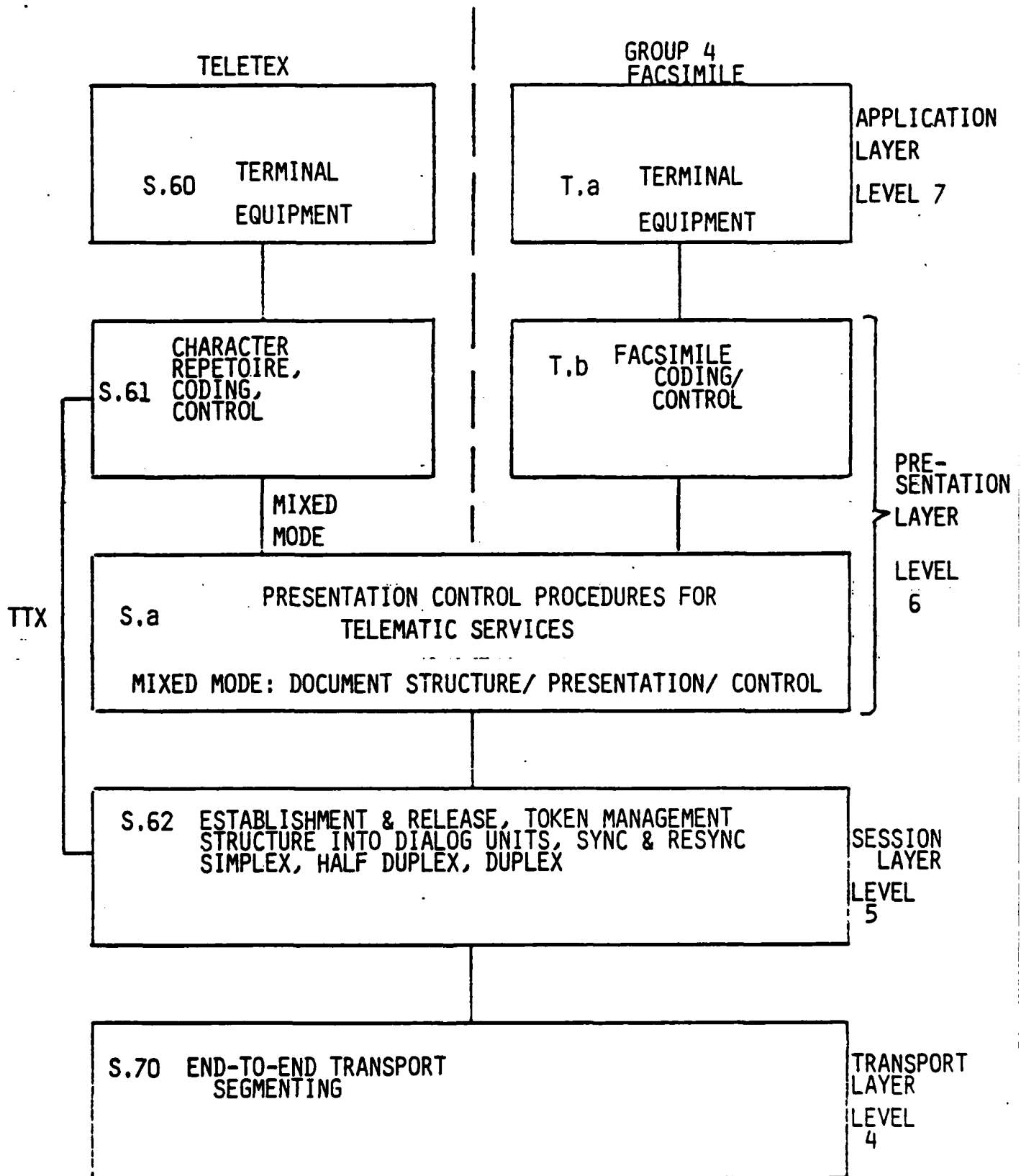
- SYMBOL REMOVAL/SCAN LINE
- SYMBOL REMOVAL/LINE OF SYMBOLS
- EXTENDED TELETEX
- SYMBOL REMOVAL/HYBRID

Section 2 presents descriptions of the four mixed-mode segmentation alternatives considered. Section 3 describes the assumptions and methodology for measuring compression, and the compression computations themselves. Section 4 discusses the commonality of each alternative with Group 3 facsimile, Group 4 facsimile, and TELETEX. It also summarizes compression and discusses the complexity of implementation of each technique. Finally Section 5 compares the alternatives and draws conclusions.

The CCITT has determined that the 7 layer OSI (Open System

^{1/} TELETEX refers to a CCITT recommendation which is now under development for communication between word processors.

Interconnect) protocol which has been developed by the ISO (International Standards Organization) will be used for Group 4 facsimile. Figure 1-2 illustrates the top 4 OSI levels emphasizing the relationship between the Teletex and Group 4 facsimile services. The S. and T. are the designations of the CCITT Recommendations for each protocol layer. Note that S.a is the key recommendation for mixed mode operation. This standard has not yet been finalized. The most recent draft of this recommendation is included in Appendix A for reference purposes.



FRAMEWORK OF CCITT RECOMMENDATIONS
FOR GROUP 4 FACSIMILE APPARATUS

FIGURE 1-2

2.0 Task 1 - Develop Candidate Mixed Mode Algorithms

Four mixed-mode segmentation techniques are selected for consideration in this study. The techniques are:

Symbol Removal/Scan Line

Symbol Removal/Line of Symbols

Extended Teletex

Symbol Removal/Hybrid

In the three symbol removal techniques, the black pels associated with recognized symbols are coded and "removed" (changed to white), and then the entire document is encoded using the Modified READ code, including areas where the symbols were. In the Extended TELETEX technique, the Modified READ code is used only for areas that do not have encoded characters. All techniques presume existence of a stored library of symbols.

2.1 Symbol Removal/Scan Line

This coding technique is very similar to the Combined Symbol Matching algorithm which is described in Appendix B. In this approach the document is scanned, from top to bottom, and from left to right, until a group of black pels is encountered that matches a symbol in the stored library. All black pels within the rectangular symbol space are then changed to white, and the symbol code and position are recorded. After the symbols have been "removed", the document is rescanned in principle and encoded

using the Modified READ code ($k=00$, no EOL code). The detected symbol codes are inserted before the READ code of the scan line in which the top of the symbol occurs. The presence of a symbol code rather than a READ code, is indicated by a single bit at the beginning of every scan line. If the bit indicates that there are symbols on the scan line, the 8-bit symbol code follows, and this in turn is followed by an 11-bit horizontal position code word, ($2^{11}=2,048$, being greater than the 1,728 pels in the scan line). This may be followed by additional symbol/horizontal-position code pairs for any other symbols that may have been detected on the scan line (in order of horizontal position). Finally, the symbol data is terminated by a special 8-bit symbol code that indicates there are no more symbols on the scan line. Then the READ code for that line is transmitted.

Notice that in this technique the recognized symbols will be encoded as they are first encountered by the scanning process, regardless of where they appear relative to other symbols or graphics. The vertical position of the symbols is implied by the scan line on which the symbol code appears.

2.2 Symbol Removal/Line of Symbols

In this technique, as in other symbol removal approaches, the symbols are detected, "removed", and their codes and positions are recorded. The symbols are then organized into lines of symbols, based on symbol position, height, hang down, etc. Account is taken of small amounts of line skew, and a single vertical position is assigned to the entire line of symbols. When this

process is complete, each printed line on the document should be contained within a line of symbols. Spaces between symbols having several different widths up to about 2 normal symbol spaces, are filled with appropriate blank characters. If the space between symbols is greater than 2 symbol spaces, the line of symbols is broken into segments.

The entire document, less recognized symbols, is transmitted using Modified READ code. When a scan line having the vertical position of a line of recognized symbols is encountered, a special 12-bit code (which could be an EOL code) is inserted. This changes the mode from graphics to symbols. This is followed by an 11-bit code giving the horizontal position of the first symbol. Then the symbol codes for each symbol in the segment are sent, followed by a special 8-bit end-of-segment symbol code. This, in turn, is followed by an 11-bit distance-to-the-next-segment symbol code. The last segment of symbols on the line is followed by a special 8-bit end-of-line symbol code instead of the end-of-segment code. This changes the mode back to graphics, and Modified READ code is continued until another scan line with a line of symbols is encountered.

As with the other symbol removal techniques, a recognized symbol will be encoded wherever it is located, since lines of symbols may overlap vertically, and each line of symbols may contain as few as one symbol. There may be some inaccuracies in positioning symbols, since spaces between symbols of 1 or 2 pels will probably not be encoded, and the horizontal position of a symbol code could be in error at the end of a long line of

symbols.

2.3 Extended TELETEX - CR/LF option

In this approach the entire document is divided into character spaces, except for areas that are defined as graphics, as discussed below. All character symbols, including blanks, are transmitted using 8-bit symbol codes.

The graphics are transmitted by Modified READ code as they occur within a line of symbols. First, a special 8-bit symbol code is used to designate the transition from symbol codes to graphics. This is followed by an 11-bit code giving the width of the graphics area. (The height of the graphics area is defined by the height of the symbol font.) Then the READ code for the graphics is sent. The length of the READ code is defined by the width and height of the graphics area, so the transition back to symbol codes does not require a code.

In the CR/LF option, instead of transmitting a series of blank symbol codes at the right of the line, a special 8-bit code is used to designate the last-symbol-on-the-line. This, of course, would have to be to the right of any graphics on the line. This last-symbol-on-the-line code would direct the receiver to start on the next line of symbols, and would replace the CR and LF codes of TELETEX. For reasons of commonality it may be preferable to keep the two standard TELETEX symbols for this purpose.

This technique is considered primarily as a method to incorporate graphics (such as logos and signatures), into

computer-generated text. Therefore graphics areas are defined, probably by the user, as rectangular areas which may contain a mixture of graphics and symbols.

Since all lines of symbols must have proper spacing, symbols that are not aligned with the majority of the symbols must be treated as graphics.

2.4 Symbol Removal/Hybrid

This technique combines features of the other two symbol removal techniques to make it more robust than either in that it is designed to handle both isolated (or arbitrarily located symbols) and symbol strings in lines or segments.

In this technique, as in other symbol removal approaches, the symbols are detected, "removed", and their codes and positions are recorded. Spaces between symbols (up to 2) are filled with appropriate blank characters.

The presence or absence of a symbol code, rather than a READ code, is indicated by a single bit at the beginning of every scan line. In addition, a single bit preceding each symbol code indicates whether the symbol is contiguous or not, i.e., not followed by more than 2 blank spaces. If the symbol is not contiguous, it is preceded by a horizontal position code otherwise the symbol code follows immediately.

A special 8-bit symbol code terminates the symbol string at the end of the line of symbols. Then the READ code for that line is transmitted.

3.0 Task 2 - Measurement of Compression

3.1 Methodology for Measuring Compression

For each of four proposed mixed mode techniques, an estimate of compression has been made. Estimates of compression were made for CCITT test Document 1 and for two computer generated documents.

It should be recognized that compression values calculated in this report are estimates only, and should not be regarded as actual measured numbers. However, it is expected that the relative compressions of the various segmentation techniques are accurate, since the same assumptions were used for all of them.

3.2 Assumptions

In making compression estimates, the following assumptions were made:

- (1) Each symbol is encoded using 8 bits, which allows up to 256 different symbols.
- (2) Several of the 256 symbol codes can be made available for indicating termination of symbol transmission, or other requirements of the segmentation technique employed.

(3) A stored library, suitable for the document being transmitted, is available at both sending and receiving terminals.

(4) Bits required to identify the proper symbol library to the receiving terminal are neglected.

(5) The stored library will accommodate either fixed or proportionally spaced fonts, including several widths for word spaces.

(6) All characters of the principal font used in the document are in fact recognized as such, and will be encoded as symbols, subject to the rules of the proposed technique.

(7) Lines of symbols can be accommodated despite slight skews of the printed lines.

(8) The characters of the principal font include math symbols, italics, and Greek letters, but not subscripts or superscripts, or long horizontal or vertical lines.

(9) Graphic data is transmitted using the modified READ code, without EOL's and with $k=\infty$.

(10) The number of bits required to transmit increased width of white spaces by means of Modified READ can be neglected. This follows because the spacing between groups of black pels (such as symbols) usually only has

to be specified once, and the READ code length does not grow rapidly with the length of a white run.

(11) Each A4 source document normally 216 x 297MM, consists of 2,376 rows with 1,728 pels per row (ie, resolution equals 8 pels per mm, or approximately 200 pels per inch).

(12) Scanned documents, stored on tape, are retrieved onto disk as an image file commensurate with a 16-bit computer word size. Thus the computer image file consists of 2,336 rows of 1,728 bits each row. Since the computer image contains all the black pels of the original document, any process, such as the modified READ algorithm, performed on the computer image may represent the same process performed on the original document with negligible error or vice versa.

(13) Code transmissions will not experience any transmission errors, so addition of redundancy for error control is not required.

3.3 Calculating Compression

For each technique the number of bits required to construct the message is totaled. This includes any flags, tag bits, symbol codes, end of symbols on line, end of segment graphics, symbol mode changes and horizontal and vertical positions. The number of bits required for each of these functions is given in Table 3-1:

Table 3-1

Table of Bit Requirement for Data Function

<u>Data Function</u>	<u>Data Requirement (Bits)</u>
Scan Line Flag	1
Contiguous Symbol Flag	1
Symbol Code	8
End of Symbols on Line or Segment	8
Symbol to Graphics	
Vertical Position-Symbols on Line	12
Graphics to Symbols	
Horizontal Position	11

The compression is calculated by dividing the total message bits into the total number of pels, as referred to the source document, which is always $2,376 \times 1,728 = 4,105,728$.

In the three symbol removal techniques, the black pels associated with recognized symbols are coded and "removed" (changed to white), and then the entire residual document is encoded using the Modified READ code ($k=00$, no EOL code), including the areas formerly occupied by the "removed" symbols. In the Extended TELETEX technique, the Modified READ code is used only for areas that do not have encoded characters.

Values for document related parameters used in calculating compression estimates are given in Table 3-2.

**Document Related
Parameters used for Compression
Estimates**

	Document	
	Scanned CCITT-1	Computer Generated A B
No. of typewritten Symbols on Page	802	753 602
No. of typewritten Symbols on page including normal spaces between symbols (up to two).	934	900 724
No. of typewritten symbols on page including all spaces between symbols and spaces to indent both text and graphics.	1994	1165 1032
No. of lines of text on page	23	16 14
No. of segments or character strings more than 2 spaces apart	1	0 1
Scan lines per line of text (COLADD/11) /27 based on CCITT Document No. 1 and Appendix A	8.4	8.4 8.4
Scan Lines on which symbols detected, based on 8.4 x No. of lines of text	194	135 118
Maximum no. of lines per document, based on lines per inch spacing of text	70	70 70
Maximum no. of lines occupied by graphics, based on 6 lines per inch spacing of text	12	22 44

Table 3-2

Table of Document Related Parameters
used for Compression Estimates

3.4 Scanned Document - CCITT No. 1, Figure 3-1

3.4.1 Symbol Removal/Scan Line

Figure 3-2 illustrates the composition of a mixed-mode message using this technique. As indicated in Figure 3-3, all of the typewritten symbols are recognized, encoded and "removed". The presence or absence of a symbol code, rather than a READ code, is indicated by a single bit at the beginning of every scan line. A special 8 bit symbol code terminates the symbols on the scan line and indicates the end of symbols and start of graphics. Each symbol's horizontal position is independently encoded. Since the vertical position is implied by the scan line on which the symbol code appears, no account need be taken of spaces between words or between line segments. The Modified READ code ($k=0$, no EOL code) is applied to the residue (Figure 3-4) after removal of typewritten symbols. A summary of the compression estimate for the Symbol Removal/Scan Line Technique applied to the CCITT-1 document is presented in Table 3-3.

3.4.2 Symbol Removal/Line of Symbols

Figure 3-5 illustrates the composition of a mixed-mode message using this technique. As indicated in Figure 3-6 all typewritten symbols are recognized, encoded and "removed". The symbols are organized into lines of symbols with spaces between symbols (up to two) filled by blank characters. The resulting string of symbol codes is preceded by a graphics-to-symbols code, a horizontal-position-of-the-first-symbol code and is terminated

THE SLEREXE COMPANY LIMITED

SAPORS LANE · BOOLE · DORSET · BH 25 8 ER

TELEPHONE DIALS (945 13) 51617 · TELEFAX 123456

Our Ref. 350/PJC/EAC

18th January, 1972.

Dr. P.M. Cundall,
Mining Surveys Ltd.,
Holroyd Road,
Reading,
Berks.

Dear Pete,

Permit me to introduce you to the facility of facsimile transmission.

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogous electrical video signal. This signal is used to modulate a carrier, which is transmitted to a remote destination over a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

Probably you have uses for this facility in your organisation.

Yours sincerely,

Phil.

P.J. CROSS
Group Leader - Facsimile Research

Registered in England: No. 2828
Registered Office: 20 Victoria Lane, Epsom, Surrey.

Figure 3-1
CCITT Document Number 1

Figure 3-2

MESSAGE COMPOSITION

SYMBOL REMOVAL/SCAN LINE

Scan Line	SYM	PRES
201	0	G
202	0	G
203	0	G
204	0	G
205	1	S HPOS EOS G
206	0	G
207	1	S HPOS S HPOS S HPOS EOS G
208	0	G
209	0	G
210	0	G
211	1	S HPOS S HPOS EOS G
212	0	G

LEGEND

- SYM PRES 1 indicates at least one symbol on scan line - 1 bit
- G graphics mode using Modified READ code - variable bits
- S symbol code - 8 bits
- HPOS horizontal position of symbol - 11 bits
- EOS end of symbols on scan line - 8 bits

THE SLEREXE COMPANY LIMITED

SAPORS LANE . BOOLE . DORSET . BH 25 4 ER

TELEPHONE BOOLE (945 13) 31617 . TELEX 123486

Our Ref. 350/PJC/EAC

18th January, 1972.

Dr. P.M. Cundall,
Mining Surveys Ltd.,
Holroyd Road,
Reading,
Berks.

Dear Pace,

Permit me to introduce you to the facility of facsimile transmission.

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogue electrical video signal. This signal is used to modulate a carrier, which is transmitted to a remote destination over a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

Probably you have uses for this facility in your organisation.

Yours sincerely,

Phil.

P.J. CROSS
Group Leader - Facsimile Research

Registered in England: No. 2222
Registered Office: 20/21, 1, Ave. Road, Basingstoke.

Figure 3-3
Symbol Removal/Scan Line
CCITT Document Number 1

THE SLEREXE COMPANY LIMITED

SAPURS LANE . BOOLE . DORSET . BH25 8ER

TELEPHONE BOOLE (945 13) 51617 - TELEX 123456

Phil.

Registered in England: No. 2008
Registered Office: 40 Victoria Lane, Luton, Bucks.

Figure 3-4
Residue after removal of symbols
CCITT Document Number 1

Table 3-3

Summary of Compression Estimate

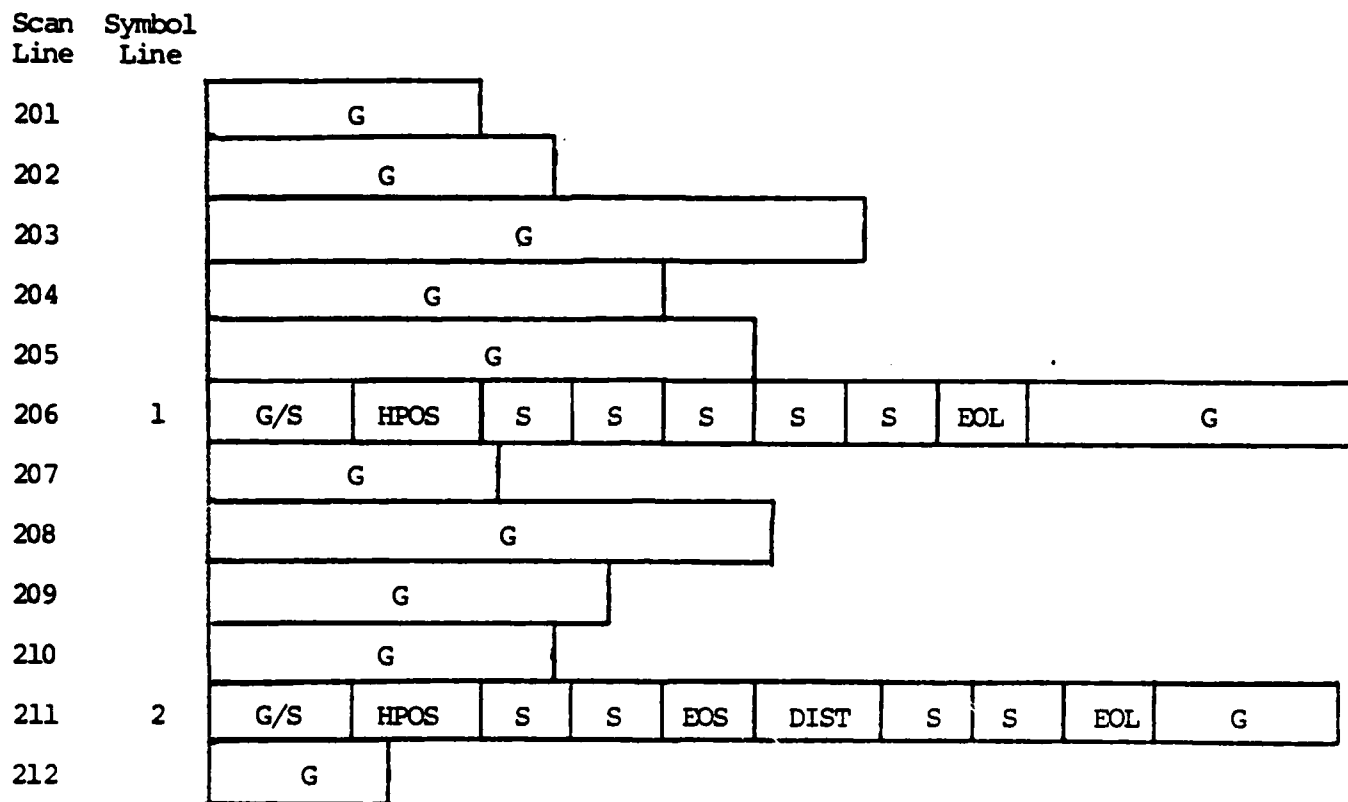
	Quantity	Bits
Symbol Codes (8 bits)	802	6,416
Symbol present on scan line (1 bit)	2,376	2,370
Symbol Horizontal Position (11 bits)	802	8,822
End of Symbol-Start of Graphics Code (8 bits)	194	1,552
Residue Encoded using Modified READ Code		<u>28,331</u>
		47,497

$$\text{Compression} = \frac{2,376 \times 1,728}{47,497} = 86.4$$

Symbol Removal/Scan Line

Figure 3-5

MESSAGE COMPOSITION
SYMBOL REMOVAL/LINE OF SYMBOLS



LEGEND

- G graphics mode using Modified READ code - variable bits
- S symbol code - 8 bits
- G/S indicates change from graphics to symbols - 12 bits
- HPOS horizontal position of first symbol in line - 11 bits
- EOL end of symbols on line - 8 bits
- EOS end of symbols on segment - 8 bits
- DIST distance between segments - 11 bits

THE SLEREXE COMPANY LIMITED

SAPORS LANE . BOOLE . DORSET . BH25 5ER

TELEPHONE BOOLE (945 13) 51417 - TELE 123466

Our Ref. 250/PJC/EAC

16th January, 1972.

Dr. P.M. Cundall,
Mining Surveys Ltd.,
Holroyd Road,
Reading,
Berkshire.

Dear Pete,

Permit me to introduce you to the facility of facsimile transmission.

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogous electrical video signal. This signal is used to modulate a carrier, which is transmitted to a remote destination over a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

Probably you have uses for this facility in your organisation.

Yours sincerely,

Phil.

P.J. CROSS
Group Leader - Facsimile Research

Registered in England: No. 2828
Registered Office: 20 Victoria Lane, Dover, Kent.

Figure 3-6
Symbol Removal/Line of Symbols
CCITT Document Number 1

with an end-of- symbols-on-line code . More than two spaces between symbols on a line breaks the line into segments which except for the last segment are each followed by an end-of-symbols-on-segment code and a distance- between-segments code. The last segment in a line is terminated with an end-of-symbols-on-line code. The Modified READ code (k=0, no EOL code) is applied to the residue Figure 3-4. A summary of the compression estimate for the Symbol Removal/Line of Symbols Technique applied to the CCITT-1 document is presented in Table 3-4.

3.4.3 Extended TELETEX - CR/LF Option

Figure 3-7 illustrates the composition of a mixed-mode message using this technique. Figure 3-8 shows symbols encoded by the Extended TELETEX technique (with transmission of CR/LF symbol) and also shows boxed-in areas of Graphics transmitted by the Modified Read code (k=0, no EOL code). All typewritten symbols are encoded as are blank characters used to space over: to the first character on each line of text and to the left hand horizontal position of the boxed-in areas of Graphics (see figures 3-9 through 3-14). Symbols-to-Graphics and Graphics Width codes guide deployment of the Modified Read Code. A summary of the compression estimate for the Extended TELETEX-CR/LF Option technique applied to the CCITT-1 document is presented in Table 3-5.

Table 3-4

Summary of Compression Estimate

	Quantity	Bits
Symbol + Blank Space Codes (8 bits)	934	7,472
Symbols on Line-Graphics-to-Symbols Code (12 bits)	23	276
End of Symbols on segment code (8 bits)	1	8
Distance between segments Code (11 bits)	1	11
End of Symbols on Line Code (8 bits)	23	184
Residue Encoded using Modified READ code		<u>28,331</u>
		36,535

$$\text{Compression} = \frac{2,376 \times 1,728}{36535} = 112.4$$

Symbol Removal/Line of Symbols

Figure 3-7

MESSAGE COMPOSITION

EXTENDED TELETEX

Symbol
Line

5	S	S	S	CR/LF															
6	S	S	S	S	S	S	S	S	S	S	CR/LF								
7	S	S	S	S/G	Width	G				CR/LF									
8	S	S	S	S	S	S	S	CR/LF											
9	S/G	Width	G				S	S	S	CR/LF									
10	S	S	S	S	S	S	S	S	CR/LF										
11	S	S	S	S	S	S	S	CR/LF											
12	S/G	Width	G		S	S/G	Width	G		S	CR/LF								
13	S	S	S	S	CR/LF														

LEGEND

- G graphics mode using Modified READ code - variable bits
- S symbol code - 8 bits
- CR/LF carriage return/line feed - 8 bits
- S/G indicates change from symbols to graphics - 8 bits
- Width width of graphics - 11 bits

THE SLEREXE COMPANY LIMITED

SAPORS LANE . BOOLE . DORSET . BH 25 1ER

TELEPHONE DIALS (945 13) 51617 . TELEX 123456

Our Ref. 350/PJC/EAC

18th January, 1972.

Dr. P.N. Cundell,
Mining Surveys Ltd.,
Holroyd Road,
Reading,
Berks.

Dear Pete,

Permit me to introduce you to the facility of facsimile transmission.

In facsimile a photocell is caused to perform a raster scan over the subject copy. The variations of print density on the document cause the photocell to generate an analogous electrical video signal. This signal is used to modulate a carrier, which is transmitted to a remote destination over a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video signal, which is used to modulate the density of print produced by a printing device. This device is scanning in a raster scan synchronised with that at the transmitting terminal. As a result, a facsimile copy of the subject document is produced.

Probably you have uses for this facility in your organisation.

Yours sincerely,

Phil.

P.J. CROSS
Group Leader - Facsimile Research

Registered in England: No. 2888
Registered Office: 68 Vauxhall Lane, London, E.C.4A

Figure 3-8
Extended TELETEX-CR/LF Option
CCITT Document Number 1

12:28 PM MON.. 21 NOV.. 1983
PLOT <LOGO > STARTING AT PEL #

1 (APPROX.) - RECORD LENGTH 88



LINES READ = 88.

Figure 3-9
Logo Graphic
CCITT Document Number 1

12:27 PM MON., 21 NOV., 1983
PLOT <SLRX > STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 848

THE SLEREXE COMPANY LIMITED

LINES READ = 44.

Figure 3-10

SLEREXE Graphic

CCITT Document Number 1

12:26 PM MON., 21 NOV., 1983
PLOT <SAPR > STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 624

SAPORS LANE - BOOLE - DORSET - BH 25 8 ER

LINES READ = 38.

Figure 3-11
SAPORS LANE Graphic
CCITT Document Number 1

12:25 PM MON., 21 NOV., 1983
PLOT <PHON > STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 544
TELEPHONE BOOLE (945 13) 51617 - TELEX 123456

LINES READ = 30.

Figure 3-12
TELEPHONE Graphic
CCITT Document Number 1

12:38 PM MON., 21 NOV., 1983
PLOT <SIGN > STARTING AT PEL #

1 (APPROX.) - RECORD LENGTH 192



LINES READ = 88.

Figure 3-13
Signature Graphic
CCITT Document Number 1

12:12 PM MON., 21 NOV., 1983
PLOT <REGIS > STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 568

Registered in England: No. 2038
Registered Office: 80 Vicars Lane, Ilford, Essex.

LINES READ = 43.

Figure 3-14
Registration Graphic
CCITT Document Number 1

Table 3-5
 Summary of Compression Estimate

	Quantity	Bits
Symbols + Blank Codes (8 bits)	1,994	15,952
Symbols to Graphics Codes (8 bits)	12	96
Graphics Width Codes (11 bits)	12	132
CR/LF Codes (8 bits)	70	560
Boxed-in Graphics Encoded using Modified READ Code		<u>25,026</u>
		41,766

$$\text{Compression} = \frac{2,376 \times 1,728}{41,766} = 98.3$$

Extended TELETEX - CR/LF Option

3.4.4 Symbol Removal/Hybrid

Figure 3-15 illustrates the composition of a mixed-mode message using this technique. As shown in Figure 3-16 all typewritten symbols, including spaces between symbols (up to two), are recognized, encoded and "removed". The presence or absence of a symbol code, rather than a READ code, is indicated by a single bit at the beginning of every scan line. In addition a single bit preceding each symbol code indicates whether the symbol is contiguous or not. If the symbol is not contiguous it is preceded by a horizontal position code otherwise the symbol code follows immediately. A special 8 bit symbol code terminates the symbol string at the end of the line of symbols. The Modified READ code ($k=\infty$, no EOL code) is applied to the residue Figure 3-4. A summary of the compression estimate for the Symbol Removal/Hybrid technique applied to the CCITT-1 document is presented in Table 3-6.

3.5 Computer Generated Documents - A and B

Computer generated documents A and B are shown respectively in Figures 3-17 and 3-18. Compression estimates for the computer generated documents are calculated in the same fashion as previously described in Section 3.4 for the CCITT document number 1. In the Symbol Removal/Scan Line Technique symbols are removed as illustrated: Document A, Figure 3-19; Document B, Figure 3-21. After symbol removal READ Code is applied to the residue: Document A, Figure 3-20; Document B, Figure 3-22.

Figure 3-15
 Message Composition
 Symbol Removal/Hybrid

Scan Line Sym Pres

201	0	0	G									
202	0	0	G									
203	0	0	G									
204	0	0	G									
205	1	0	HPOS	S	1	S	1	S	1	EOS	G	
206	0	0	G									
207	1	0	HPOS	S	1	S	0	HPOS	S	1	EOS	G
208	0	0	G									
209	0	0	G									
210	0	0	G									
211	1	0	HPOS	S	1	S	1	EOS	G			
212	0	0	G									

Legend

Sym Pres 1 indicates at least one symbol on scan line - 1 bit
 G graphics mode using Modified READ code - variable bits
 S symbol code - 8 bits
 HPOS horizontal position of symbol - 11 bits
 EOS end of symbols on scan line - 8 bits

THE SLEREXE COMPANY LIMITED

SAPORS LAKE · BOOLE · DORSET · BH 25 4 ER

TELEPHONE BOOLE (945 13) 51617 · TELEX 123466

Our Ref. 350/PJC/2AC

18th January, 1972.

Dr. P.H. Cundall,
Mining Surveys Ltd.,
Molroyd Road,
Reading,
Bucks.

Dear Sirs,

Permit me to introduce you to the facility of facsimile
transmission.

In facsimile a photocell is caused to perform a raster scan over
the subject copy. The variations of print density on the document
cause the photocell to generate an analogous electrical video signal.
This signal is used to modulate a carrier, which is transmitted to a
remote destination over a radio or cable communications link.

At the remote terminal, demodulation reconstructs the video
signal, which is used to modulate the density of print produced by a
printing device. This device is scanning in a raster scan synchronised
with that at the transmitting terminal. As a result, a facsimile
copy of the subject document is produced.

Probably you have uses for this facility in your organisation.

Yours sincerely,

Phil.

P.J. CROSS
Group Leader - Facsimile Research

Registered in England: No. 2828
Registered Office: 60 Victoria Lane, Dover, Kent.

Figure 3-16
Symbol Removal/Hybrid
CCITT Document Number 1

Table 3-6

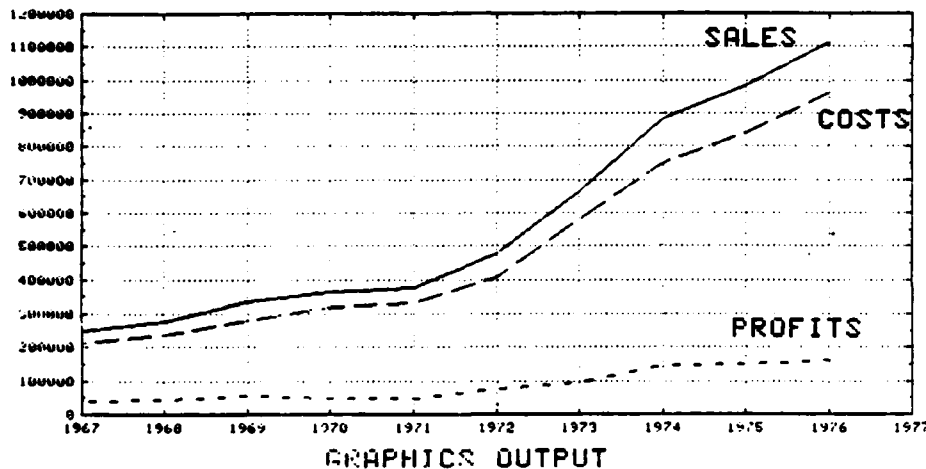
Summary of Compression Estimate

	Quantity	Bits
Symbol Codes + Blank (8 bits)	934	7,472
Symbol present on scan line (1 bit)	2,376	2,376
Contiguous Symbol or not Code (1 bit)	934	934
Symbol String Horizontal Position Code (11 bits)	24	264
End of Symbol-Start of Graphics Code (8 bits)	23	184
Residue Encoded using Modified READ Code		28,331
		<hr/> 39,561

$$\text{Compression} = \frac{2,376 \times 1,728}{39561} = 103.8$$

Symbol Removal/Hybrid

Income is an inflow of assets, but it must be recognized that there are inflows of assets which are not income. Obviously an inflow of capital funds from stockholders is not income to a corporation, nor should a business regard as income an inflow of assets which is offset by an increase in liabilities. Income consists of an inflow of assets in the form of cash receivables, or other property from customers and clients, and is related to



the disposal of goods and the rendering of services. If income is earned by selling goods, it may also be called profit; the term profit is not properly applied to income derived from the rendering of services.

A basic criterion for the determination of the period in which income may be regarded as earned may be stated as follows: Income should not be regarded as earned until an asset increment has been realized, or until its realization is reasonably assured.

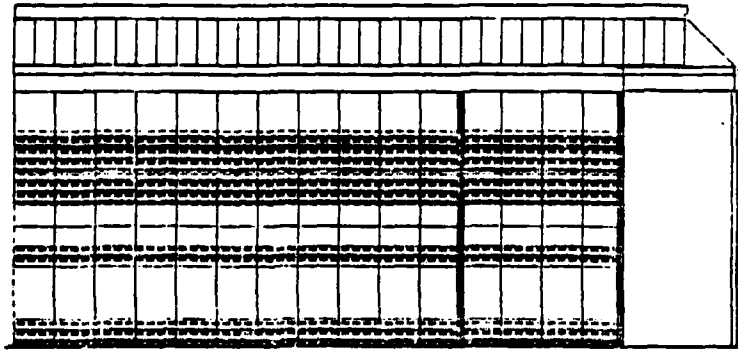


FIGURE 2 EXTERIOR VIEW OF PROPOSED BUILDING

We are running out of oil - and natural gas. Whether it's exactly 30 years or more makes very little difference in the long run. As we begin to drill more deeply into hard-to-reach reserves, the supply will become more spotty and more expensive. So start planning for oil-gas alternatives.

The best is coal. It's conservatively estimated that we have 300 years of coal reserves. However, the cost of mining and transporting it will grow sharply as demand builds. (Much of the coal will be difficult to reach, too.)

Should a company convert its boilers to coal-burning from oil or natural gas-burning? In many cases the answer is yes.

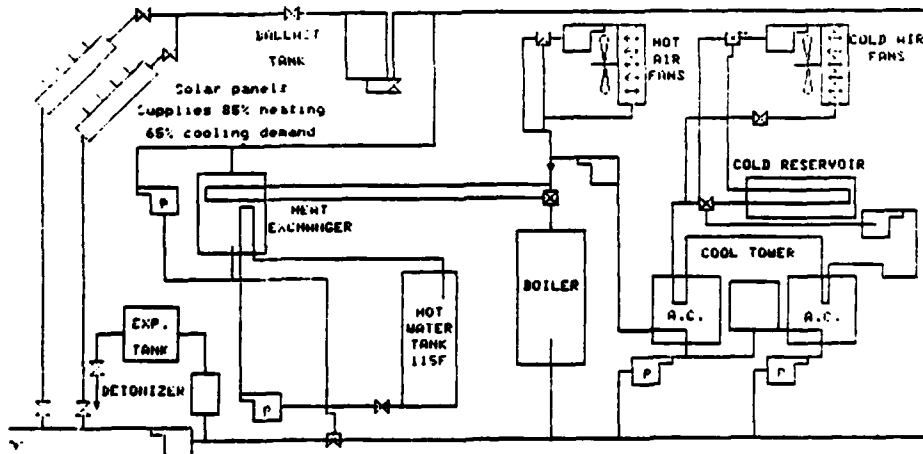
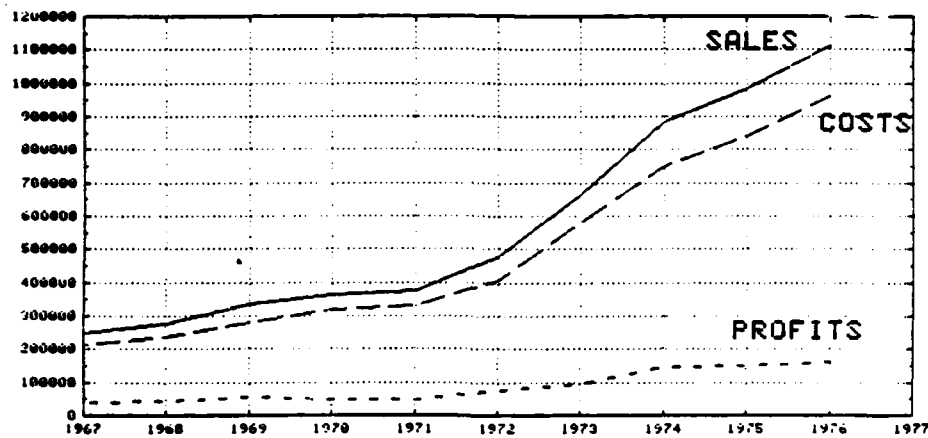


FIGURE 3 PIPR LAYOUT DIAGRAM

Income is an inflow of assets, but it must be recognized that there are inflows of assets which are not income. Obviously an inflow of capital funds from stockholders is not income to a corporation, nor should a business regard as income an inflow of assets which is offset by an increase in liabilities. Income consists of an inflow of assets in the form of cash receivables, or other property from customers and clients, and is related to



GRAPHICS OUTPUT

the disposal of goods and the rendering of services. If income is earned by selling goods, it may also be called profit; the term profit is not properly applied to income derived from the rendering of services.

A basic criterion for the determination of the period in which income may be regarded as earned may be stated as follows: Income should not be regarded as earned until an asset increment has been realized, or until its realization is reasonably assured.

Page 1

Figure 3-19
Symbol Removal/Scan Line
Document A

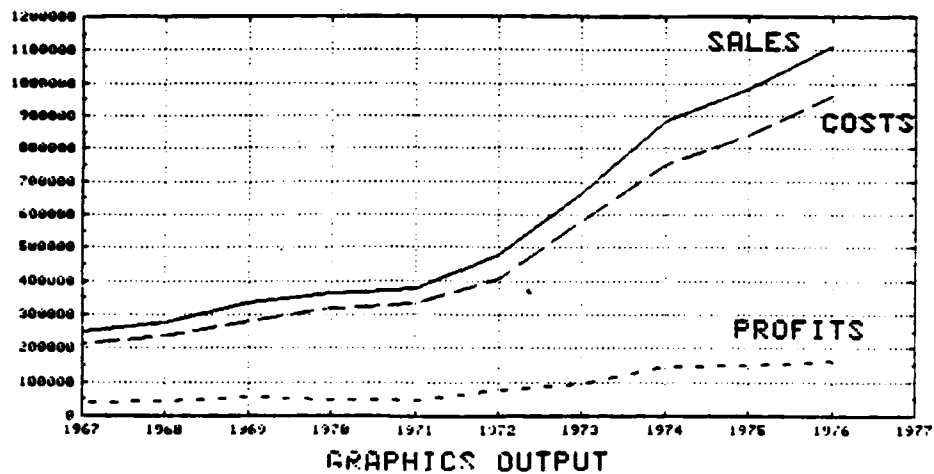


Figure 3-20

Residue after removal of symbols
Document A

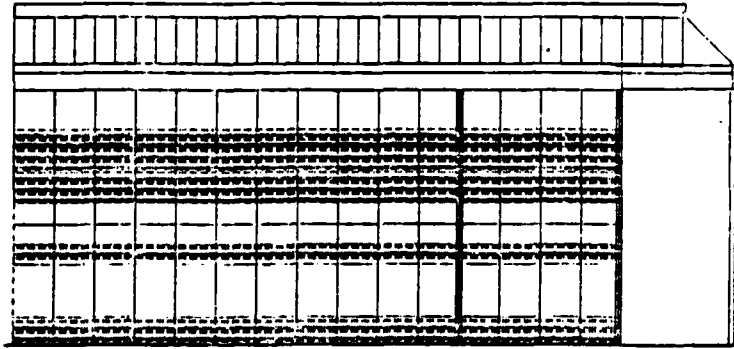


FIGURE 2 EXTERIOR VIEW OF PROPOSED BUILDING

We are running out of oil - and natural gas. Whether it's exactly 30 years or more makes very little difference in the long run. As we begin to drill more deeply into hard-to-reach reserves, the supply will become more spotty and more expensive. So start planning for oil-gas alternatives.

The best is coal. It's conservatively estimated that we have 300 years of coal reserves. However, the cost of mining and transporting it will grow sharply as demand builds. (Much of the coal will be difficult to reach, too.)

Should a company convert its boilers to coal-burning from oil or natural gas-burning? In many cases the answer is yes.

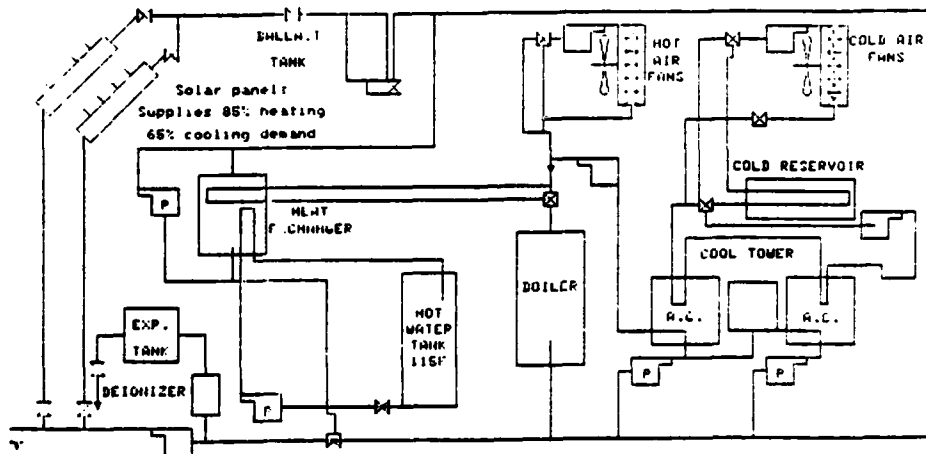


FIGURE 3 PIPR LAYOUT DIAGRAM

Page 2

Figure 3-21
Symbol Removal/Scan Line
Document B

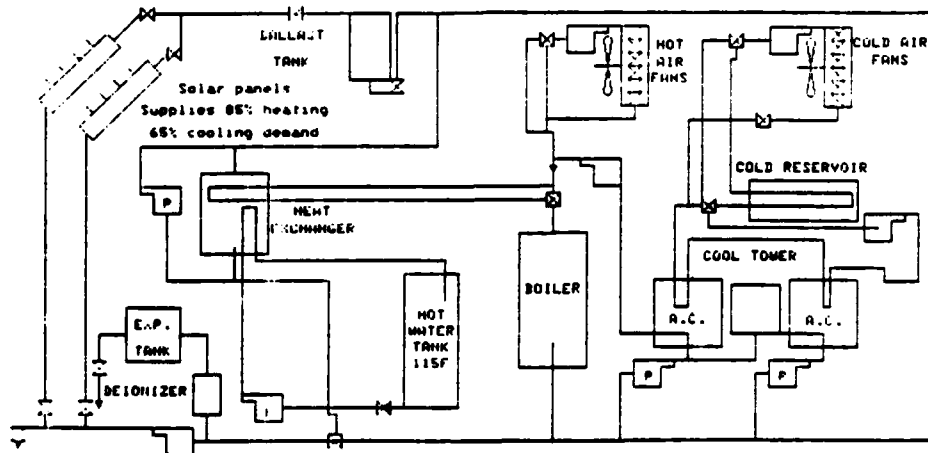
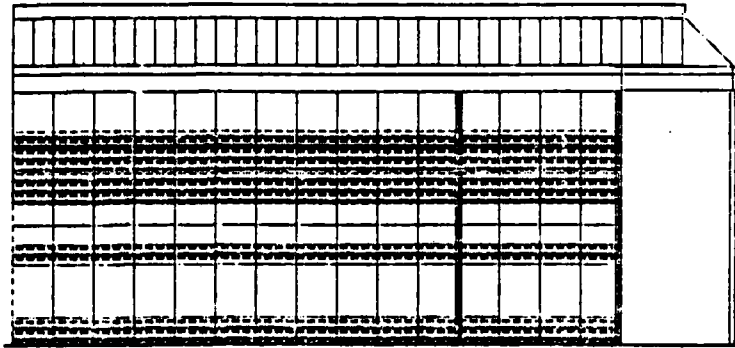


Figure 3-22
Residue after removal of symbols
Document B

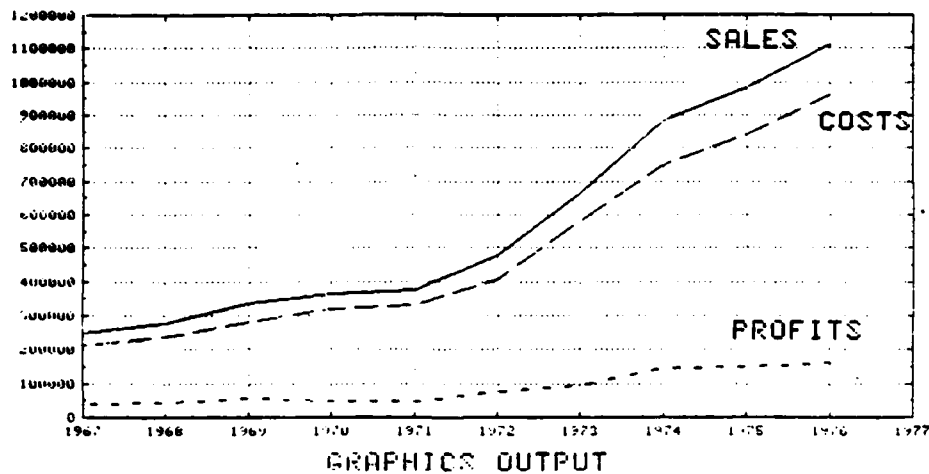
Similarly in the Symbol Removal/Line of Symbols Technique, symbols are removed as illustrated: Document A, Figure 3-23; Document B, Figure 3-24.

The Extended TELETEX-CR/LF Option Technique is applied as illustrated: Document A, Figure 3-25; Document B, Figure 3-27. READ code is applied to the boxed-in graphics: Document A, Figure 3-26; Document B, Figure 3-28 and 3-29.

In the Symbol Removal/Hybrid Technique symbols are removed as illustrated; Document A, Figure 3-30; Document B, Figure 3-31.

The results for the four mixed mode compression techniques are presented in Tables 3-7 to 3-10 for computer generated documents A and B.

Income is an inflow of assets, but it must be recognized that there are inflows of assets which are not income. Obviously an inflow of capital funds from stockholders is not income to a corporation, nor should a business regard as income an inflow of assets which is offset by an increase in liabilities. Income consists of an inflow of assets in the form of cash receivables, or other property from customers and clients, and is related to



the disposal of goods and the rendering of services. If income is earned by selling goods, it may also be called profit; the term profit is not properly applied to income derived from the rendering of services.

A basic criterion for the determination of the period in which income may be regarded as earned may be stated as follows: Income should not be regarded as earned until an asset increment has been realized, or until its realization is reasonably assured.

Page 1

Figure 3-23
Symbol Removal/Line of Symbols
Document A

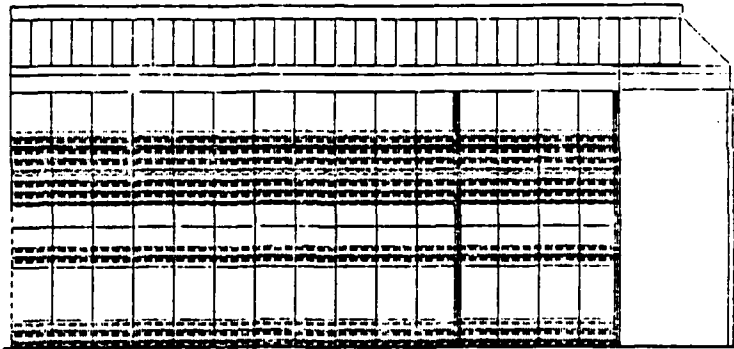


FIGURE 2 EXTERIOR VIEW OF PROPOSED BUILDING

We are running out of oil - and natural gas. Whether it's exactly 30 years or more makes very little difference in the long run. As we begin to drill more deeply into hard-to-reach reserves, the supply will become more spotty and more expensive. So start planning for oil-gas alternatives.

The best is coal. It's conservatively estimated that we have 300 years of coal reserves. However, the cost of mining and transporting it will grow sharply as demand builds. (Much of the coal will be difficult to reach, too.)

Should a company convert its boilers to coal-burning from oil or natural gas-burning? In many cases the answer is yes.

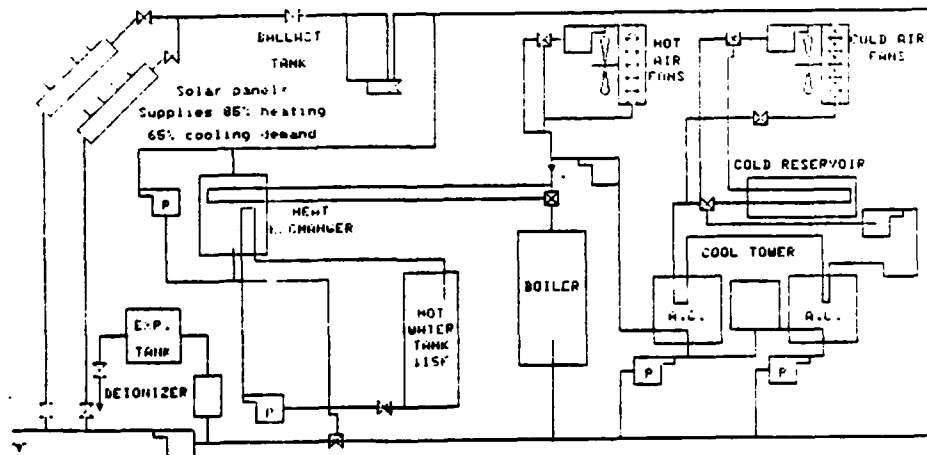
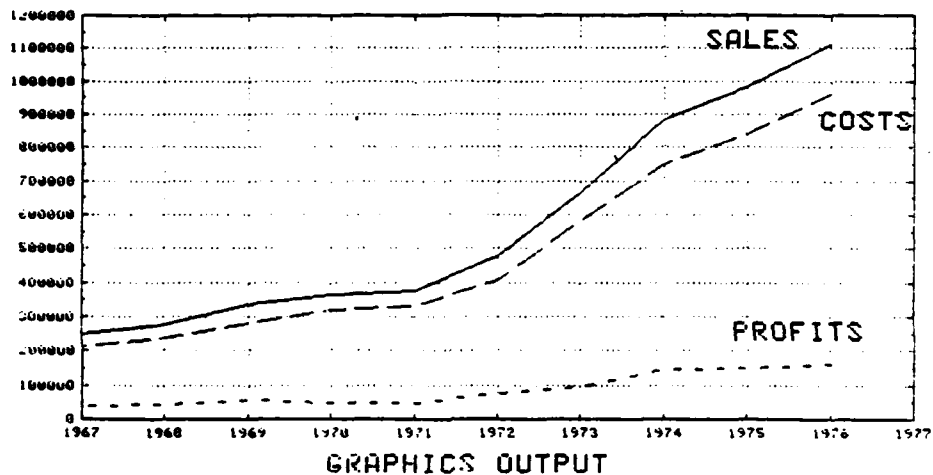


FIGURE 3 PIPR LAYOUT DIAGRAM

Income is an inflow of assets, but it must be recognized that there are inflows of assets which are not income. Obviously an inflow of capital funds from stockholders is not income to a corporation, nor should a business regard as income an inflow of assets which is offset by an increase in liabilities. Income consists of an inflow of assets in the form of cash receivables, or other property from customers and clients, and is related to



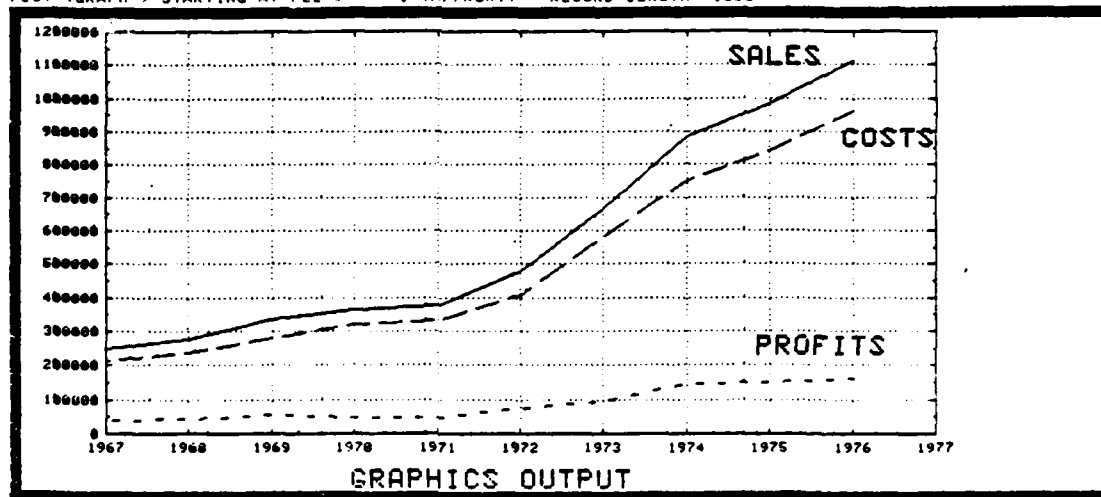
the disposal of goods and the rendering of services. If income is earned by selling goods, it may also be called profit; the term profit is not properly applied to income derived from the rendering of services.

A basic criterion for the determination of the period in which income may be regarded as earned may be stated as follows: Income should not be regarded as earned until an asset increment has been realized, or until its realization is reasonably assured.

Page 1

Figure 3-25
Extended TELETEX - CR/LF Option
Document A

12:53 PM FRI., 21 OCT., 1983
PLOT <GRAPH > STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 1688



LINES PEAD = 714.

Figure 3-26
GRAPH Graphic
Document A

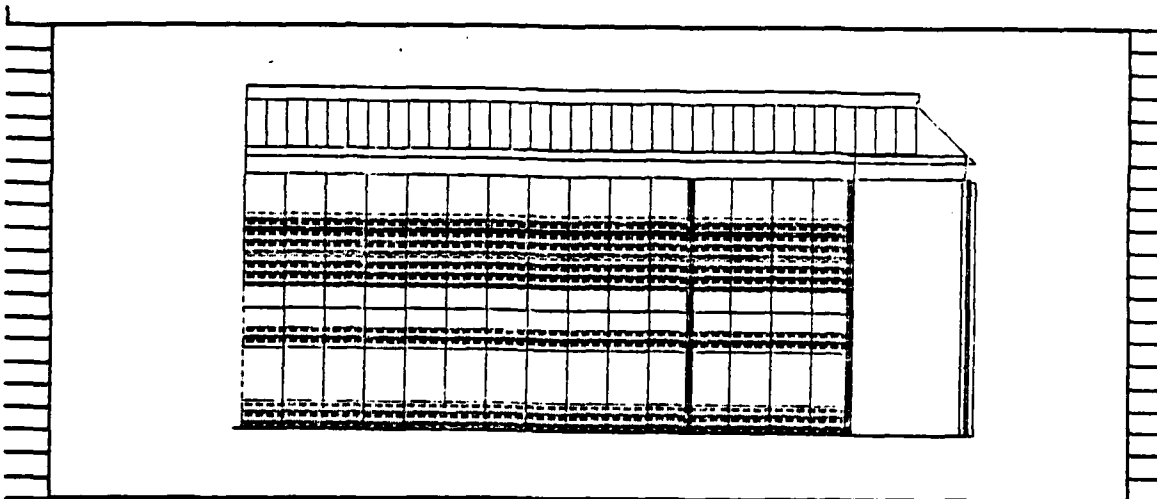


FIGURE 2 EXTERIOR VIEW OF PROPOSED BUILDING

We are running out of oil - and natural gas. Whether it's exactly 30 years or more makes very little difference in the long run. As we begin to drill more deeply into hard-to-reach reserves, the supply will become more spotty and more expensive. So start planning for oil-gas alternatives.

The best is coal. It's conservatively estimated that we have 300 years of coal reserves. However, the cost of mining and transporting it will grow sharply as demand builds. (Much of the coal will be difficult to reach, too.)

Should a company convert its boilers to coal-burning from oil or natural gas-burning? In many cases the answer is yes.

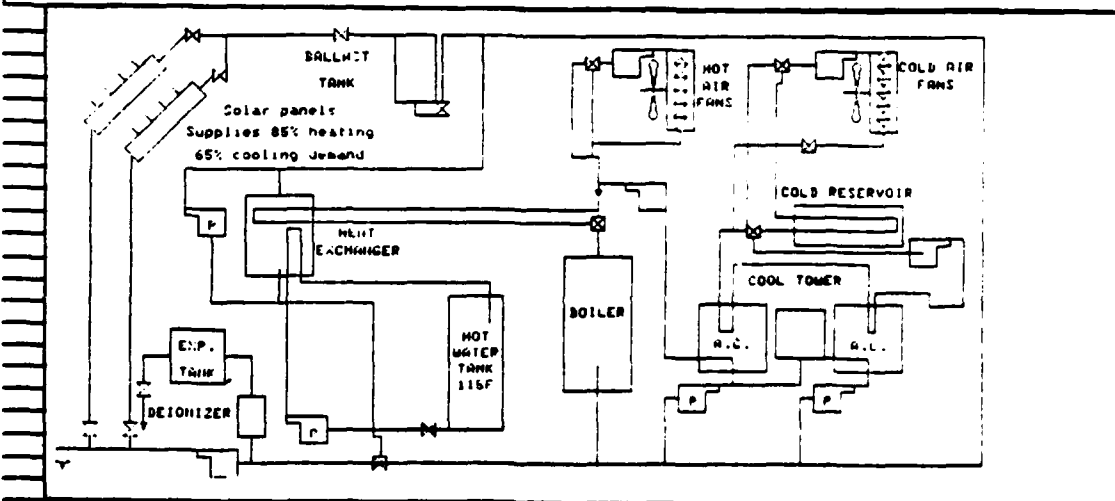
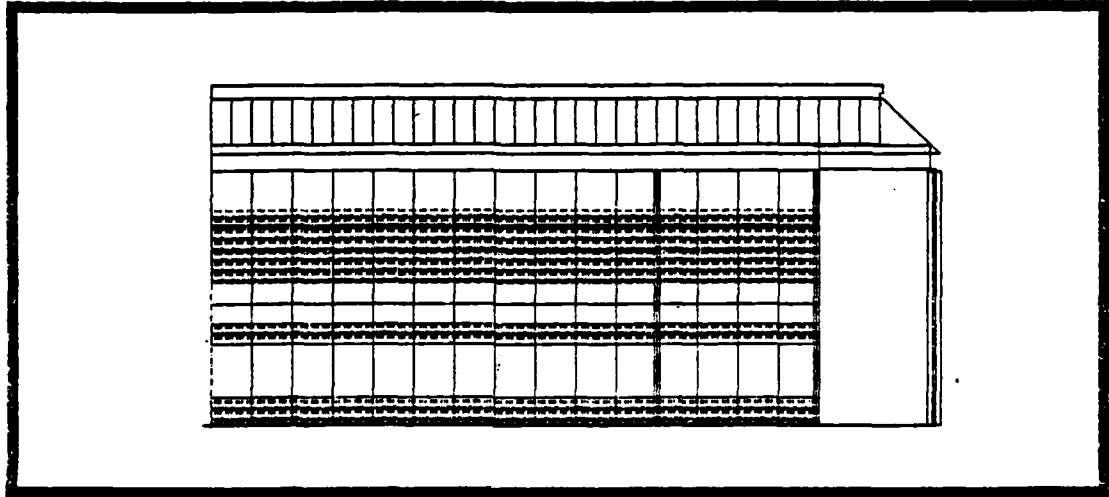


FIGURE 3 PIPR LAYOUT DIAGRAM

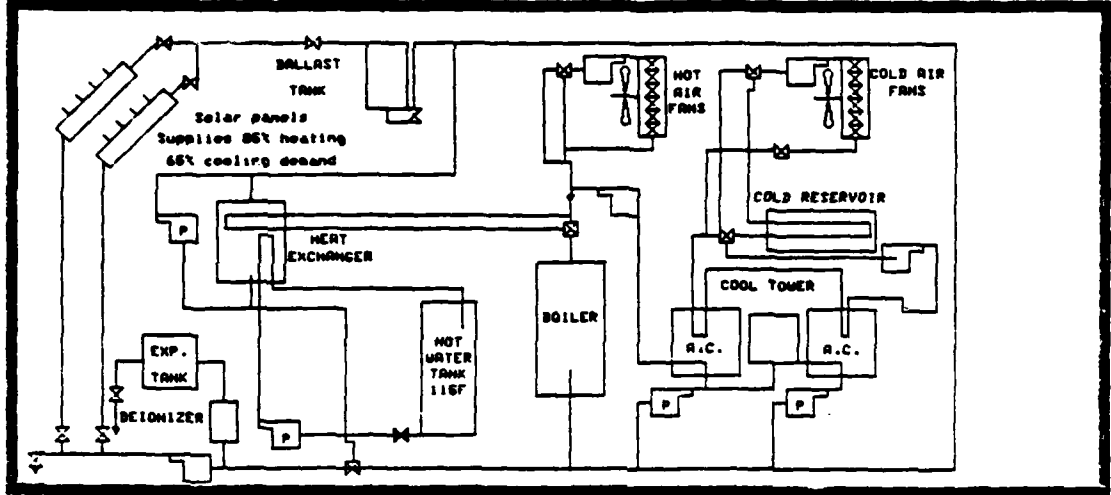
12:54 PM FRI., 21 OCT., 1983
PLOT <BUILD > STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 1688



LINES READ = 714.

Figure 3-28
Building Graphic
Document B

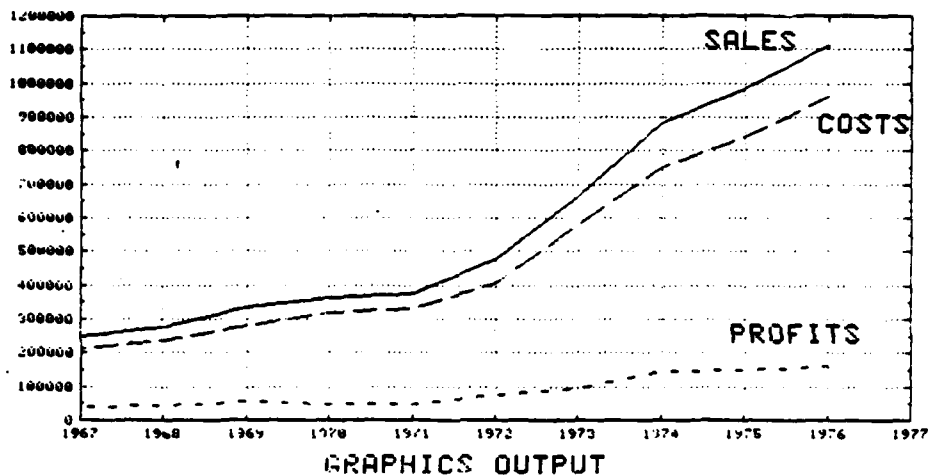
12:54 PM FRI., 21 OCT., 1983
PLOT <PIPED> STARTING AT PEL # 1 (APPROX.) - RECORD LENGTH 1688



LINES READ = 714.

Figure 3-29
Piping Graphic
Document B

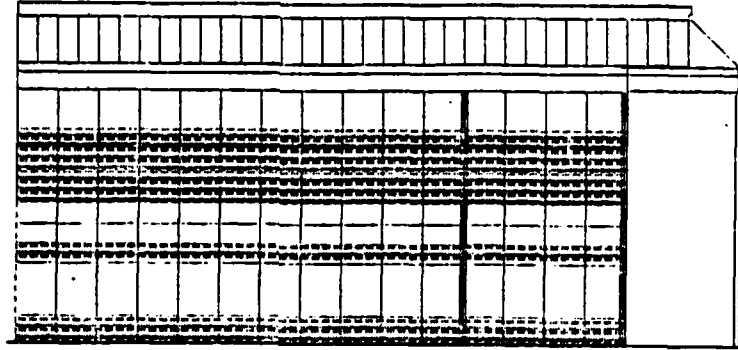
[Income is an inflow of assets, but it must be recognized that there are inflows of assets which are not income. Obviously, an inflow of capital funds from stockholders is not income to a corporation, nor should a business regard as income an inflow of assets which is offset by an increase in liabilities. Income consists of an inflow of assets in the form of cash receivables, or other property from customers and clients, and is related to]



[the disposal of goods and the rendering of services. If income is earned by selling goods, it may also be called profit; the term profit is not properly applied to income derived from the rendering of services.]

[A basic criterion for the determination of the period in which income may be regarded as earned may be stated as follows: Income should not be regarded as earned until an asset increment has been realized, or until its realization is reasonably assured.]

Figure 3-30
Symbol Removal/Hybrid
Page 1
Document A

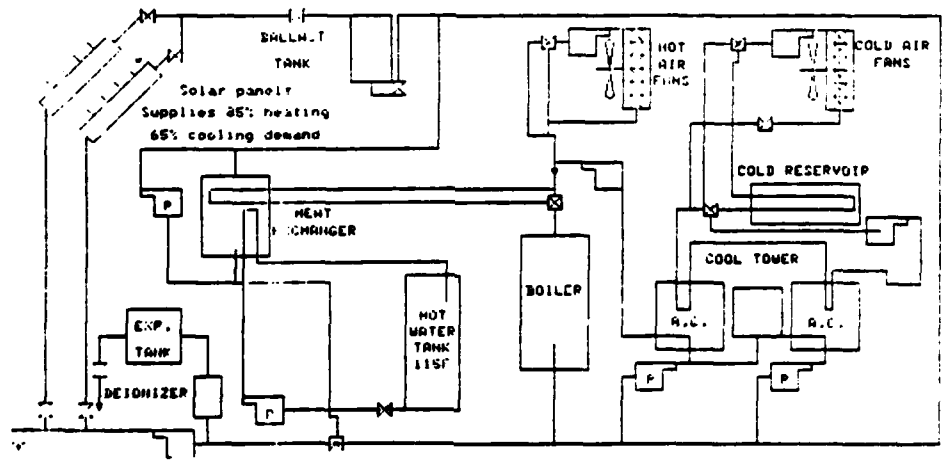


[FIGURE 2 - EXTERIOR VIEW OF PROPOSED BUILDING]

We are running out of oil and natural gas. Whether it's exactly 30 years or more makes very little difference in the long run. As we begin to drill more deeply into hard-to-reach reserves, the supply will become more spotty and more expensive. So start planning for oil-gas alternatives.

The best is coal. It's conservatively estimated that we have 100 years of coal reserves. However, the cost of mining and transporting it will grow sharply as demand builds. (Much of the coal will be difficult to reach, too.)

Should a company convert its boilers to coal-burning from oil or natural gas-burning? In many cases the answer is yes.



[FIGURE 3 - PIPER LAYOUT DIAGRAM]

Table 3-7

Summary of Compression Estimate

	Document A Quantity	Bits	Document B Quantity	Bits
Symbol Codes (8 bits)	753	6,024	602	4,816
Symbol present on scan line (1 bit)	2376	2,376	2,376	2,376
Symbol horizontal position (1 bits)	753	8,283	602	6,622
End of Symbol - Start of Graphics (8 bits)	135	1,080	118	944
Residue Encoded using Modified READ Code		<u>65,938</u>		<u>158,410</u>
		83,701		173,228
Compression = $\frac{2,376 \times 1,728}{\text{Total Bits}}$		49.0		23.7

Symbol Removal/Scan Line

Table 3-8

Summary of Compression Estimate

	Document A Quantity	Bits	Document B Quantity	Bits
Symbol + Blank Space Codes (8 bits)	900	7,200	724	5,792
Symbols on line - Graphics to Symbols Code (12 bits)	16	192	14	168
Horizontal Position of 1st Symbol Code (11 bits)	16	176	14	154
End of Symbols on Segment Code (8 bits)	0	0	1	8
Distance between Segments Code (11 bits)	0	0	1	11
End of Symbols on Line Code (8 bits)	16	128	14	112
Residue Encoded using Modified READ Code		<u>65,936</u>		<u>158,410</u>
		73,634		164,715
Compression = $\frac{2,376 \times 1,728}{\text{Total Bits}}$		55.8		24.9

Symbol Removal/Line of Symbols

Table 3-9

Summary of Compression Estimate

	Document A Quantity	Bits	Document B Quantity	Bits
Symbols + Blank Codes (8 bits)	1,165	9,320	1,032	8,256
Symbols to Graphics Codes (8 bits)	22	176	44	352
Graphics Width Codes (11 bits)	22	242	44	484
CR/LF Codes (8 bits)	70	560	70	560
Boxed-in Graphics Encoded using Modified READ Code		<u>64,272</u>		<u>157,514</u>
		74,540		167,166
Compression = $\frac{2,376 \times 1,728}{\text{Total Bits}}$		55.1		24.6

Extended TELETEX - CR/LF option

Table 3-10

Summary of Compression Estimate

	Document A Quantity	Bits	Document B Quantity	Bits
Symbol + Blank Codes (8 bits)	900	7,200	724	5,792
Symbols present on scan line (1 bit)	2,376	2,376	2,376	2,376
Contiguous Symbol or not Code (1 bit)	900	900	724	724
Symbol String Horizontal Position Code (11 bits)	16	176	14	154
End of Symbol - Start of Graphics Code (8 bits)	16	128	14	112
Residue Encoded using Modified READ Code		<u>65,938</u>		<u>158,410</u>
		76,718		167,628
Compression = $\frac{2,376 \times 1,728}{\text{Total Bits}}$		53.5		24.5

Symbol Removal/Hybrid

4.0 Task 3 - Analysis of Results

4.1 Compression

The compression estimates, calculated in Section 3 for all four segmentation techniques, are summarized in Table 4-1. For comparison, the compression for normal Group 4 Modified READ ($k=\infty$, no EOL) is also included in Table 4-1.

As expected, the results support the general conclusions that with increasing graphics relative to text content: the amount of compression decreases, the distinction between the four mixed-mode techniques diminishes as does their advantage over straight forward application of Modified READ ($k=\infty$, no EOL).

While the Symbol Removal/Line of Symbols technique provides the greatest compression of all, it is not robust with respect to arbitrary location of symbols. The Symbol Removal/Scan Line does handle the case of arbitrary symbol location but does not take advantage of the occurrence of lines of symbols. Furthermore the Symbol Removal/Scan Line technique provides the poorest compression performance. The Symbol Removal/Hybrid technique remains robust with respect to arbitrary symbol location and this technique also takes advantage of the occurrence of symbols in lines thereby providing compression close to that of the Symbol Removal/Line of Symbols technique.

The Extended TELETEX with CR/LF option provides compression intermediate between the Line of Symbols and Hybrid Symbol Removal

Summary of Compression Ratio Estimates

Compression Technique	Scanned Document				Computer Generated Document					
	CCITT No. 1				A		B			
	Rank	Comp. Ratio	%		Rank	Comp. Ratio	%	Rank	Comp. Ratio	%
Symbol Removal/Scan Line	4	86.4	76.87		4	49.0	87.81	4	23.7	95.18
Symbol Removal/Line of Symbols	1	112.4	100.00		1	55.8	100.00	1	24.9	100.00
Extended TELETEX - CR/LF	3	98.4	87.54		2	55.1	98.74	2	24.6	98.80
Symbol Removal/Hybrid	2	103.8	92.35		3	53.5	95.88	3	24.5	98.39
Modified READ (K=∞, no EOL)	5	27.9	24.82		5	18.0	32.26	5	13.6	54.62

Table 4-1

techniques, being marginally better than Symbol Removal/Hybrid for the more Graphics intensive computer generated documents and slightly inferior to Symbol Removal/Hybrid for the CCITT No. 1 document.

4.2 Complexity of Implementation

Not much separates the various techniques in complexity. In all cases accommodation of scanned documents requires symbol recognition. The three Symbol Removal techniques have the same image storage requirements independently of whether the recognized symbols are to be organized into lines of symbols or not. Generally the $32 \times 1728 = 55,296$ bits required to accommodate the larger fonts and hang-down characters will permit symbol recognition, removal and organization into lines.

In the Symbol Removal/Scan Line technique each recognized symbol is incorporated into the transmission as the recognition occurs. In the Symbol Removal/Line of Symbols and the Symbol Removal/Hybrid techniques provision must be made for calculating the linear regression of horizontal and vertical symbol positions to identify line skew and perform the vertical realignment, of the recognized symbols to a baseline, necessary to remove the skew.

Extended TELETEX, with CR/LF option, is presented as a technique for generating a mixed-mode message by a computer; not as a method for scanning a document and segmenting it into graphics and text.

4.3 Commonality

This section discusses the commonality or ability of a Mixed Mode machine to transmit messages to, or receive messages from such existing machines as:

- (1) TELETEX
- (2) Standard Group 4 FACSIMILE, without mixed-mode capabilities
- (3) Group 3 FACSIMILE

Changes to these machines are not considered permissible because they are already in the field; rather, here are considered changes to Group 4 mixed-mode machines necessary to provide commonality with existing machines.

The basic core of commonality between the existing machines and mixed-mode machines is built-in in that all mixed-mode techniques considered use the TELETEX code and the Modified READ II code proposed for Group 4 FACSIMILE machines. The Group 4 code modifies the Group 3 code by:

- (1) Using $k=\infty$ instead of $k=4$ for 7.7 lines/mm,
- (2) Deleting the EOL code for each line,
- (3) Eliminating bit stuffing to achieve minimum line time.

General compatibility between Group 4 and Group 3 machines, with respect to the encoding differences, is expected to be the rule rather than the exception. Therefore commonality with Group 3 machines is implicit in the discussion of commonality between standard and mixed-mode option Group 4 machines. No special

discussion of commonality with Group 3 machines is necessary.

For all of the mixed-mode techniques to achieve commonality it may be necessary to inhibit information about the stored library to be used. Other than this general requirement, the Symbol Removal/Line of Symbols technique requires merely the inhibiting of symbol recognitions to produce Group 4 transmissions while reception of Group 4 transmissions requires no modification whatsoever.

In addition to the above, for Symbol Removal/Scan Line and Symbol Removal/Hybrid, code bits that change mode must be deleted on transmission and inserted on reception. For both techniques this is a single bit that precedes each scan line.

For Extended TELETEX, a Group 4 transmission can be obtained by inhibiting all symbol recognitions, including blanks, which will force the entire line to be transmitted as graphics. In addition, the codes for the last symbol on the line and the graphics width would have to be deleted. For reception, the last symbol code and a graphics width code of 1728 would have to be added before each scan line to correct the Group 4 transmission for compatibility with the mixed-mode receiver.

For Extended TELETEX, to receive TELETEX transmissions no change is required except adding the code that identifies the stored library to use. In transmission, the graphics mode must be inhibited, with space symbol codes being transmitted whenever material that cannot be recognized as symbols is encountered. Also CR/LF codes must be inserted for each line.

For all techniques except extended TELETEx, in transmitting, the graphics mode must be inhibited, and a blank symbol used to replace each 20 pels of all-white or graphics pels. Also CR/LF codes must be inserted at the end of each line (approximately 33 scan lines). Corresponding changes must be made for reception, namely adding coding for approximately 33 all-white scan lines for each LF, and deleting the CR/LF codes.

In addition for Symbol Removal/Line of Symbols, the 12-bit (EOL) code that indicates a change from graphics to symbol mode must be deleted on transmission, and added on reception.

5.0 Conclusions And Recommendations

Figure 5-1 summarizes the subjective evaluations given to each mixed mode technique for each of the topics considered in the study. Note that there is a slight preference for the Symbol Removal/Line of Symbols technique relative to the other three algorithms.

This study assumed that the OCR function was performed perfectly for the scanned input document. AT&T has submitted a recommendation to the CCITT for a mixed mode system describing a specific approach to the OCR function. It is recommended that this algorithm be simulated in order to properly evaluate the impact of a realistic OCR system on mixed mode performance.

	Commonality		Complexity		Compression
	Group 3/4	TELETEX	Image Storage	Line Organization	
Symbol Removal/Scan Line	Good	Good	Good	Excellent	Fair
Symbol Removal/Line of Symbols	Excellent	Good	Good	Good	Excellent
Extended Teletex	Fair	Excellent	Good	Fair	Good
Symbol Removal/Hybrid	Good	Good	Good	Good	Good

Figure 5-1 Relative Evaluation of Alternative Mixed Mode Techniques

APPENDIX A
CCITT DRAFT RECOMMENDATION S.a
DOCUMENT INTERCHANGE PROTOCOL
FOR THE TELEMATIC SERVICES

SU 1/6/83

Temporary Document No. 84

CCITT
STUDY GROUP VIII
WORKING PARTIES VIII/2, 3 AND 4

Geneva, 24 May - 3 June 1983

Question :

SOURCE : SECOND DRAFT OF DRAFT RECOMMENDATION S.a

TITLE : DRAFTING GROUP ON DRAFT RECOMMENDATION S.a

Document Interchange Protocol
for the Telematic Services

Contents:

1. General

1.1 Scope

1.2 Fundamental principles

1.3 Definitions

2 Functions for the interchange of text in image form

2.1 Layout structure, objects and attributes

2.2 Layout descriptors

2.3 Text units

2.4 Relationships between descriptors and text units

3 Functions for the interchange of text
in processable form

4 Specification of protocol elements -2-

- 4.1 General
- 4.2 Layout descriptors
- 4.3 Text units
- 4.4 Logical descriptors

LJZ 830601-1

5) Coding

- 5.1 Principle of coding
- 5.2 Coding of protocol elements
- 5.3 Coding of parameters
- 5.4 Parameters values

6) Application rules

- 6.1 General
- 6.2 Fac simile application
- 6.3 Teletex mixed mode application

ANNEX A. Terms and definitions

ANNEX B. Document architecture and interchange formats.

ANNEX C. Examples of document description

Document Interchange Protocol

~~Presentation Control Procedure~~ for the Telematic Services.

1) General

1.1. Scope

1.1.1. Concerning the service aspects :

- Recommendation F.200 lays down the operation provision for the automatic international Teletex service. The service requirements unique to the mixed mode of operation are described in the annex C of the recommendation F.200.
- Recommendation F... should describe the service requirement for G4 fac simile service.

1.1.2. On the technical side :

1.121. The terminal equipment is defined by

- Recommendation S.60 for the teletex,
- Recommendation T.a for the G4 facsimile

1.122. Concerning the information coding :

- Recommendation S.61 defines the character repertoire and coded character set for the international teletex service,
- Recommendation T.b. defines the coding scheme used in G4 facsimile equipments,
- Recommendation S.100 defines the coding scheme used in videotex services.

1.123. Recommendation S.62 specifies the control procedure for the teletex and G4 facsimile services.

Note : the generalized session protocol, under discussion between CCITT and ISO, should be also considered.

1.124. Recommendation S.70 specifies the network independent basic transport for teletex and G4 facsimile services.

1.1.3

This Recommendation S.a defines the *Document Interchange Protocol* to be used within the Telematic Services when a document structure is required e.g. for Mixed Mode Teletex and for Group 4 Facsimile. *(for the time being)*

S.a is embedded in a Framework of Recommendations for Telematic Services *as shown in Figure 1.*

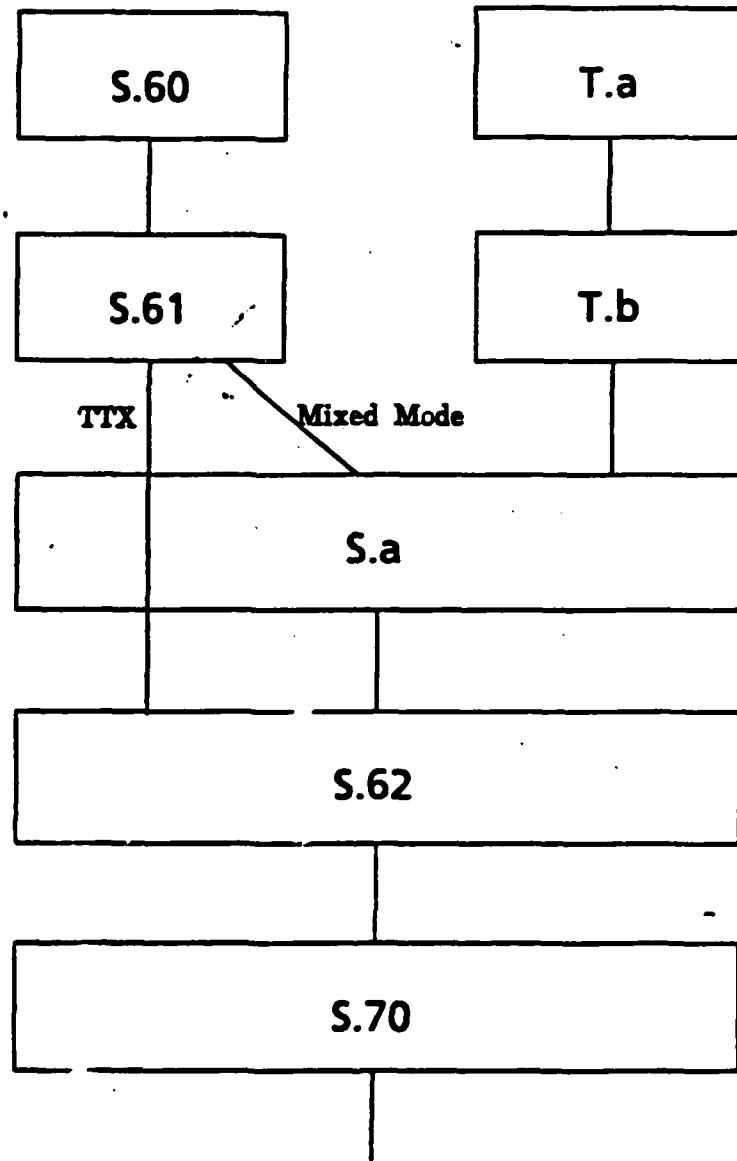


Figure 1. Framework of Recommendations for Telematic Terminals

1.2 Fundamental principles

1.2.1 Document architecture concept (see further details in Annex B)

1.2.1.1 For the purpose of this Recommendation, a document is an ~~amount~~ of text that is interchanged between telematic terminals.

1.2.1.2 A document can be interchanged for two major purposes.

- It may be interchanged as an original in a final form allowing for printing, displaying and ~~filing~~^{storing} at the recipient.
- It may be interchanged in a revisable form allowing for *processing* at the recipient.

Processing includes editing, reformatting, filing and other manipulations.

1.2.1.3 Text is information for human comprehension that can be presented in a two-dimensional form, e.g. printed on paper or displayed on a screen.

1.2.1.4 Text consists of graphic elements such as character box elements, geometric elements and photographic elements, which constitute the content of a document.

Content

1.2.1.5 The ~~parts~~ of a document need be separated into various portions in order to :

- delimit ~~the~~ presentation ~~units~~ ^{objects} (e.g. ~~pages~~) such as pages
- delimit logical objects such as paragraphs
- use different types of coding
- allow processing after communication.

The description of these portions of text and their relationship constitute the document architecture

1.2.1.6 The document architecture recognizes two structures

- the layout structure
- the logical structure

The layout structure relates the content of ^{portions} a document ^{layout objects} for their positioning and rendition on the presentation media.

The logical structure relates the content of ^{portions} a document to logical text objects such as portions, serving specific purposes, sections, headings, paragraphs, footnotes and figures.

The architecture that is particular to a given document is called specific document architecture.

1.2.17 A given document may contain ^(layout objects with common) predefined content portions representing logos, forms etc. which may appear several times in the document.

These predefined content portions and their relationship are described by the generic layout structure

In a similar manner there may be predefined logical objects and structuring rules which constitute the generic logical structure

The interchanged generic document structures help

- to improve the transmission efficiency,
- to maintain the consistency of the document with the document class during revision at the recipient and
- to facilitate the creation of new documents of a certain class.

Therefore a comprehensive description of a document may comprise specific and generic structures as shown in Figure 2.

In the case the logical structure is used there are associated layout directives allowing for the control of formatting or reformatting.

4.2.18 By the use of different components of the document architecture different interchange formats can be derived.

Two major types of formats are distinguished

- text image format
- text processable format.

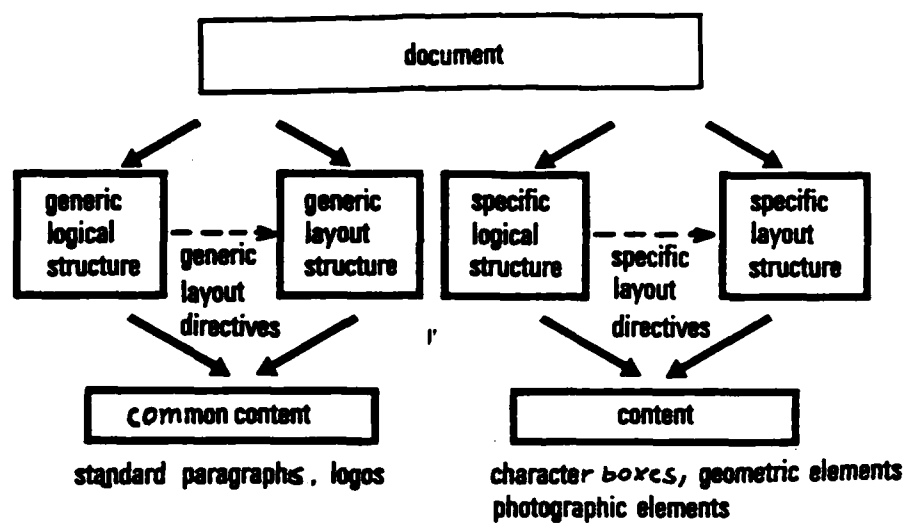


Figure 2. Document Architecture

1.2.2. Communication concepts.

- 1.221. The capabilities of the *document interchange* ~~presentation~~ protocol ~~defining such a Document Structure~~ are negotiated in the session establishment phase.

The terminal capabilities required in order to receive the document are globally indicated by the ~~"Presentation Level Protocol"~~ parameter.
document interchange

- 1.222. The necessary elements describing the *document* ~~Structure~~ are defined in this Recommendation S.a. and will be carried inside the the Session Service Data Units (CSUI-CDUI S.62 commands).

2 Functions for the interchange of text in image form

2.1 Layout structure, objects and attributes

The layout structure consists of the specific layout structure and, optionally, the generic layout structure.

2.1.1 Specific layout structure

The specific layout structure is a tree structure with a variable number of hierarchical levels. Its nodes are named specific layout objects. The actual number of levels depends on the given document.

The top level is the document. The layout objects below the level of document are named pages. The levels below that of page are optional. If there is any layout structure below the level of page, it consists of one or more levels of frame and, at the bottom level, blocks.

The page is the unit of presentation. It is a rectangular area with dimensions equal to the nominal paper size (after possible trimming) or equal to the scanning area (in case of facsimile). Paper size tolerances, skew, etc. are not considered in this concept of page.

A frame is a rectangular container within a page or within a frame of a hierarchically higher level, with its sides parallel to the sides of the page.

A block is a rectangular container within a frame with its sides parallel to the sides of the page.

If there is any layout structure below the level of page, there is at least one level of frame. ~~This~~
~~is~~ The highest level of frame is named image area and may be defined implicitly.

All blocks and all frames, if any, below the level of image area are positioned relatively to the next higher level of frame.

LJZ 830601-3

The content is divided into portions which correspond either to the pages or to the blocks.

Each such portion consists of graphic elements of a single category (character box elements or photographic elements) only.

Any number of blocks may be overlaid, partially or fully, independent of the category of graphic element. Thus, an image may consist of a photographic block overlaid by a character box block.

A block may overlay other blocks transparently or opaquely. This is specified by attributes of the block concerned.

2.1.2 Generic layout structure

The generic layout structure is optional. It consists a tree structure with a variable number of hierarchical levels. The nodes are named generic layout objects.

[to be continued]

LJZ 830601-4

2.1.3 Attributes

Attributes are parameters of the specific and generic layout objects that specify characteristics of and relationships between the objects.

Examples of attributes are:

- identification attributes, specifying the type of object (specific or generic page, block, etc.) and identifying the individual object;
- structure attributes, specifying the ~~value~~ hierarchical relationships between objects, ~~and~~ the correspondence between specific and generic objects, and the correspondence between layout objects and content portions;
- positioning attributes, specifying the dimensions and positions of the objects,
- presentation attributes, specifying in detail how the content ~~is to be changed~~ of the objects is to be changed, e.g. character spacing, line spacing, resolution.

The attributes of the specific and generic layout objects are defined in 2.1.3.2.

LJ2830601-5
Version 2

2.1.3.1 Default attributes

Certain attributes, viz. some of the positioning ~~and~~ presentation attributes, can be specified either explicitly for the object to which they apply directly, or at higher levels of the hierarchy. In the latter case, the attributes are interpreted as default values for the lower levels. They can be overridden by generic and specific attributes at the lower levels.

For example, it is possible to specify the default page size at document level, or the default resolution for photographic blocks at page level.

In addition, standard default values (to be used when no particular values are specified) are defined in this Recommendation.

To determine the attributes of a specific layout object, the priority order is:

1. Attributes specified explicitly for the object concerned override any defaults.
2. If the specific object concerned corresponds to a generic object, the generic attributes override any defaults specified in the specific structure.
3. Unless the specific object concerned is the document itself, the attributes of the object at the next higher level act as default values for the attributes of the object concerned.
4. If no attributes are specified explicitly even at document level, the default values defined in this Recommendations apply.

2.1.3.2 Attribute definitions

The attributes of the specific and generic layout objects are defined below. Some attributes apply only to certain types of objects. Where this is the case, it is mentioned in the definition. Otherwise, the attributes apply to all object types.

Object type

This attribute specifies whether the object concerned is a document, a page set, a page, a frame or a block, and whether it is a specific or a generic object.

Identifier

This attribute identifies the object uniquely and is used to refer to the object concerned.

Reference to corresponding generic object

This attribute applies to a specific object only, and only when it corresponds to a generic object. is the identifier of the corresponding generic object.
The value of this attribute

References to subordinate objects

This attribute applies to any object, unless it is at the lowest level of the hierarchy. It consists of a list of the identifiers of the relevant subordinate objects.

References to text units

This attribute applies only to an object at the lowest level of the hierarchy. It consists of the identifier or identifier(s) of the corresponding text unit or text units (see 2.3).

Reference to set of default values

This attribute refers to a set of default values (e.g. a combination of page size, image area, character spacing, line spacing, resolution, etc.) defined in this Recommendation.

Page size

This attribute applies to a page (and may be specified as a default at higher levels). It refers to a standard page size, e.g. ISO A4, defined in this Recommendation.

Position

This attribute applies to a frame or a block out. It consists of a pair of coordinates representing the position of the object relatively to the object at the next higher level in the hierarchy, i.e. the containing frame or page, in Basic Positioning Units (see - - -).

Dimensions

This attribute applies to a frame or a block out. It consists of a pair of dimensions representing the size of the object in Basic Positioning Units (see - - -).

Overlay characteristics

This attribute applies to a block (and may be specified as a default at higher levels). It identifies the objects that may be overlaid opaquely, and the objects that may be overlaid transparently, by the object concerned.

Presentation attributes

This is a group of attributes that apply to an object at the lowest level of the hierarchy (and may be specified as defaults at higher levels). They consist of:

- a) context type;
- b) character box attributes:
 - character repertoire
 - character path
 - character orientation
 - character spacing
 - line progression
 - line spacing
 - character box size
 - base line offset
 - graphic rendition (font, style, colour, etc.)
- c) photographic attributes:
 - pel path
 - line progression
 - resolution

2.2 Layout descriptors

-19-

A layout descriptor is an element of the Document Interchange Protocol that represents the attributes of a specific or generic layout object. Each specific or generic layout object is represented by one layout descriptor.

A layout descriptor is a data structure consisting of subordinate data structures and elementary data items. Each subordinate data structure again consists of subordinate data structures and/or elementary data items. The elementary data items are of a small number of basic data types such as numbers, character strings and bit strings.

LJ2 830601-6

2.3 Text units

A text unit is an element of the Document Interchange Protocol that represents a portion of document content corresponding to a specific or generic layout object.

A text unit consists of two main parts:

~~a) a data structure that represents some parameters related to the portion of document content, including an identifier (used to refer to the text unit) and~~

- a) a data structure representing parameters that identify the text unit, the type of document content (character box elements or photographic elements) and the method of coding the portion of document content;
- b) an information field representing the portion of document content concerned.

The elementary data items subordinate to a text unit are of a small number of basic data types such as numbers, character strings and bit strings.

2.4 Relationships between descriptors and text units

~~There are three types of relationships between descriptors, and between descriptors and text units.~~

The relationships between descriptors, and between descriptors and text units, are represented by pointers. A pointer is a data item, used within a descriptor, that uniquely identifies a descriptor or a text unit and refers to that descriptor or text unit.

There are three types of relationships between descriptors, and between descriptors and text units

a) Hierarchical relationships between objects.

These are represented by an index of pointers in the descriptor of a layout object, referring to the descriptors of the subordinate objects at the next hierarchical level. For example, a page descriptor refers to the descriptors of the highest-level frames (if any) or to the descriptors of the blocks within the page concerned.

- b) Correspondence between a specific object and a generic object.

If such a correspondence exists, it is represented by a pointer in the descriptor of the specific object concerned, referring to the descriptor of the corresponding generic object.

- c) Correspondence between objects and contents.

Portions of document content represented by text units correspond either to pages or to blocks. This correspondence is represented by a pointer in the descriptor of the page or the block concerned, referring to the corresponding text unit.

4.1 General

This section provides the detailed specification of the descriptors and text units which were introduced in Section 2 above.

The specification makes use of a formalized notation which describes the type of each component of the data structure. The notation is defined in —

NOTE: For the moment, a notation based on that of Draft Recommendation X.4154 is used.

For ease of comprehension, the notation is embedded in explanatory text.

AD-A137 592

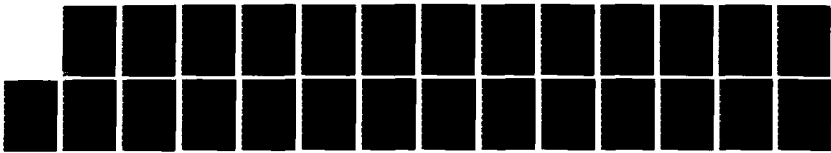
MIXED MODE FOR GROUP 4 FACSIMILE SYSTEMS(U) DELTA
INFORMATION SYSTEMS INC JENKINTOWN PA A DEUTERMANN
07 NOV 83 NCS-TIB-83-7 DCA100-82-C-0047

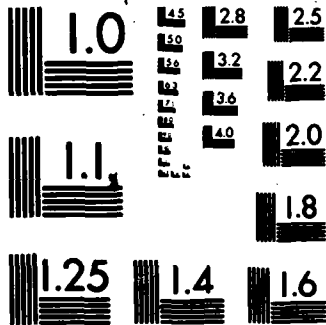
2/2

UNCLASSIFIED

F/G 14/5

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

4.2 Layout Descriptors

As described in Section 2.2 above, a layout descriptor represents a layout object and its attributes.

LayoutDescriptor ::= SEQUENCE {
LayoutObjectType,
OtherAttributes }
(FORMERLY CPS)

The layout object type defines the level of the object, i.e. Document, Page, Frame or Block, and whether it is generic or specific.

LayoutObjectType ::= SEQUENCE {
level INTEGER { document (0), page (1), frame (2),
block (3) },
INTEGER { generic (0), specific (1) } }

All other attributes of the object can also be represented in the descriptor.

OtherAttributes ::= SET {
Object Identifier,
GenericObjectIdentifier, -- OPTIONAL, -- only if the object itself is specific
CHOICE {
Index of Subordinate Objects,
Reference To Text Units },
Default Values OPTIONAL,
Page Size, -- only if page or document
Position and Dimensions, -- only if frame or block
Overlay Attributes, -- only if frame or block
Layout Attributes }

The object identifier is represented by... a number?

ObjectIdentifier ::= INTEGER -- ?? --

The descriptor for a specific object may contain a reference to a corresponding generic object (of the same level). The reference consists of the object identifier as contained in the descriptor of the corresponding generic object.

GenericObjectIdentifier ::= ObjectIdentifier

The descriptor of an object contains either an index of the objects subordinate to it, or (if there are no subordinate objects) references to one or more text units.

The index of subordinate objects consists of an (ordered) sequence of the object identifiers of each of the subordinate objects.

IndexOfSubordinateObjects ::= SEQUENCE OF ObjectIdentifier

The references to text units also form an ordered sequence in general, although frequently there will be only one such reference.

ReferenceToTextUnits ::= SEQUENCE OF TextUnitIdentifier

The predefined sets of default values for attributes are each denoted by a number. Each descriptor may include a reference to one such set of defaults

DefaultValues ::= INTEGER

The Page Size is also represented by a number (from a set of allowed values)

PageSize ::= INTEGER {A4(1), NA(2), }

The position and dimensions of a frame or block are each specified as a pair of numbers.

PositionAndDimension ::= SEQUENCE { position Vector
dimension Vector }

Vector ::= SEQUENCE { x INTEGER, y INTEGER }

The overlay attributes of a frame or block are represented by two lists, one of the ~~list~~ objects to which the object being described is transparent and one of those to which it is opaque.

OverlayAttributes ::= SET {
transparent List,
opaque List }

List ::= SEQUENCE OF ObjectIdentifier

All descriptors may contain a representation of layout attributes: they apply only to blocks but may appear at other levels in order to set up defaults for subordinate blocks.

LayoutAttributes ::= SEQUENCE {
ContentType,
CHOICE {
CharacterBoxAttributes,
PhotographicAttributes } }

The content type reflects the selection between character box and photographic.

ContentType ::= INTEGER { characterBox (p), photographic (1) }

The character box attributes are included only if the content type is 'character box'.

CharacterBoxAttributes ::= SET {
 characterPath Angle,
 characterOrientation Angle,
 characterSpacing INTEGER,
 lineProgression Angle, -- 90 and 270 only
 lineSpacing INTEGER,
 characterBoxSize Vector,
 baselineOffset INTEGER,
 GraphicCondition,
 proportionalPitch BOOLEAN -- for further study--}

Several of the attributes are represented by angles. In such cases the angle is one of ϕ , 90, 180 or 270 (although not all of these values are permitted in every case)

Angle ::= INTEGER {d ϕ (ϕ), d90(1), d180(2), d270(3)}

Graphic Condition itself is a multidimensional quantity.

GraphicCondition ::= SET {
 underline BOOLEAN,
 colour -- for further study--,
 font -- for further study--,
 type style -- for further study--}

The photographic attributes are included only if the content type is 'photographic'.

PhotographicAttributes ::= SET {
 pelPack Angle,
 lineProgression Angle, -- 90 only or 270 too?
 resolution INTEGER}

Annex B: (to Recommendation 5a)

Document architecture and interchange formats

4 Document architecture

4.1 Specific document architecture

4.1.1 The architecture that is particular to a given document is called specific document architecture. It consists of the following components (see Figure 1):

- content
 - layout structure
 - logical structure
 - layout directives
- } specific document structure

4.1.2 For the presentation on paper or screen the content of a document is physically structured into pages, blocks, lines, character boxes etc.. This structure is called layout structure and the objects building up this structure are called

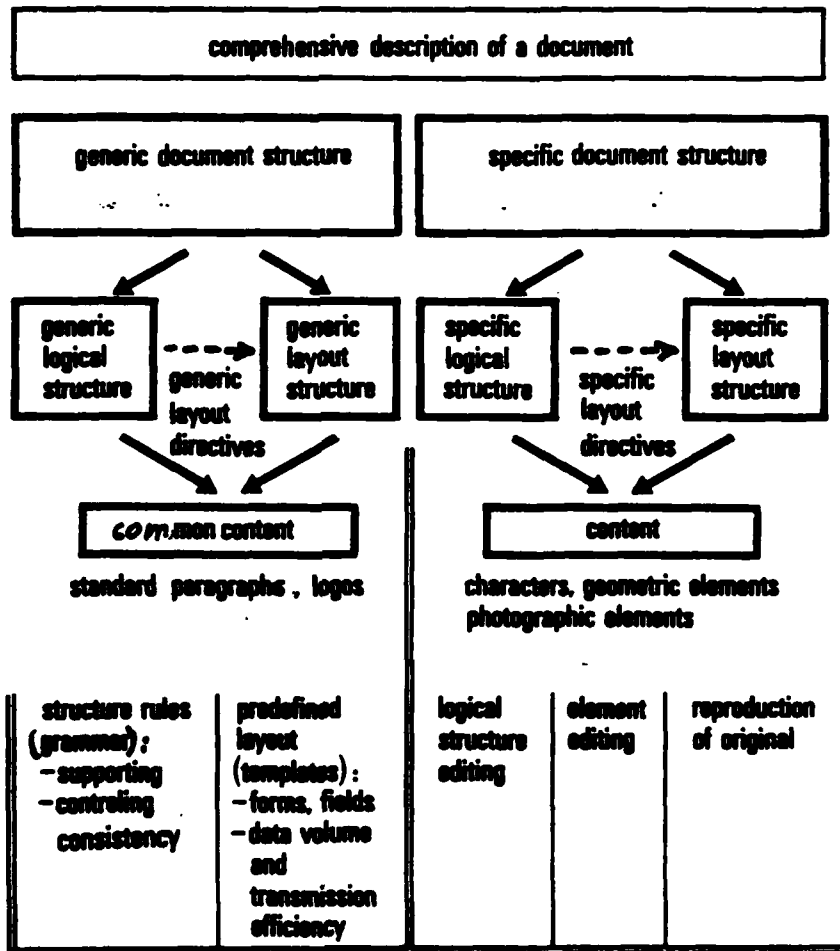


Figure 4. Document Architecture Model : Structure, Functions

layout objects . In the final form the content is divided into portions which belong to this layout objects.

- .4.1.3 On basis of the final form only such processing can be done efficiently which causes no reformatting in the environment of the manipulated layout object: It may comprise layout revisions like to scale and to move blocks within empty space or to overlay them transparently or opaquely. In cases where the environment needs to be reformatted, this has to be done manually by the user.

- .4.1.4 Documents use to be logically structured in order to enhance the comprehension of the text. In the final form of documents the logical structure may be implicitly expressed by the layout of the content, i.e. its arrangement within pages and its type style.

- .4.1.5 In the revisable form the logical structure of a document is explicitly represented by logical objects, e.g. like sections, paragraphs, footnotes, figures etc.. The content is divided into portions, which belong to the logical objects. The logical structure can be edited (revised) by commands like "insert, delete, move" etc. applied to logical objects like a paragraph. Figure 2 shows an example of a specific logical structure.

- .4.1.6 In the revisable form layout directives can be associated as attributes to the logical objects. These layout directives allow for the control of an automatic formatting and layout of the content portions belonging to the logical objects during editing (revision). Such layout directives may be "centered, left aligned, two columned" etc. and "emboldened, underlined, italic" etc. applied to paragraphs, sections etc..

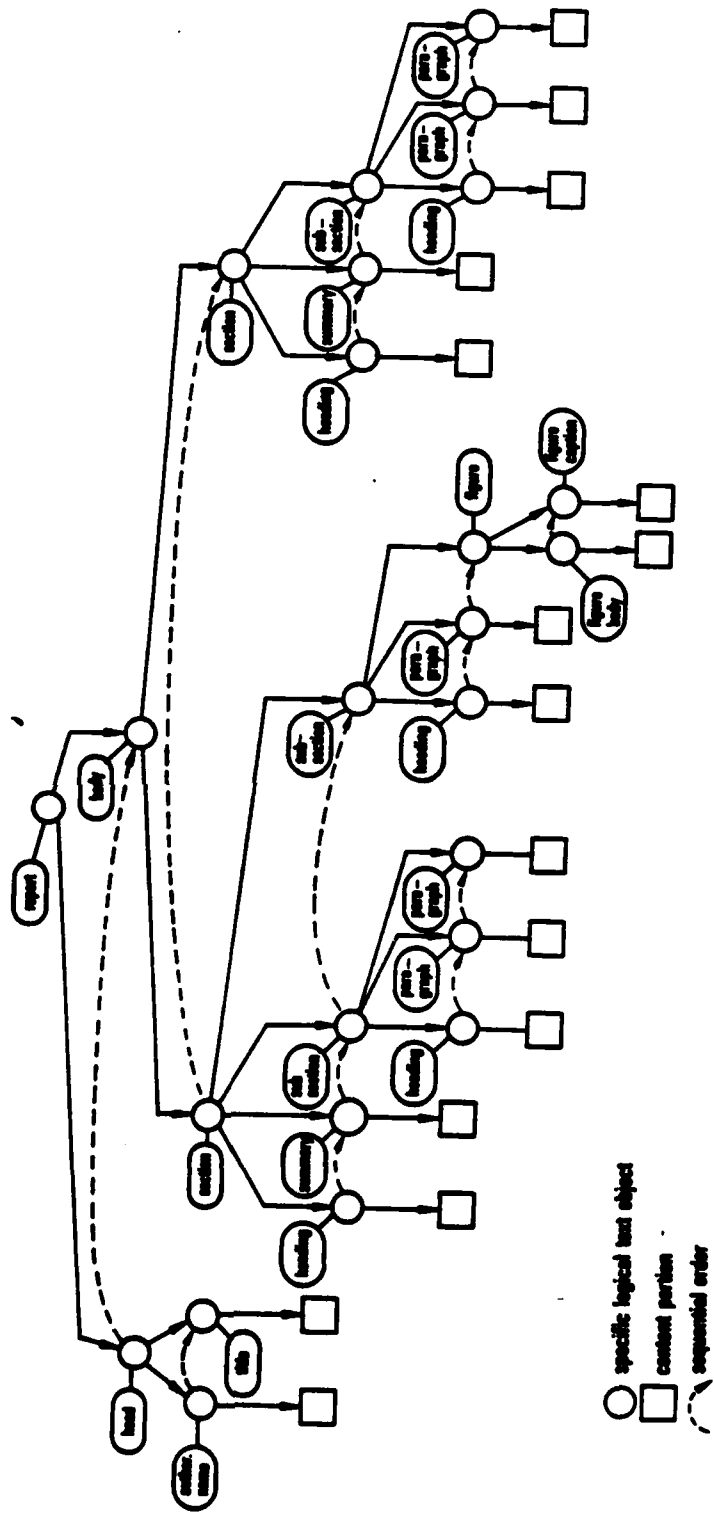


Figure 2. Example of a specific logical structure of a report (corresponds to Figure 4.)

In a given document common content portions may occur several times like a logo on several pages or a standard paragraph within several sections. For purpose of transmission efficiency the common content portions need to be transmitted only once in that part of the document interchange format, which contains the generic document structure. In the specific document structure there need to be only references to that common content portions.

4.2.4 The generic layout structure defines layout templates for pages containing the position of predefined blocks with common content (e.g. logos) and of "frames" (e.g. the image area, an address area) within which the content of the logical objects may be dynamically formatted. Such a template might represent a standardized form like ISO 3535. Figure 3 shows the example of a template of a form.

The generic layout structure allows also for predefined sequences of pages with predefined layout, e.g. a template for the "cover page" followed by a template for the "introduction" page and a template for all "new section pages" of a certain document class

4.2.5 The generic logical structure allows for the definition of types of logical objects, named generic logical objects, which are characteristic for the document class, and of the definition of their hierarchical order and their possible sequential order. The hierarchical order is expressed by a "consist of" relation, e.g. for a section consisting of none, or one or more subsections, which may consist of one or more paragraphs. The sequential order is expressed by a "followed by" relation, e.g. a head followed by none or one abstract, followed by one or more sections, followed by none or one reference list. The generic logical structure can be regarded as a set of rules (i.e. a grammar) from which specific logical structures can be derived. Figure 4 shows an example of a generic logical structure.

The formatting and layout process creates or modifies the layout structure and results in layout objects like blocks arranged and styled according to the layout directives.

4.2 Generic document architecture

4.2.1 A given document can be regarded as a member of a document class like a business letter, report, purchase order etc. The generic document architecture provides the user with means to define rules for the logical structure and templates for the layout which are characteristic for a given document class. A document class is defined by the application. It is not intended to standardize any document class by Recommendation S.a.

4.2.2 The specific architecture of a given document can be built according to the rules and templates of its document class. They are described by the generic document architecture of the document class which consists of the following components (see Figure 1):

- common content portions
 - generic layout structure
 - generic logical structure
 - generic layout directives.
- } generic
} document structure

The interchanged generic document structures help

- to improve the transmission efficiency,
- to maintain the consistency of the document with the document class during revision at the recipient and
- to facilitate the creation of new documents of a certain class.

4.2.3 The common content portions of a document class are predefined portions of text like the geometric elements of logos in forms, the character box elements of standard paragraphs in authority documents etc. which are common for all specific documents of that class.

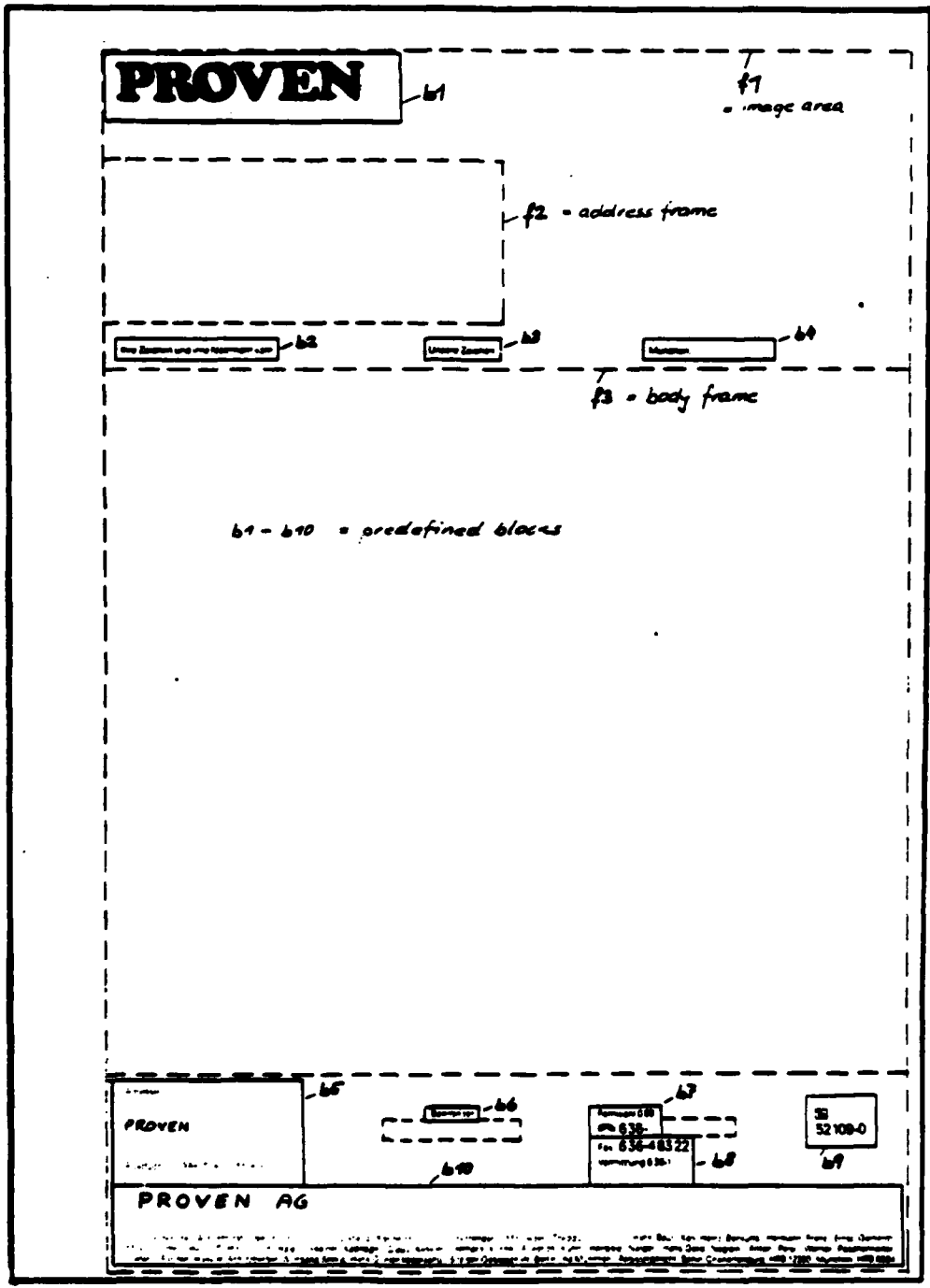
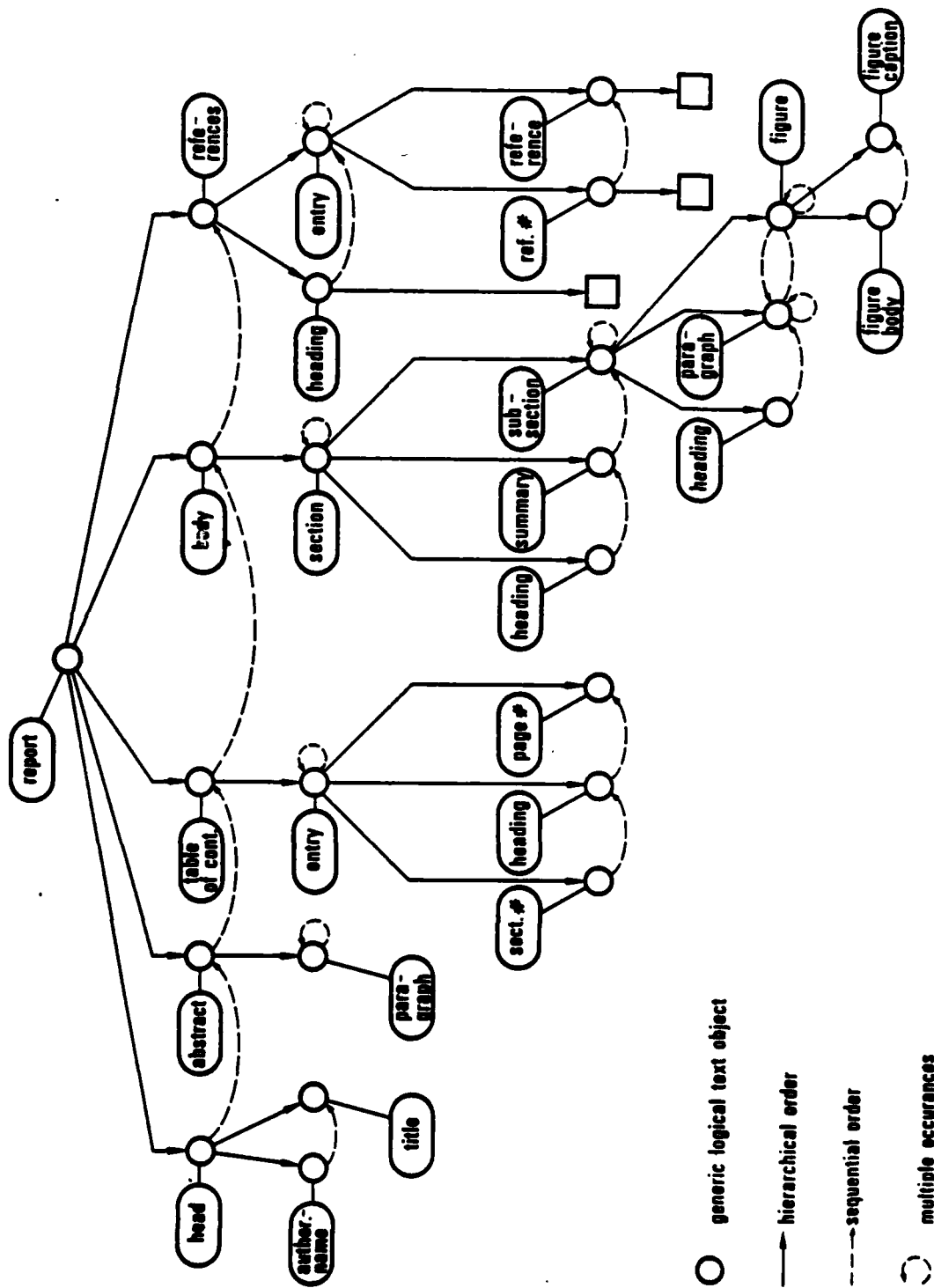


Figure 3. Example of a template of a letter form



- generic logical text object
- hierarchical order
- - - sequential order
- multiple occurrences

Figure 4 Example of a generic logical structure of a document class named "report"

4.2.6 The generic layout directives are attributes of the generic logical objects and apply to all specific logical objects of the same type. Similar to the specific layout directives, which are associated only to single logical objects the generic layout directives allow for the control of an automatic layout of the content of logical objects on the presentation media. There are two types of layout directives, the one effecting the positioning and the other effecting the type style of logical objects. Specific layout directives may overwrite generic layout directives. If a given document has been changed by this way it is no more a member of that class from which it originally has been derived. Figure 5 demonstrates the functions of the generic structures.

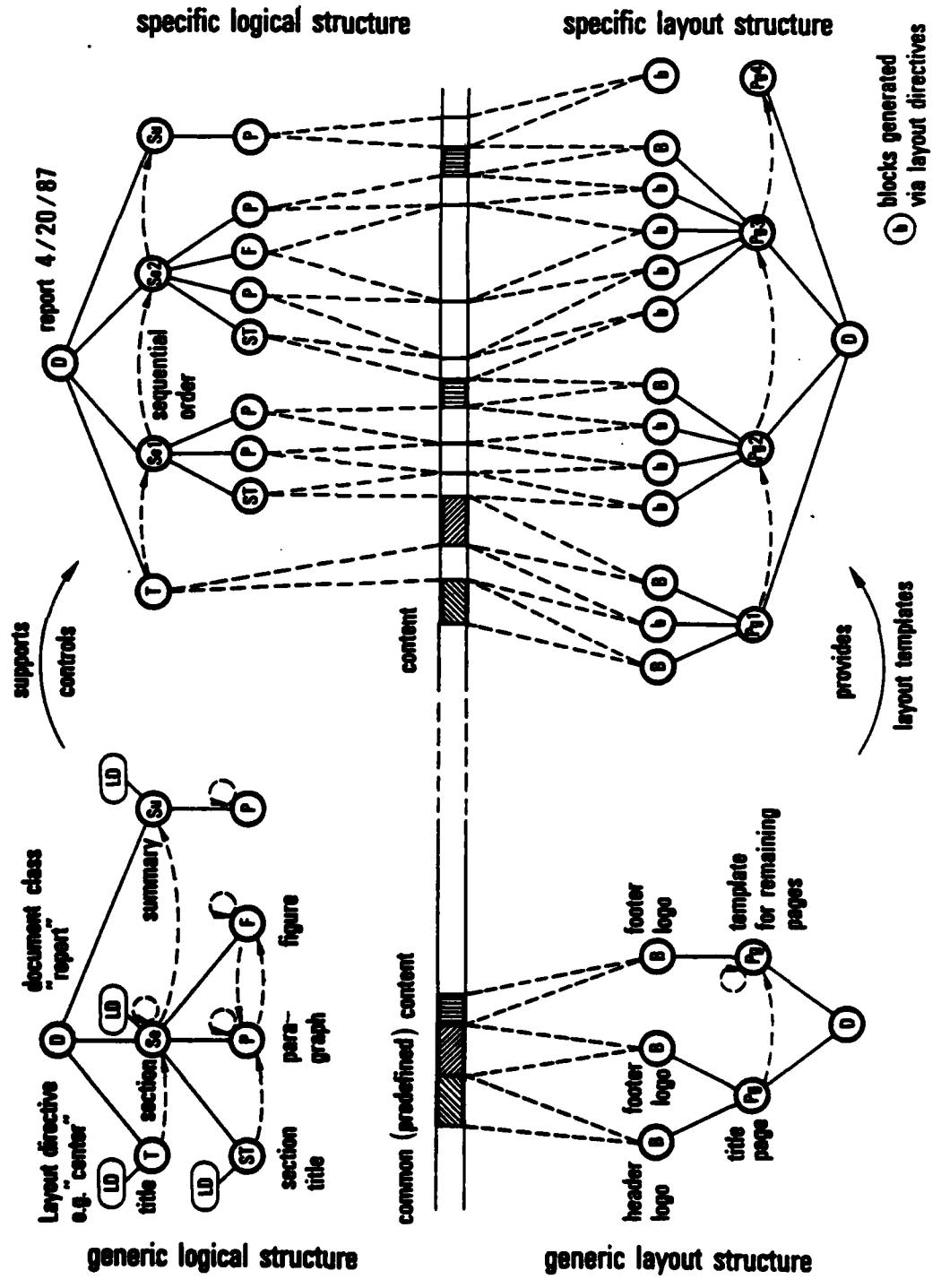


Figure 5. Functions of the generic structures

2 Interchange formats

2.1 Categories of interchange formats

By appropriate selection among the components of the document architecture different interchange formats with different capabilities can be derived.

Two major categories of interchange formats are distinguished

- text imaging formats (TIF)
- text processing formats (TPF)

2.2 Text imaging formats

2.2.1 Text Imaging Formats mainly support the imaging (printing, displaying) of documents at the recipient. The document is interchanged as an original in the final form by interchanging its content and its layout structure. The content and layout structure of such received documents can be edited (revised), however only by doing any reformatting manually.

The text imaging format offers no support for an automatic reformatting.

There are two text imaging formats. TIF.1 (Basic TIF) and TIF.2 (see Figure 6).

2.2.2 The text imaging format TIF.1 is called Basic TIF. It contains the content structured by the objects "pages", "frames" and "blocks" of the specific layout structure

The Basic TIF represents the final form of a formatted document and allows for an exact reproduction of its image at the recipient as created by the originator. There is at least one frame, which represents the image area ($\hat{=}$ printable area).

	specific layout structure	generic layout structure	specific logical structures + layout directives	generic logical structure + generic layout directives	features
TIF. 1	X				transmission of an original
TIF. 2	X	X			transmission of an original + transmission efficiency
TPF. 1		X	X		unformatted document transmission + full processability
TPF. 2		X	X	X	unformatted document transmission + full processability + consistency control
TPF. 3	X	X	X	X	transmission of an original + full processability + consistency control

Figure 6: Overview of the text interchange formats

For the easy repositioning of logically related blocks which need to retain their relative positions during layout revision at the recipient, there might be additional frames which enclose these blocks. Such a frame could be one enclosing a diagram block and several caption blocks of a figure. However these frames need not necessarily be interchanged.

Note: In the context of TPF frames have an additional essential function. They define boundaries within which the content of the objects of the logical structure can be automatically formatted. Therefore layout directives which effect the positioning refer to frames.

2.2.3 The text imaging format TIF.2 adds to the objects of the Basic TIF the objects of the generic layout structure.

It allows for transmission efficiency. Predefined content of layout objects which occur repetitively on different pages, like components of forms, are transmitted only once within the generic part of the text imaging format. In cases where the document class is known by the recipient, e.g. within a company or for standard forms such as e.g. ISO 3535, TIF.2 contains only the name of the document class within the generic part of the format and the detailed generic information is added by the recipient.

2.3 Text Processing Formats

2.3.1 Text Processing Formats support the processing of documents at the recipient. The document is interchanged in the revisable form by interchanging its content, the logical structure and the layout directives. The content, the logical structure and the layout structure can be edited at the recipient supported by automatic reformatting.

There are three text processing formats, TPF.1 (Basic TPF), TPF.2 and TPF.3 (see Figure 6)

1.3.2 The text processing format TPF.1 is called Basic TPF. It contains the content structured by the objects of the specific logical structure, the specific layout directives and the objects of the generic layout structure. In the case of TPF.1 the positioning layout directives refer to the generic frames of the generic layout structure (e.g. the image area).

The TPF.1 allows for transmission efficiency in saving the interchange of a not yet final layout. The unformatted document can be formatted at the recipient according to the specific layout directives given by the originator which may be accomplished and revised by the recipient. The content, the logical structure and the layout structure of the document can be edited together with an automatic formatting according to the layout directives.

1.3.3 The text processing format TPF.2 adds to TPF.1 the rules and objects of the generic logical structure and the generic layout directives.

The TPF.2 allows for the interchange of unformatted documents which can be formatted at the recipient according to the generic layout directives given by the document class and the specific layout directives explicitly defined by the originator. The document can be revised under control by the generic logical structure which helps to maintain the consistency with the properties of the document class. The types of logical objects which may occur in the specific logical structure and their possible hierarchical and sequential orders are defined. In order to format the document, layout directives have to be added to the logical ob-

jects by the recipient. In cases where the document class is known at the recipient the generic part of the format contains only the name of the document class and the detailed generic information is added by the recipient.

2.3.4 The text processing format TPF.3 adds to the objects and layout directives of TPF.2 the objects of the specific layout structure.

The TPF.3 allows for the interchange of an already formatted but still fully revisable document. It adds to the processing capabilities offered by the TPF.2 the capability of reproducing the image at the recipient exactly identical to the one originated by the sender. This format may be named "full text format".

Combined Symbol Matching Facsimile Data Compression System

WILLIAM K. PRATT, SENIOR MEMBER, IEEE, PATRICE J. CAPITANT, MEMBER, IEEE,
WEN-HSIUNG CHEN, ERIC R. HAMILTON, MEMBER, IEEE, AND ROBERT H. WALLIS

Abstract—A facsimile data compression system, called combined symbol matching (CSM), is presented. The system operates in two modes: facsimile and symbol recognition. In the facsimile mode, a symbol matching operator isolates document symbols such as alphanumeric characters and other recurring binary patterns. The first symbol encountered is placed in a library, and as each new symbol is detected, it is compared with each entry of the library. If the comparison is within a tolerance, the library identification code is transmitted along with the symbol location coordinates. Otherwise, the

new symbol is placed in the library and its binary pattern is transmitted. Nonisolated symbols are left behind as a residue, and are coded by a two-dimensional run-length coding method. In the symbol recognition mode, the library is preworded and each entry is labeled with its ASCII code. As each character is recognized, only the ASCII code is transmitted.

Computer simulation results are presented for the CCITT standard documents. With text-predominant documents, the compression ratio obtained with the CSM algorithm in the facsimile mode exceeds that obtained with the best run-length coding techniques by a factor of two or more, and is comparable for graphics-predominant documents. In the symbol recognition mode, compression ratios of 250:1 have been achieved on business letter documents.

Manuscript received September 26, 1979; revised November 9, 1979.
The authors are with Compression Labs, Inc., 10440 North Tustin Avenue, Cupertino, CA 95014.

0018-9219/80/0700-0786\$00.75 © 1980 IEEE

I. INTRODUCTION

MOST facsimile coding systems previously developed have been based on the concept of run-length coding [1]. Run-length coding methods provide a relatively high compression ratio for a graphics type of document or an alphanumeric document containing a small amount of text [2]. But, the achievable compression ratio drops appreciably if a document is filled densely with alphanumeric characters because the black and white run-lengths become quite short. Dense alphanumeric documents can be efficiently coded by symbol recognition techniques in which individual symbols are detected and coded by a prototype library code [3], [4]. However, such a method cannot effectively handle documents containing a mixture of alphanumerics and graphics. One proposed approach to this problem has been to segment a document into stripes containing alphanumeric text or graphics data, and then code the former by symbol matching and the latter by run-length coding [5]. The problems with this approach are the difficulty of document segmentation and the drop in compression performance if the segmentation is not accurate. This paper introduces a new concept of hybrid symbol-matching/run-length coding in which a document is dynamically segmented into symbol and graphics regions [6].

Conceptually, the symbol versus graphics segmentation process employed in the facsimile compressor is quite simple. A document is scanned line by line, and all isolated symbols that are expected to recur in the document are extracted and coded by a symbol-matching process. The remainder of the document, called the residue, is coded by two-dimensional run-length coding. This segmentation method permits document symbols to be coded by symbol matching without interference from the graphics portions of a document, and eliminates symbols from that portion of the document which is run-length coded. The result is an efficient match between the type of data and the chosen coding methods.

The symbol-matching process previously described has been adapted to recognize alphanumeric characters in a document. In this symbol recognition mode of operation, the document is represented by conventional printer codes: character, space, carriage return, etc.

The following sections describe the combined symbol matching (CSM) algorithm for both the facsimile and symbol recognition modes of operation.

II. FACSIMILE CODING MODE

The block diagram of Fig. 1 describes the basic elements of the CSM coding system for facsimile coding. In operation, a number of scan lines equal to about two to four times the average character height are stored in a scrolled buffer. This data is then examined line by line to determine if a black pixel exists. If the entire line contains no black pixel, this information is encoded by an end-of-line code. If a black pixel exists, a blocking process is conducted to isolate the symbol. For those isolated symbols, further processing is required to determine if a replica of the symbol under examination already exists in the library. This process involves the extraction of a set of features, a screening operation to reject unpromising candidates, and finally a series of template matches. The first blocked character and its feature vector are always put into the prototype library, and as each new blocked character is encountered, it is compared with each entry of the library that passes the screening test. If the comparison is successful, the library identification (ID) code is transmitted

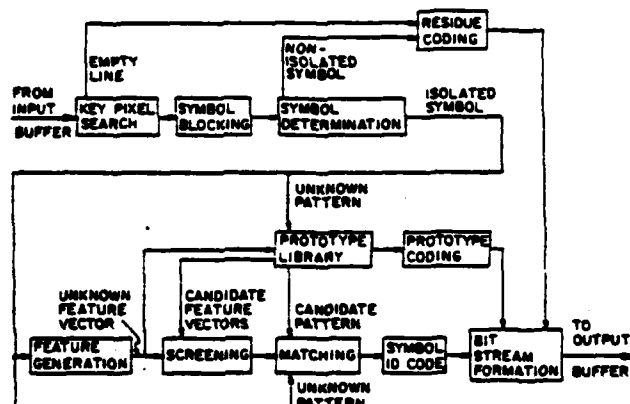


Fig. 1. CSM facsimile coder.

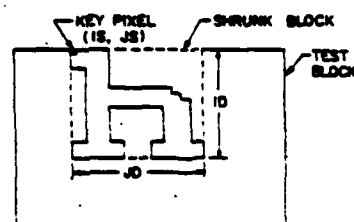


Fig. 2. Block shrinking.

along with the location coordinates of the symbol. If the comparison is unsuccessful, the new symbol is both transmitted and placed in the library. Those areas of a document in which the blocker cannot isolate a valid symbol are assigned to a residue, and a two-dimensional run-length coding technique is used to code the residue data. The following sections describe key elements of the coder in greater detail.

A. Symbol Blocking

The function of the symbol blocker is to examine the input buffer in a systematic fashion, and to locate the position and size of any isolated characters. Fig. 2 illustrates the blocking process. A black pixel in the buffer, denoted by the character "1" is considered to be a *key pixel* whenever the four neighbors located above it and to its left are white, as shown below

000

01.

Whenever a key pixel is encountered, the blocker is initiated. The blocker extracts those pixels from the buffer that are contiguous with the key pixel, or enclosed by a set of contiguous black pixels. For example, with the lower case letter "s," all black pixels and the enclosed white "island" will be extracted by the blocker.

B. Feature Extraction

The most straightforward method to determine whether a match exists between an unknown symbol and one of the symbols stored in the library is to perform a template match between the unknown and every library symbol. However, a two-dimensional template match is costly in terms of processing time and equipment. A method of reducing the number of such matches is required. The approach that has been taken is

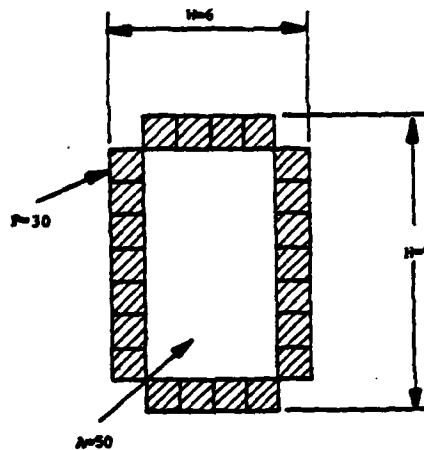


Fig. 3. Perimeter features, W = symbol width, M = symbol height, P = symbol outer perimeter, A = black area plus enclosed white area.

to extract a set of scalar "features" from the various symbols in the library. These features are used to reduce or "screen" the number of candidates for a template match to a tiny fraction of all the possibilities in the library.

The features used in the screening process are the block height, block width, symbol perimeter, and pixel area enclosed by the outer boundary of the symbol. Fig. 3 provides an example of features derived from a character.

C. Candidate Screening

The purpose of the screening process is to reduce the burden on the template matcher by passing only "good prospects" to the matcher. This is accomplished by calculating the feature space distance between the unknown and each library entry, and selecting the library candidate with the smallest distance as the best prospect for a match. If this match is rejected, the next best candidate is considered, and so forth. The distance "metric" used to determine how "close" an unknown is to a particular candidate is the "city block" distance defined by

$$D(U, C) = \sum_{i=1}^{N_F} |F_C(i) - F_U(i)| \quad (1)$$

where $F_C(i)$ is the i th feature of the candidate, $F_U(i)$ is the i th feature of the unknown, $|\cdot|$ denotes the absolute value, $D(U, C)$ is the distance between the unknown and candidate, and N_F is the number of features.

D. Template Matcher

The template matcher forms a comparison between the binary patterns of a detected symbol and a library prototype symbol. Consider a two-dimensional binary pattern represented by $A(C, R)$ where $C = 1, 2, \dots, N_C$ and $R = 1, 2, \dots, N_R$. A conventional template matcher calculates the similarity between a pair of vector patterns $A(C, R)$ and $B(C, R)$ by summing the number of picture elements (pixels) for which $A(C, R)$ and $B(C, R)$ differ. This EXCLUSIVE OR error is defined as

$$E = \sum_{C=1}^{N_C} \sum_{R=1}^{N_R} A(C, R) \odot B(C, R) \quad (2)$$

where \odot denotes the Boolean EXCLUSIVE OR operation.

A major shortcoming of the conventional template matcher described above is that it treats all errors alike regardless of where they occur spatially. An improved matcher, to be described, utilizes a "weighted EXCLUSIVE OR" error criterion that is based on the context in which the error occurs.

The motivation behind the weighted EXCLUSIVE OR error criterion may be appreciated by examining the EXCLUSIVE OR error (denoted $A \oplus B$) in Figs. 4 and 5. Compare the EXCLUSIVE OR pattern for the "c" and "o" in Fig. 4 with the pattern for the pair of "e's" in Fig. 5. Note that the EXCLUSIVE OR error count for the pair "c" and "o" (count = 23) is actually *less* than that for the pair of "e's" (count = 29) implying that, by this error metric, "c" and "o" are "closer" than the pair of "e's" are to each other. However, the error pattern for the pair of "e's," which should be declared a match, is composed of *sparsely distributed* pixels, while the error pattern for the "o" and "c" shows a *dense node* of error pixels corresponding to the missing right segment of the "o." One way to quantify the density of such a "node" is to form a summation in which the "local density" of every black pixel is merely the sum of all the pixels in its 3×3 neighborhood if the pixel is 1, and 0 if the pixel is 0. The patterns above labeled "weighted XOR error" have been calculated in this manner. Note that by this criterion, the associated counts indicate that the pair "c" and "o" are more separated (count = 131) than are the pair of "e's" (count = 73).

In the template matcher, the weighted EXCLUSIVE OR error is computed for nine translation shifts of a pair of patterns corresponding to horizontal and vertical single pixel shifts of the patterns. The minimum error is then compared to a threshold in order to determine whether or not a match should be declared. The value of the threshold is a non-linear function of the symbol's black count, and is obtained by an empirically determined look-up table.

E. Library Maintenance

A fixed size library is used in the CSM system. The first blocked character and its feature vector occupy the first library slot. The subsequent library slots are occupied by those blocked characters for which no match is found. In order to prevent the library from overflowing, a scoring system is employed to track the usefulness of the library elements. When the library is filled, the least used prototype

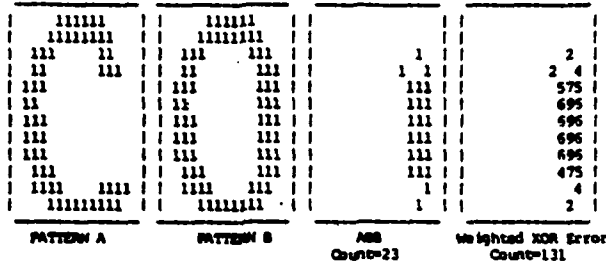


Fig. 4. Example of exclusivity pattern for c and o.

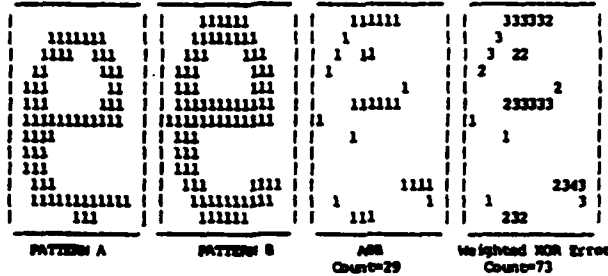


Fig. 5. Example of exclusivity pattern for pair of e's.

```

dddddddddddddddddddddddd
llllllllllllllllllllllllll
eeeeeeeeeeeeeeeeeeeeeeee
.....
tttttttttttttttttttttttttt
bbbbbbbbbbbbbbbb
oooooooooooooooooooooooo
nnnnnnnnnnnnnnnnnnnnnnnn
cccccccccccccccccccccccc
mmmmmmmmmmmmmmmmmmmmmm
ssssssssssssssssssssss
pppppppppppppppppppppppp
uuuuuuuuuuuuuuuuuuuuuuu
GGGGGG
TTTT
VVVVVVVV
FFFFFFFFFFFFFFFFFFFFFFFF
    
```

Fig. 6. Partial library plot of CCITT no. 4.

```

間間間間
あああ
oooooooooooooooooooooooo
.....
るるるる
-----
111111
-----
第第第第
oooooooooooooooooooooooo
-----
△△△△△
-----
信信信
nnnnnnnnnnnnnnnnnnnnnnn
上上上上上
找找找找找找找
    
```

Fig. 7. Partial library plot of CCITT no. 7. (Symbol blocking performance using algorithm defined in [6].)

is bumped out of the library and replaced by the new prototype. At the receiver, the same size library and the same scoring system are utilized to maintain synchronization with the transmitter. With a library size of N elements, the scoring system gives every "new prototype" or "matched symbol" at least N chances for a match.

Figs. 6 and 7 contain partial library plots of isolated symbols from two facsimile documents, one a French journal article (CCITT #4), and the other a Japanese language document (CCITT #7). The first item on the list is the first isolated prototype symbol, and all following symbols represent matches to the prototype.

est en fait plus variable que $T/\Delta f$ en plus.
 A cet égard la figure 2 représente le tracé courbé
 est (dC/dt) en fonction de f pour les valeurs nom-
 indiquées page précédente.

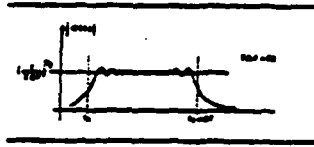
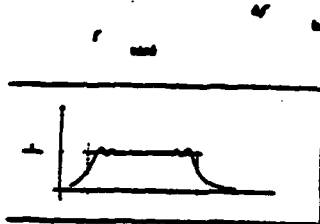


FIG. 2

en cas, le filtre adapté pourra être considéré,
 relatif à la figure 1, par la somme :

d'un filtre passe-bande de transfert unitaire pour
 $f < f_c + \Delta f/2$ et de transfert quasi nul pour
 $f > f_c + \Delta f/2$. Il ne transmet pas la phase
 importante le transfert :

(a) ORIGINAL



(b) RESIDUE

elle figure à retard est donné

$$\phi = -2\pi \int_0^T T_c \Delta f$$

$$\phi = -2\pi \left[T_c + \frac{\Delta f T_c^2}{2} \right]$$

Et cette phase est bien l'op-
 à un déphasage constant ;
 et à un retard T_c , près (le
 Un signal réel $S(t)$ traversé
 donne à la sortie (à un retard
 sans plus de la paroi) un sig-
 de Fourier en réalité, complexe
 et celle de part et d'autre de
 être un signal de fréquence
 dans l'enveloppe à la forme i
 où l'on a approximé constant
 et le signal $S_c(t)$ correspond
 de filtre adapté. On compare
 à compression d'impulsion c
 filtre adapté : la « largeur » (à
 prise) doit être à $1/\Delta f$, le r
 est de $\frac{T}{1/\Delta f} = T\Delta f$

$$\int T \Delta f$$

$$\left[\frac{T}{\Delta f} \right]$$

T

$\frac{T}{\Delta f}$

Fig. 2. Magnified section of CCITT no. 4 and its residue.

F. Prototype Coding

After a symbol has been blocked, a decision threshold is applied to each prototype element of the library that has passed the screening test. If a match is indicated, only the matching library ID and horizontal location with respect to the previous symbol are coded. Otherwise, the binary pattern of the blocked symbol is transmitted along with the symbol width, symbol height, and horizontal location, in addition to being placed in the library as a new prototype element.

The simplest method of prototype coding is to binary code the pixels within a block in a raster scan fashion. On the average, about 30 percent of the prototype code bits can be eliminated by scanning the prototype pixels in a folded "basket weave" sequence and applying one-dimensional Huffman coding of the run lengths. The disadvantages of this approach are additional implementation complexity and possible loss of bitstream synchronization when a channel error occurs. The binary coding approach has been adopted for a high-performance version of the CSM facsimile coder, and the folded run-length coding method is used for a very-high-performance version.

Residue Coding

In many documents, there are black pixel patterns that do not meet the criteria of prototype characters. Examples in-

clude exceptionally large or exceptionally small alphanumeric characters, segments of company logos, and segments of handwritten script. In the CSM system, these patterns are rejected by the symbol blocker, and then left behind as a residue to be coded by two-dimensional run-length coding. Fig. 8 presents a blow up of a section of a facsimile document (CCITT #4) and its corresponding residue.

Conceptually, the CSM system could employ any type of run-length coding method for residue coding. The selection should be made on the basis of coding performance, tolerance to channel errors, implementation complexity, and compatibility with facsimile standards. Considering these factors, a modified version of the CCITT two-dimensional run-length coding algorithm has been selected for the residue coder. By inhibiting the symbol matching process, the CSM coder will automatically revert to a pure residue coder, which can be made exactly compatible with the CCITT standard.

H. Transmission Code

The CSM facsimile coding system produces an asynchronous code that is dependent upon the contents of the document to be coded. Table I contains a detailed specification of the code elements and Fig. 9 contains a state diagram defining the code. The code words lengths in this specification have been optimized for a scan resolution of 8 X 8 pixels/mm.

TABLE I
CLI COMBINED SYMBOL MATCHING FACSIMILE CODE

CODE NAME	CODE DEFINITION	WORD SIZE (BITS)	DESCRIPTION
LINSYN	LINE SYNC	9	ENCODING OF HORIZONTAL SCAN LINE, TRANSMITTED EVERY K LINES
SYMPLG	SYMBOL FLAG	1	0 = NO SYMBOL ON LINE 1 = AT LEAST ONE SYMBOL ON LINE
COLADD	COLUMN ADDRESS	11	HORIZONTAL LOCATION OF FIRST PIXEL ON LINE
MATCHG	MATCH FLAG	1	0 = NO MATCH 1 = MATCH
LIBID	LIBRARY INDEX	7	BINARY CODE OF LIBRARY INDEX
NONPOS	NON POSITION	2	VERTICAL MATCH SHIFT 00 = NO SHIFT 01 = UP SHIFT 10 = DOWN SHIFT
BLKSIZ	BLOCK SIZE	10	BLOCK SIZE (HEIGHT, WIDTH)
BLKCD	BLOCK CODE	VAR.	BINARY CODE OF BLOCK CONTENTS
DELCO	DELTA COLUMN	6,17	HORIZONTAL DISTANCE FROM CURRENT BLOCK TO NEXT BLOCK (UNIQUE CODE WORD INDICATES LAST SYMBOL ON LINE)
RESCO	RESIDUE CODE	VAR.	TWO-DIMENSIONAL RUN-LENGTH CODE
STRT	STUFF BITS	VAR.	BITS INSERTED IN TRANSMISSION CODE TO PREVENT FACSIMILE CODE FROM ASSUMING STATE OF LINE SYNC CODE

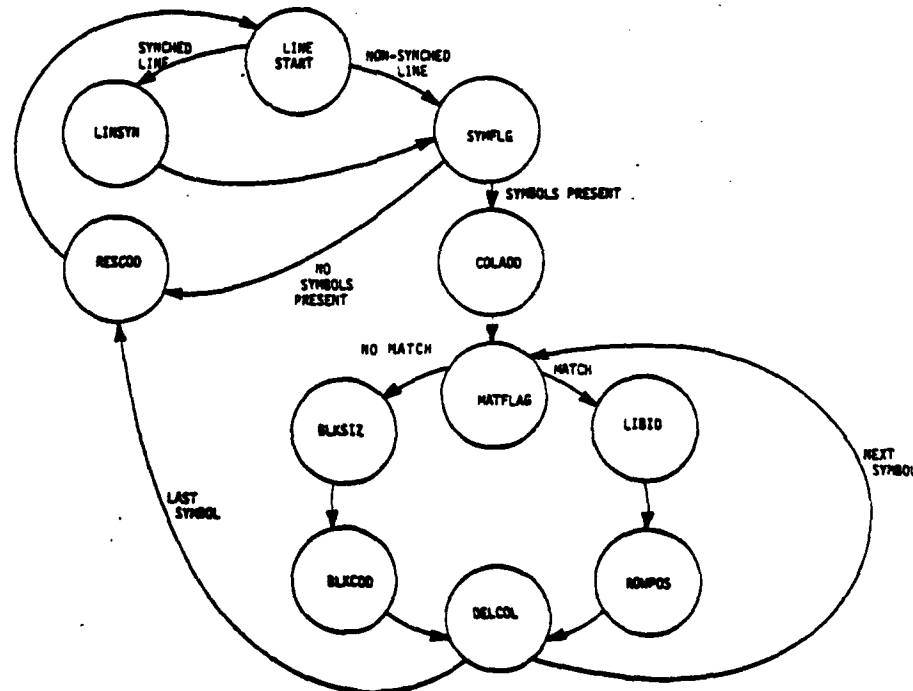


Fig. 9. Transmission code state diagram.

1. Extensions of CSM Concept

In a typical business letter scanned at 8 X 8 pixels/mm, about 40 percent of the compressed code bits are devoted to the transmission of prototype symbols. Almost all of this portion of the transmission code can be eliminated if the documents to be transmitted are restricted to a fixed set of symbols, for example, Courier typewriter font. In this case, the transmitter and receiver libraries can be prestored with

symbols. Isolated unknown symbols detected in the key pixel scanning process that do not match a library entry can be placed in the residue for subsequent run-length coding.

The symbol matching process in the CSM system is not exact; a match tolerance is permitted between symbols to accommodate perturbations in symbol shape caused by the scanning process. As a consequence, in the basic CSM system, a reconstructed document is not an exact pixel-by-pixel replica

of the original document. Although symbol substitution errors are extremely rare, there may be applications in which exact coding is demanded. This mode of operation can be accommodated in the CSM system by a simple modification of the coder and decoder. At the coder, after a successful match, the EXCLUSIVE OR between the pair of matched symbols is formed and placed in the residue for subsequent run-length coding. At the decoder, the pixel arrays generated from reconstructed symbols and reconstructed residue are combined in an EXCLUSIVE OR fashion to correct for differences in the pair of matched symbols. In this manner, exact reproduction is achieved. However, the "overhead" associated with the exact reproduction mode of operation can reduce the achievable compression ratio by as much as 50 percent at 8×8 pixel/mm resolution.

III. SYMBOL RECOGNITION MODE

The CSM algorithm achieves facsimile data compression by the matching of document symbols against a library of symbols accumulated during the document scan. If a match occurs, the library index is transmitted rather than the symbol binary pattern. This basic concept can be extended to perform symbol recognition by preloading the library with the binary symbol patterns of a predetermined set of symbol fonts. The coder can then operate in a symbol recognition mode in which only the ASCII codes are transmitted and all other document data such as a signature or logo are ignored.

A. Line Tracking

In the western world, printed matter is "read" from left to right and from top to bottom. Therefore, a symbol blocking system that transmits its output to a serial ASCII terminal must do the same. However, the CSM algorithm extracts characters from the document being scanned in a totally different fashion. As the line buffer scrolls through the page from top to bottom, the tallest of first encountered characters are removed from the document and processed through the recognition algorithm. Thus characters emerge from the CSM process in a sequence which would be totally incomprehensible if viewed in chronological sequence. In the conventional CSM facsimile transmission mode, this is of no consequence, since characters are placed in their appropriate address locations regardless of their order of occurrence. In the serial symbol recognition mode, the transmitter will assign each character an ASCII code, assemble the codes into lines, inserting blanks, line feeds, carriage returns, etc., and transmit the lines serially to the receiver. For single spaced or rotated documents, this "line tracking" is more difficult than one would imagine. The problem is basically that of grouping the characters into lines. Determining the sequence in which they should be transmitted is relatively easy since the characters may be sorted by their column addresses. A significant benefit of this serial ASCII mode is that no information on character location need be transmitted, since the correct sequence is all that is required in order to properly reconstruct the received document.

The line-tracking algorithm is based on a straight line fit of the key pixel coordinates of characters on a text line, as illustrated in Fig. 10. The straight line is defined parametrically as

$$R = S \cdot C + O \quad (3)$$

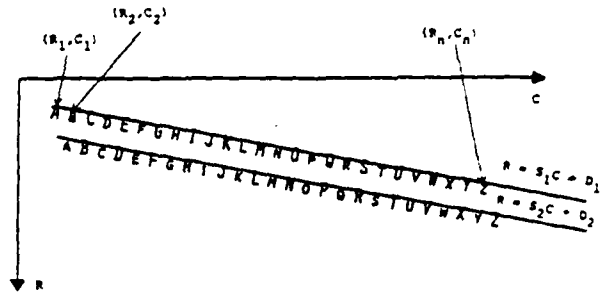


Fig. 10. Line tracking.

where R represents the row index, C is the column index, S denotes the text line slope, and O is its offset. As characters are encountered, they are assigned to the nearest straight line representing a text line. The algorithm is as follows:

- 1) The coordinates (C, R) of the first encountered character are used as a "seed" to start a cluster at $S = 0, O = R$.
- 2) The (C, R) coordinates of the next character encountered are used to compute $E = [R - S \cdot C]^2$ for the slope and offset of each cluster.
- 3) If the error is less than a threshold for a given cluster, the character is assigned to that cluster (next line). If it is greater than the threshold for all clusters, the oldest cluster is dumped, and a new cluster is started.
- 4) If the character was added to an existing cluster, the values of slope and offset are updated by use of minimum-mean-square error techniques.

B. Handling of Special Characters

A number of characters which consist of two "subcharacters" must be treated as special cases in the symbol-recognition mode. This is because the blocker/matcher would otherwise fragment them into their constituent parts and give misleading results. These characters are: (i), (j), (!), (?), (:), (;), (=), and ("). After recognition of the two parts of the character, the system will check if two compatible symbols are on top or almost on top of each other. If so, the two symbols are merged into one. For example two (·)'s on top of each other will be merged into a (:).

IV. COMPRESSION RATIO EVALUATION

The CSM system has been extensively evaluated by computer simulation to optimize its performance and to determine its compression ratio with respect to other coding methods.

A. Facsimile Mode Evaluation

The CCITT document set of eight digitized documents of 200×200 line/in (8×8 pixels/mm) resolution, shown in Fig. 11, has been used for evaluation of the CSM system in its facsimile mode of operation. Tables II and III contain listings of the compression ratios for each of the documents for the high-performance and very-high-performance versions of the CSM algorithm, respectively. These tables also contain the bit allocations for each of the code elements defined in Table I.

Table IV presents a summary comparison of the compression ratios of the high-performance and very-high-performance CSM systems with several other facsimile coding methods. The modified Huffman code is the CCITT adopted standard for one-dimensional run-length coding [2]. The IBM code

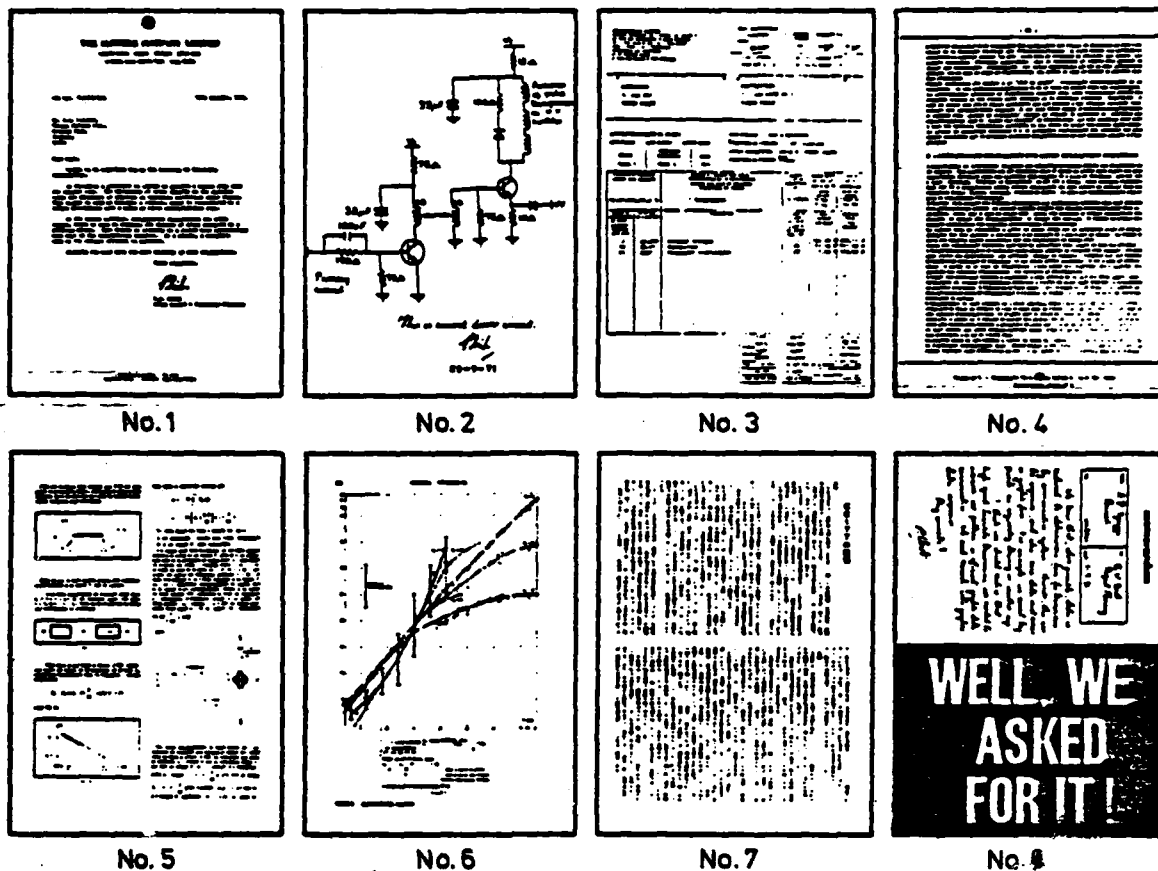


Fig. 11. CCITT facsimile document set.

[7], READ code [8], and BPO code [9] are proposals for a CCITT standard employing two-dimensional run-length coding. These algorithms all provide for an end-of-line code. All of the algorithms in Table IV have been simulated and evaluated on the same set of digitized documents scanned at the University of Hannover, Germany. The K factor indicates the number of lines in which the coder is operated in its two-dimensional mode before it reverts to a one-dimensional mode to limit the propagation of errors.

Comparison of the compression performance of these algorithms indicates that the CSM methods outperform the run-length coding techniques substantially for text-predominate documents, and perform at about the same level as the best of the two-dimensional run-length coding methods for graphics-predominate documents.

B. Symbol Recognition Mode Evaluation

The symbol recognition mode system has been tested with 86 sets of data, each containing 1000 samples of one of the 86 symbols of the Courier 10 font. In these tests, no mismatches occurred, and only very badly damaged characters were rejected.

Fig. 12 contains an example of a business letter and its reconstruction with the symbol matching coding mode of operation. It should be noted that the reconstructed letter has been printed with a different font than the original, however, the format and spacing of the two letters are in basic agreement.

The compression factor obtained for this document for operation of the CSM system in the symbol matching mode is about 257:1 and for operation in the facsimile mode is about 49:1.

V. SYSTEM IMPLEMENTATION

Although the CSM system is more complex to implement than a conventional two-dimensional run-length coding system, with the advent of high-speed and relatively inexpensive memory, discrete logic circuits, and microprocessors, implementation complexity has ceased to be a deterrent to the development of high-performance systems. A 100 X 100 lines/in (4 X 4 pixel/mm) facsimile coder using the CSM algorithm was introduced by Compression Labs, Inc. of Cupertino, CA, in Fall 1978. This unit utilizes a microprocessor to implement the algorithm for transmission at sub-minute page rates. A discrete logic implementation of the CSM algorithm is being developed by Compression Labs for transmission rates of less than 5 s for a 200 X 200 lines/in page.

VI. SUMMARY

A new high-performance method of facsimile data compression, called CSM, has been introduced. The coding system involves segmentation of a document into symbols, that are coded by template matching, and into a residue of the remainder of the document, that is coded by two-dimensional run-length coding. Computer evaluation indicates that the compression factor for text-predominate documents is about

TABLE II
HIGH-PERFORMANCE CODER SUMMARY

	DOCUMENT							
	1	2	3	4	5	6	7	8
<u>STPBIT</u>	699	387	2557	1613	1700	2299	6036	1792
<u>LINSYN</u> <u>K=32</u>	603	603	603	603	603	603	603	603
<u>SYNPLG</u>	2128	2128	2128	2128	2128	2128	2128	2128
<u>COLADD</u>	2497	385	4169	8767	4895	1980	11176	1584
<u>MATPLG</u>	988	37	1291	4015	1754	330	2522	156
<u>LIBID</u>	5985	168	7574	26292	10773	1253	9870	434
<u>ROWPOS</u>	1710	48	2164	7512	3078	358	2820	124
<u>BLKSIZ</u>	1330	130	2090	2590	2150	1510	11120	940
<u>BLCCOD</u>	24691	3404	47036	55360	44789	30534	406471	28115
<u>DELCOI</u>	8227	233	9341	34177	14055	2431	25692	1024
<u>RESCOD</u>	18594	91323	53880	20752	42014	93847	19845	181353
<u>TOTAL</u>	67452	98846	132833	163809	127939	137273	499083	218253
<u>COMP.</u> <u>RATIO</u>	54.5	37.2	27.7	22.4	28.7	26.8	7.4	16.8

TABLE III
VERY-HIGH-PERFORMANCE CODER SUMMARY

	DOCUMENT							
	1	2	3	4	5	6	7	8
<u>STPBIT</u>	439	266	1431	1182	1152	1560	3979	936
<u>LINSYN</u> <u>K=32</u>	603	603	603	603	603	603	603	603
<u>SYNPLG</u>	2128	2128	2128	2128	2128	2128	2128	2128
<u>COLADD</u>	2497	385	4169	8767	4895	1980	11176	1584
<u>MATPLG</u>	988	37	1291	4015	1754	330	2522	156
<u>LIBID</u>	5985	168	7574	26292	10773	1253	9870	434
<u>ROWPOS</u>	1710	48	2164	7512	3078	358	2820	124
<u>BLKSIZ</u>	1330	130	2090	2590	2150	1510	11120	940
<u>BLCCOD</u>	16841	2858	30929	37399	28513	20937	322780	18239
<u>DELCOI</u>	8227	233	9341	34177	14055	2431	25692	1024
<u>RESCOD</u>	18594	91323	53880	20752	42014	93847	19845	181353
<u>TOTAL</u>	59342	98846	118600	148417	111115	126937	412535	207921
<u>COMP.</u> <u>RATIO</u>	62.0	37.8	31.8	28.3	33.1	29.0	8.9	17.7

TABLE IV
COMPRESSION RATIOS FOR CODING OF CCLTT DOCUMENT SET WITH VARIOUS CODING ALGORITHMS

CCITT DOCUMENT	CCITT 1-D	READ K=4	READ K=32	IBM K=4	IBM K=32	RP ² K=4	SPO K=32	CSH E.P. K=32	CSH V.H.P. K=32
1	15.2	21.8	24.9	20.6	23.2	20.6	23.2	54.5	62.0
2	15.1	28.7	38.0	24.1	29.1	25.7	32.6	37.2	37.8
3	8.7	13.6	16.3	13.2	15.6	13.3	15.8	27.7	31.8
4	5.3	6.6	7.1	6.6	7.2	6.5	7.0	22.4	25.3
5	8.8	12.7	14.7	12.3	14.1	12.4	14.3	28.7	33.1
6	10.2	19.0	25.1	17.6	22.5	18.1	23.5	26.8	29.0
7	4.8	6.1	6.7	6.1	6.6	6.1	6.6	7.4	8.9
8	7.9	15.1	20.2	12.9	15.8	14.0	18.2	16.8	17.7



COMPRESSION LABS, INC.

August 15, 1978

August 15, 1978

Telecommunications Manager
International Company
1111 Broadway
New York, N.Y. 10022

Telecommunications Manager
International Company
1111 Broadway
New York, N.Y. 10022

Dear Mr. Manager:

Dear Mr. Manager:

This letter will act as the standard for determination of the minimum compression ratios acceptable for the FAX-COMP, facsimile data compressor. The floppy disk of the FAX-COMP will be able to store at least nine copies of this page prior to overflowing which will guarantee a transmission time of less than 25 seconds for the page. This transmission time will be achievable using a 2400 baud digital modem for line connection.

This letter will act as the standard for determination of the minimum compression ratios acceptable for the FAX-COMP, facsimile data compressor. The floppy disk of the FAX-COMP will be able to store at least nine copies of this page prior to overflowing which will guarantee a transmission time of less than 25 seconds for the page. This transmission time will be achievable using a 2400 baud digital modem for line connection.

Compression ratios of from 5:1 up to 25:1 can be expected from other pages of information, depending upon the actual content of the pages. These compression ratios are defined when using the 96 line per inch scanning resolution only.

Compression ratios of from 5:1 up to 25:1 can be expected from other pages of information, depending upon the actual content of the pages. These compression ratios are defined when using the 96 line per inch scanning resolution only.

Very truly yours,

Clyde E. Marvin

CLOYD E. MARVIN
Vice President
Marketing

Very truly yours,

CLOYD E. MARVIN
Vice President
Marketing

01vrg

(a) ORIGINAL

01vrg

(b) REPRODUCTION

Fig. 12. Example of document compression in CSM mode.

twice that obtained with two-dimensional run-length coding and about the same for graphics-predominate documents.

The CSM system can be operated in a pure symbol recognition mode in which a document is coded by recognition of its alphanumeric symbols. Compression ratios greater than 250:1 can be achieved on business letters in this mode of operation.

REFERENCES

- [1] R. B. Arpa, "Binary image compression," in *Image Transmission Techniques*, W. K. Pratt, Ed. New York: Academic Press, 1979.
- [2] H. G. Musmann and D. Preuss, "Comparison of redundancy reducing codes for facsimile transmission of documents," *IEEE Trans. Commun.*, vol. COM-25, no. 11, pp. 1425-1433, Nov. 1977.
- [3] R. N. Ascher and G. Nagy, "A means for achieving a high degree of compaction on scan-digitized text," *IEEE Trans. Comput.*, vol. C-23, pp. 1174-1179, Nov. 1974.
- [4] W. K. Pratt, W. Chen, and C. Reader, "Block character coding," in *Proc. Soc. Photo-Optical Instrumentation Engineers*, vol. 66, pp. 222-228, Aug. 1976.
- [5] W. Chen, J. L. Douglas, and R. D. Wideryea, "Combined symbol matching—A new approach to facsimile data compression," in *Proc. Soc. Photo-Optical Instrumentation Engineers*, vol. 199, pp. 2-9, Aug. 1978.
- [6] W. Chen, J. L. Douglas, W. K. Pratt, and R. H. Wallis, "A dual mode hybrid compressor for facsimile images," in *Proc. Soc. Photo-Optical Instrumentation Engineers*, Aug. 1979.
- [7] J. L. Mitchell and G. Goertzel, "Two-dimensional facsimile coding scheme," in *Proc. Int. Communications Conf.*, pp. 8.7.1-8.7.5, 1979.
- [8] "Proposal for draft recommendation of two-dimensional coding scheme," Rep. CCITT Study Group XIV, contribution no. 42, prepared by Japan.
- [9] "Proposal for optional two-dimensional coding scheme for group 3 facsimile apparatus," Rep. CCITT Study Group XIV, contribution no. 77, prepared by British Post Office, Mar. 1979.

END

FILMED

3-84

DTIC