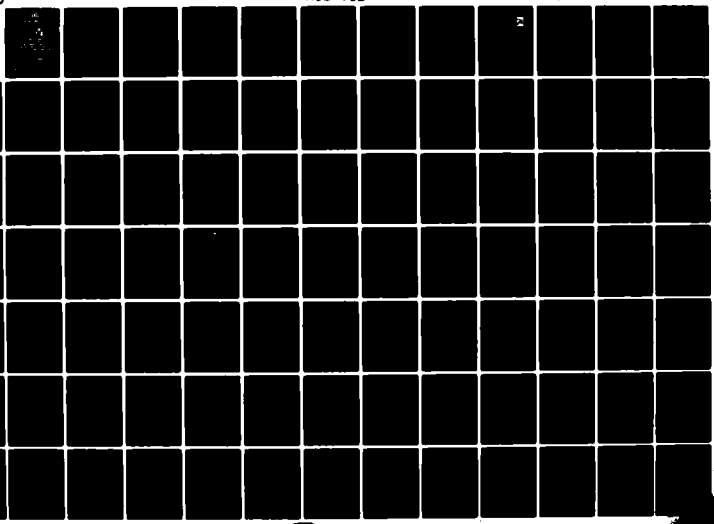
AD-A092 395 DELTA INFORMATION SYSTEMS INC JENKINTOWN PA F/G 17/2
MEASUREMENT OF COMPRESSION FACTOR AND ERROR SENSITIVITY FACTOR --ETC(U)
AUG 80 S URBAN, R SCHAPHORST DCA100-80-W-0145
UNCLASSIFIED NCS-TIB-80-6 NL

1 OF 2
AD
A092395



AD A092395

LEVEL II

(12) (18) NCS TIB- 80-6

NATIONAL COMMUNICATIONS SYSTEM

DTIC
ELECTE
DEC 3 1980
S C

(9) TECHNICAL INFORMATION BULLETIN

80-6

(6) MEASUREMENT OF COMPRESSION FACTOR AND ERROR SENSITIVITY FACTOR OF THE MODIFIED READ FACSIMILE CODING TECHNIQUE

(10) Stan Urban
Richard Schaphorst

(15) DCA100-80-M-0145

(11) AUG[redacted]80

(12) 198

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED

DDC FILE COPY

411214

80 11 25 024

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER NCS TIB 80-6	2. GOVT ACCESSION NO. AD-A092 395
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) Measurement of Compression Factor and Error Sensitivity Factor of the Modified READ Facsimile Coding Technique.	5. TYPE OF REPORT & PERIOD COVERED Final
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Steve Urban Richard Schaphorst	8. CONTRACT OR GRANT NUMBER(s) DCA100-80-M-0145
9. PERFORMING ORGANIZATION NAME AND ADDRESS Delta Information Systems, Inc. 259 Wyncote Road Jenkintown, PA 19046	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS National Communications System Office of Technology and Standards (NCS-TS) Washington, D.C. 20305	12. REPORT DATE August 1980
	13. NUMBER OF PAGES 95
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)

Distribution unlimited; approved for public release

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)

Image Coding	Error Sensitivity	Image Statistics
Digital Facsimile	Two-dimensional Coding	CCITT Standards
Facsimile Coding	Coding Algorithms	
Compression Factor	Computer Simulation	
Modified READ Code	Modified Huffman Code	

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

This Technical Information Bulletin (TIB) describes the measurement of Compression factor and error sensitivity factor of the Modified READ two-dimensional facsimile coding technique. The TIB contains detailed flow charts and computer programming associated with this coding scheme. Compression factor and error sensitivity together with statistical data have also been tabulated. This TIB is a companion document to NCS TIB's 79-7, 79-8, 79-9, and 79-10.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

2

MEASUREMENT OF COMPRESSION FACTOR AND ERROR SENSITIVITY FACTOR OF THE MODIFIED READ FACSIMILE CODING TECHNIQUE

August, 1980

DTIC
ELECTE
DEC 3 1980

FINAL REPORT

Submitted to:

NATIONAL COMMUNICATIONS SYSTEMS
8th & S. COURTHOUSE RD.
ARLINGTON, VIRGINIA 22204

Contracting Agency:

DEFENSE COMMUNICATIONS AGENCY

Purchase Order: DCA 100-80-M-0145

Submitted by:

DELTA INFORMATION SYSTEMS, INC.

259 WYNCOTE ROAD

JENKINTOWN, PENNA. 19046

NCS TECHNICAL INFORMATION BULLETIN 80-6

MEASUREMENT OF COMPRESSION FACTOR AND ERROR SENSITIVITY FACTOR OF THE MODIFIED READ FACSIMILE CODING TECHNIQUE

AUGUST 1980

PROJECT OFFICER:

DENNIS BODSON
Senior Electronics Engineer
Office of NCS Technology
and Standards

APPROVED FOR PUBLICATION:

MARSHALL L. CAIN
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 which is an element of the overall GSA Federal Standardization 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 System
ATTN: NCS-TS
Washington, D.C. 20305
(202) 692-2124

MEASUREMENT OF COMPRESSION FACTOR AND ERROR SENSITIVITY FACTOR

OF THE MODIFIED READ FACSIMILE CODING TECHNIQUE

1.0 Introduction 1-1

2.0 Measurement Parameters 2-1

3.0 Computer Program Overview 3-1

3.1 The Simulation Process 3-1

3.2 Program Structure 3-5

4.0 Error Detection and Correction Procedure 4-1

5.0 Measurement Results 5-1

6.0 References 6-1

APPENDICES

A. Draft Recommendation (T.4) for Standardization of Group 3 Facsimile Apparatus for Document Transmission A-1

B. Flow Charts for Modified READ Program B-1

C. Code Listing for Modified READ Program C-1

Accession For
NTIS GRA&I ☑
DTIC TAB ☐
Unannounced ☐
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A

1.0 INTRODUCTION

Members of the CCITT XIV COM Study Group met in Geneva in May 1980 to finalize a standard two-dimensional code for the Group 3 facsimile apparatus. A standard known as the Modified READ code was chosen and is described in Section 4.2 of Appendix A. The National Communication System, an organization of the U. S. Government, awarded a contract (Purchase Order DCA 100-80-M-0145) to Delta Information Systems, Inc. to measure the compression and error sensitivity of the Modified READ code. This document is the final report summarizing the work performed under this study. This work is a logical continuation of three prior investigations (see References 3, 6, 12) of facsimile coding techniques performed by Delta Information for the National Communication System.

The measurement parameters which were involved in the investigation are summarized in Section 2.0 of this report. Section 3.0 describes the overall simulation process and hierarchal structure of the Modified READ computer program. One of the more important aspects of this study is the simulation of the Modified READ coding algorithm in the presence of noise. For the simulation to be complete it is necessary to fully define the operation of the decoder when an error occurs. This error processing function is described in Section 4.0. The Modified READ computer program was run on the Computer Facility at the Defense Communications Engineering Center in Reston, Virginia. A total of twenty-two runs were made on the computer at different combinations of CCITT document number, resolution, K-factor, minimum scan line time, error phase, and transmission error file. Section 5.0 summarizes the results of these simulation runs. In several instances the test results are compared with the results of previous simulations of one dimensional and two dimensional coding techniques.

Appendices B and C contain the flow charts and code listing for the Modified READ code prepared on this program.

Delta Information Systems wishes to acknowledge the Contracting Officer's Technical Representative, Dennis Bodson, for the support he has provided during the course of this contract. The assistance of Marla Thomas, from the DCEC computer facility, is also greatly appreciated.

2.0 MEASUREMENT PARAMETERS

In this section, the various parameters involved in the measurement of compression and error sensitivity will be summarized. In general, Study Group XIV of the CCITT agreed upon these measurement parameters at the general meeting held in Geneva in December 1978 (see Reference 2).

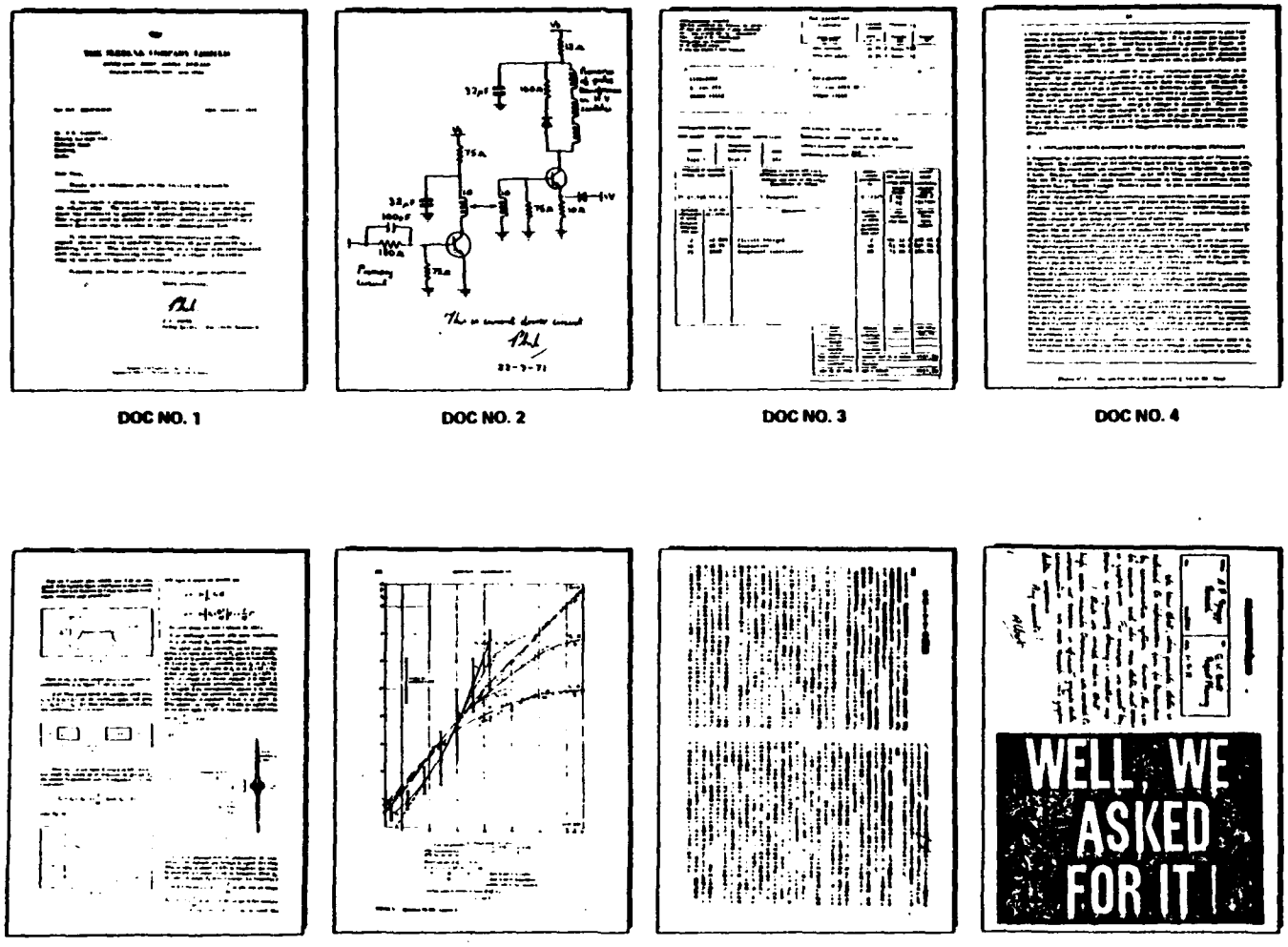
2.1 Test Documents

The test documents were chosen from the eight CCITT test documents (see Figure 2-1) since they have been widely used by data compression experimenters in the past. Documents numbered 1, 4, 5, and 7 (see Figures 2-2, 2-3, 2-4, and 2-5 respectively) were selected as the standard test images since these were considered most representative of documents to be transmitted.

The French PTT Administration has scanned the eight CCITT documents at the high resolution specified for Group 3 machines--7.7 lines/mm. They have also quantized each pel to be either black or white and stored the resultant image on magnetic tape. This tape was used as the source of input documents in this simulation program. Appendix B of Reference 3 describes the format of the test document magnetic tape supplied by the French PTT.

2.2 Resolution

It was agreed that measurements would be performed at both standard resolution (3.85 lines/mm.) and high resolution (7.7 lines/mm.). In the high resolution case, all lines on the input test documents shall be used. In standard resolution tests, every odd scan line



DOC NO. 1 DOC NO. 2 DOC NO. 3 DOC NO. 4

DOC NO. 5 DOC NO. 6 DOC NO. 7 DOC NO. 8

Figure 2-1 CCITT Standard Test Documents

THE SLEREXE COMPANY LIMITED

SAPORS LANE · BOOLE · DORSET · BH 25 8 ER

TELEPHONE BOOLE (945 13) 51617 · TELEX 123456

Our Ref. 350/PJC/EAC 18th January, 1972.

Dr. P.N. 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

Figure 2-2 CCITT Test Document No. 1

Registered in England: No. 2038
Registered Office: 60 Vicars Lane, Ilford, Essex.

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet, et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 425 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étaient récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen ; ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger, et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gérera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie Internationale pour l'Informatique a été installé à Toulouse en février ; la même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux,

Figure 2-3 CCITT Test Document No. 4

Photo n° 1 - Document très dense lettre 1,5mm de haut -

Restitution photo n° 9

Cela est d'autant plus valable que $T\Delta f$ est plus grand. A cet égard la figure 2 représente la vraie courbe donnant $|\phi(f)|$ en fonction de f pour les valeurs numériques indiquées page précédente.

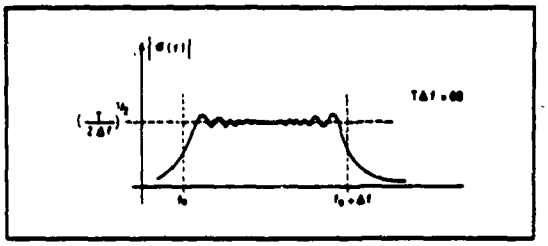


FIG. 2

Dans ce cas, le filtre adapté pourra être constitué, conformément à la figure 3, par la cascade :

— d'un filtre passe-bande de transfert unité pour $f_0 \leqslant f \leqslant f_0+\Delta f$ et de transfert quasi nul pour $f < f_0$ et $f > f_0+\Delta f$, filtre ne modifiant pas la phase des composants le traversant ;

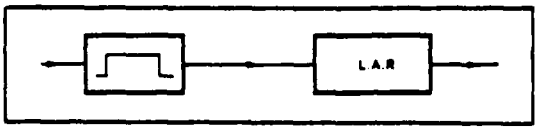


FIG. 3

— filtre suivi d'une ligne à retard (LAR) dispersive ayant un temps de propagation de groupe T_R décroissant linéairement avec la fréquence f suivant l'expression :

$$T_R = T_0+(f_0-f)\frac{T}{\Delta f} \quad \text{(avec } T_0 > T\text{)}$$

(voir fig. 4).

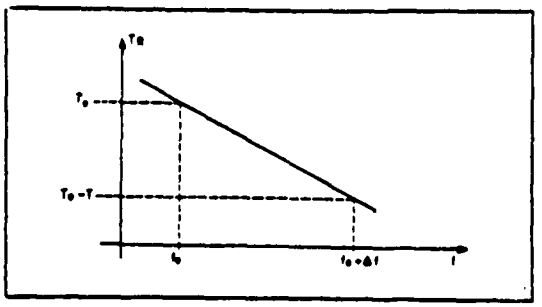


FIG. 4

telle ligne à retard est donnée par :

$$\varphi = -2\pi\int_0^f T_R\,df$$

$$\varphi = -2\pi\left[T_0+\frac{f_0T}{\Delta f}\right]f+\pi\frac{T}{\Delta f}f^2$$

Et cette phase est bien l'opposé de $\underline{/\phi(f)}$, à un déphasage constant près (sans importance) et à un retard T_0 près (inévitable).

Un signal utile $S(t)$ traversant un tel filtre adapté donne à la sortie (à un retard T_0 près et à un déphasage près de la porteuse) un signal dont la transformée de Fourier est réelle, constante entre f_0 et $f_0+\Delta f$, et nulle de part et d'autre de f_0 et de $f_0+\Delta f$, c'est-à-dire un signal de fréquence porteuse $f_0+\Delta f/2$ et dont l'enveloppe a la forme indiquée à la figure 5, où l'on a représenté simultanément le signal $S(t)$ et le signal $S_1(t)$ correspondant obtenu à la sortie du filtre adapté. On comprend le nom de récepteur à compression d'impulsion donné à ce genre de filtre adapté : la « largeur » (à 3 dB) du signal comprimé étant égale à $1/\Delta f$, le rapport de compression est de $\dfrac{T}{1/\Delta f} = T\Delta f$

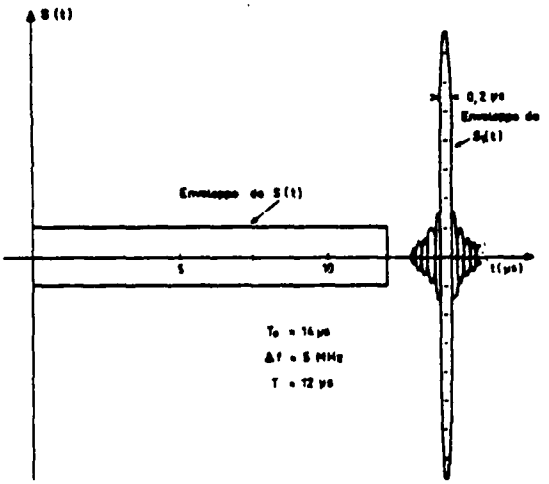


FIG. 5

On saisit physiquement le phénomène de compression en réalisant que lorsque le signal $S(t)$ entre dans la ligne à retard (LAR) la fréquence qui entre la première à l'instant 0 est la fréquence basse f_0 qui met un temps T_0 pour traverser. La fréquence f entre à l'instant $t = (f-f_0)\dfrac{T}{\Delta f}$ et elle met un temps $T_0-(f-f_0)\dfrac{T}{\Delta f}$ pour traverser, ce qui la fait ressortir à l'instant T_0 également. Ainsi donc, le signal $S(t)$

Figure 2-4 CCITT Test Document No. 5

CCITTの概要

沿革

CCITTは、国際電気通信連合（ITU）の四つの常設機関（事務総局、国際周波数登録委員会、CCIR、CCITT）の一つとして、ITUの中でも、世界の国際通信上の諸問題を真先に取上げ、その解決方法を見出して行く重要な機関である。日本名は、国際電信電話諮問委員会と称する。

CCITTの前身は、CCIF（国際電話諮問委員会）とCCIT（国際電信諮問委員会）である。CCIFは、1924年にヨーロッパに「国際長距離電話通信諮問委員会」が設置され、これが1925年のパリ電信電話会議のとき、正式に、「国際電話諮問委員会」として万国電信連合の公式機関となったものである。CCITは、同じく1925年の会議のとき、CCIFと併立する　）として設置された。

そして、CCIFは、1956年の12月に第18回総会が開催されたのち、CCITは、同年同月に第8回総会が開催されたのち、併合されて現在のCCITTとなった。このCCITTは、CCIFとCCITが解散した直後、第1回総会を開催し、第2回総会は、1960年にニューデリーで、第3回総会は、1964年、ジュネーブで、第4回総会は、1968年、アルゼンチンで開催された。

CCIFとCCITが合併したのは、有線電気通信の分野、とくに伝送路について電信回線と電話回線とを技術的に分ける意味がなくなってきたこと、各国とも大体において、電信部門と電話部門は同一組織内にあること、CCIFの事務局とCCITの事務局の合併による能率増進等がおもな理由であった。

CCITTは、上述のように、ヨーロッパ内の電信・電話の技術・運用・料金の基準を定め、あるいは統一をはかってきたので、現在でも、その影響を受け、会合参加国は、ヨーロッパの国が多く、ヨーロッパで生起する問題の研究が多い。たとえば、1960年のCCITT勧告の中で、技術上配慮する距離は約2,500kmであったが、これはヨーロッパ内領域を想定したものである。

しかしながら、1956年9月に敷設された大西洋横断電話ケーブルは、大陸間電話通信の自動化および半自動化への技術的可能性を与え、CCITTがこの問題を取り上げるに及び、CCITTの性格は漸次、汎世界的色彩を実質的に帯びるに至った。この汎世界的性格は第2次世界大戦後目ざましくなったアジア・アフリカ植民地の独立に伴ってITUの構成員の中にこれらの国が加わり、ITUの中に新しい意見が導入されたことにも起因して、技術面、政治面の双方から導入されてきた。CCITTの汎世界化は、1960年の第2回総会がニューデリーで開催されたことにもあらわれている。この総会までは、CCIT、CCIFのいずれにしろ、アメリカやアジアで総会が開催されたことがなく、CCITT委員長も、ニューデリー総会の準備文書で、この点には注目すべきであるとのべている。

任務

ITUは、全権委員会議、主管庁会議を始めとして、七つの機関をもち、それぞれの機関の権限と任務は国際電気通信条約に明記されている。そこで条約を参照してみるならば、CCITTの任務は、つぎのとおりとなっている。

「国際電信電話諮問委員会(CCITT)は、電信および電話に関する技術、運用および料金の問題について研究し、および意見を表明することを任務とする。」（1965年モントルー条約第187号）

「各国際諮問委員会は、その任務の遂行に当たって、新しい国または発展の途上にある国における地域的および国際的分野にわたる電気通信の創設、発達および改善に直接関連のある問題について研究し、および意見を作成するように妥当な注意を払わなければならない。」（同第188号）

「各国際諮問委員会は、また、関係国の要請に基づき、その国内電気通信の問題について研究し、かつ、勧告を行なうことができる。」（同第189号）

上記第187号と第188号にいわれる「意見」とは、フランス語の Avis から訳したもので、英語では、「勧告(Recommendation)」となっている。CCITTの表明する意見は、国際法的には強制力をもたないものであって、この点が、条約、電信規則、電話規則等各国を拘束する力をもっているものと異なる。もっとも意見とは称しても、技術的分野では、電信規則のごとき、各国政府が承認してその内容を実施する強制規則をもたないので、実際にある機器の仕様を定める場合には、多くの国の意見が統一されたこの「意見」に従わなければ、円滑な国際通信を行なうことができない場合が多い。この意見（または勧告）は、国際通信を行なう場合各国が直面する問題について、具体的意見を表明するもので、たとえば、大陸間ケーブルで大陸間通話を半自動化しようとする場合、その信号方式や取り扱う通話の種類および料金は、どのようにするかを研究して意見を表明する。したがって、CCITTの活動は、つねに時代の最先端を行くもので、CCITTの活動方向は、そのまま世界の国際通信の活動方向であるともいえる。

この意見は、また、電信規則以下のその他の規則のごとく、数年以上の間隔をもって開催される主管庁会議というような大会議の決定をまたなくても表明することができ、また、その改正も容易であるので、現在のように進歩の早い国際通信界では、関係国の意見を統一した国際的見解としては非常に便利である。

Figure 2-5 CCITT Test Document No. 7

should be used. Figure 2-6 is a copy of the French PTT Test Document No. 4 scanned with 7.7 lines/mm. resolution. Figure 2-7 is a copy of the same document where the even scan lines have been replaced with the line above. Therefore, this represents a document in which the vertical resolution is 3.85 lines/mm.

2.3 Minimum Scan Line Time (MSLT)

The standard MSLT to be used in the measurement program will be 5, 10, and 20 ms. with EOL ie and 0 ms. without EOL-code. It was later clarified in a memo from the chairman of the Working Committee (see Reference 7) that if, for reasons of test economy, only one value of MSLT can be used in the test program, that value shall be 20 ms.

2.4 Transmission Bit Rate

The standard transmission bit rate is 4800 bits/sec.

2.5 Measurement of Compression

Two standard measures of compression have been established--(1) number of coded bits (2) Compression Factor. The number of coded bits is the number of bits required to transmit a document, including all overhead bits such as End of Line (EOL) and Fill bits. The Compression Factor is computed by dividing the total number of picture elements (pels) per test document by the number of coded bits. It was further agreed that the Compression Factor and coded bits should be computed for two different conditions--with overhead and without overhead. The measurement with overhead applies to the

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet, et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 425 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étaient récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen ; ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles, voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger, et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gèrera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie Internationale pour l'Informatique a été installé à Toulouse en février ; la même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux.

Figure 2-6 Test Document Scanned/Printed 7.7 lines/mm.

Photo n° 1 - Document très dense lettre 1,5mm de haut -

Restitution photo n° 9

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet, et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 425 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étaient récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen ; ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles, voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger, et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gèrera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie Internationale pour l'Informatique a été installé à Toulouse en février ; la même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux.

Figure 2-7 Test Document Scanned 7 lines/mm. Printed 3.85 lines/mm.

Photo n° 1 - Document très dense lettre 1,5mm de haut -

Restitution photo n° 9

Group 3 situation while the measurement without overhead applies to the Group 4 case.

2.6 Measurement of Error Sensitivity

An objective measure of error sensitivity is obtained by encoding the test documents with the proposed techniques (all overhead bits must be included), subjecting the resulting bit stream to transmission errors, decoding the transmission to obtain the received image, and comparing the original image with the received image to determine the number of pels in error. The Error Sensitivity Factor (ESF) is calculated as the total number of document pels in error divided by the total number of transmission bits that are in error. In this way, the ESF represents the average disturbance to the output image caused by a single transmission error.

2.6.1 Transmission Error Pattern

It was agreed that a record of actual bit errors incurred over telephone lines will be used in the error sensitivity test. The Federal Republic of Germany (see Reference 8) has obtained a record of such errors by transmitting a known psuedo-random sequence at 4800 bits/sec. using a V27 ter modem over a switched telephone network. The resultant error pattern has been recorded on magnetic tape and made available to experimenters. Appendix C of Reference 3 describes the format of the transmission error magnetic tape. This tape was used in the measurement of error sensitivity described in this report.

2.6.2 Error Phases

One concern with the ESF measurement is the high degree of sensitivity to those few errors which may affect the end of line code and can cause an inordinate number of incorrect pels. If the error pattern happened to fall in an unfortunate phase relative to the encoded bits, a large number of pels could be affected. On the other hand, the error pattern could fall fortuitously and affect a relatively few number of pels. To insure experimenters can achieve an adequate level of statistical validity, the concept of error phases has been introduced. In the basic zero phase, the first bit of the error record is aligned with the first bit of the encoded transmission. In the case of Phase 2, the transmitted bit information is delayed by 1,024 bits relative to the previous run. The transmission bit information is delayed by 2,048 bits for Phase 2. Experimenters would have a higher confidence level in the average of the three phases compared to any one ESF taken alone.

2.6.3 Error Correction

In order to precisely measure the error sensitivity, both the encoding technique and the decoding algorithm must be completely defined. If more than one decoding algorithm is proposed (for example, to achieve differing levels of error control), each must be tested separately. Collective Letter No. 87 from the CCITT (see Reference 7) outlines an error correction procedure to be used for simulating two-dimensional algorithms where an error correction procedure has not been otherwise specified. In this procedure, the erroneous line is replaced

by the previous line and following lines are replaced by white lines until a one-dimensional coding line is correctly decoded.

3.0 COMPUTER PROGRAM OVERVIEW

This section contains a general overview of the computer program architecture used under this contract. The description is divided into two parts. Section 3.1 focuses on the overall simulation process from a flow perspective with particular emphasis on the simulation inputs and outputs. Section 3.2 presents the hierarchical structure of the programs illustrating the structure and organization of the main program and subroutines. For convenience of the reader, a detailed flow chart, and the actual program code listing, has been included in Appendices B and C. All computer programs have been written in conventional Fortran IV language.

3.1 The Simulation Process

Figure 3-1 illustrates the interrelationship between the major functions of each simulation program developed on the subject contract. There are two input data sets to each simulation which originate on magnetic tape. One tape, supplied by the French PTT Administration, contains all eight of the CCITT test documents. The format of this input image tape is described in Appendix B of Reference 3. The other tape, supplied by the Federal Republic of Germany, contains transmission error data from actual switched telephone circuits. The format of this input tape is described in Appendix C of Reference 3. A program called "REDTAP" was prepared to read the data from the input document tape while the error tape is read in directly. Data from the two input tapes are placed on disc in the computer system to be accessed during the simulation process. A separate file is established for each of the

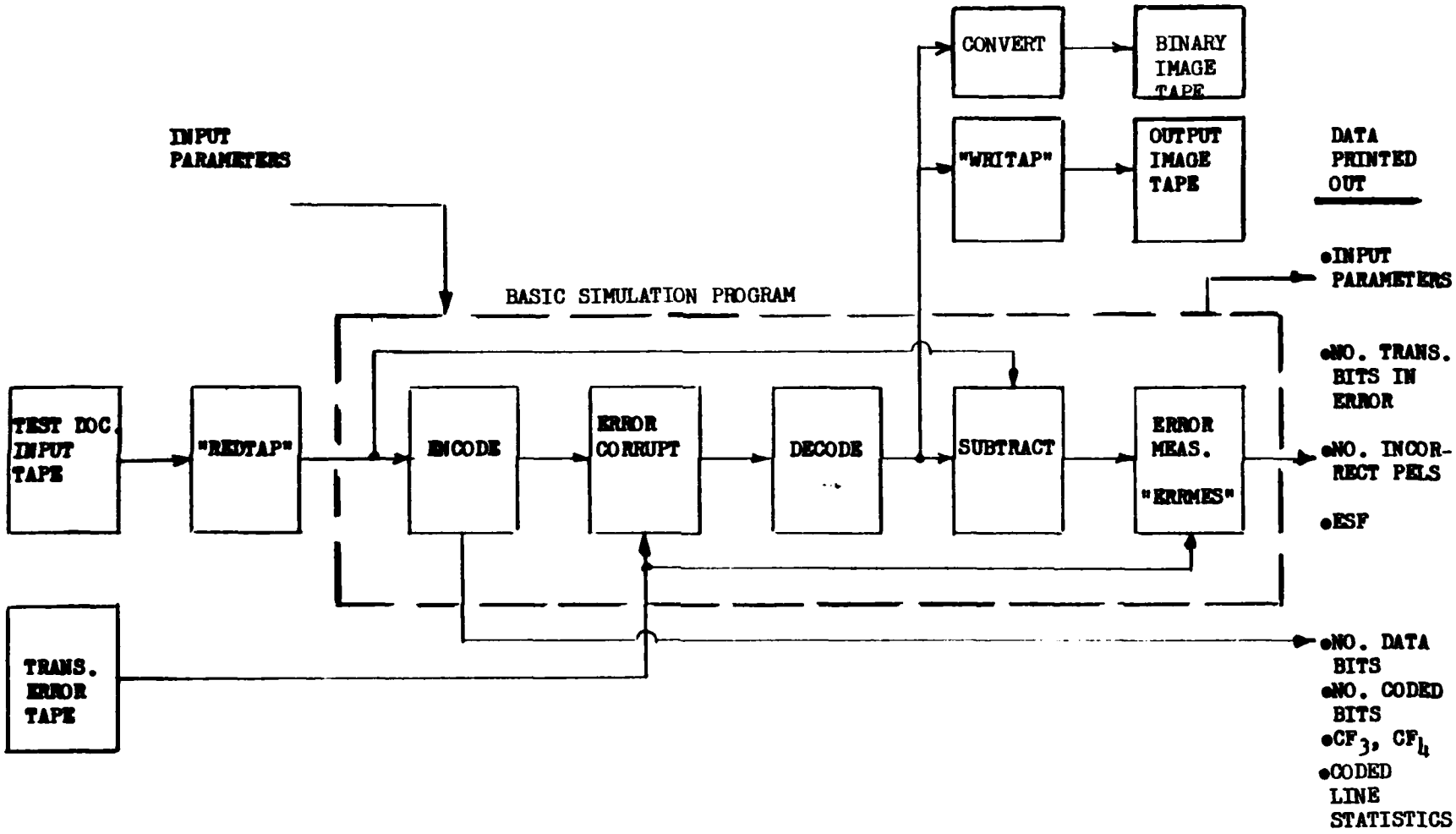


FIGURE 3-1 DIAGRAM OF THE SIMULATION PROCESS

test documents. The transmission error tape is divided into four files, one for each of four different circuit error conditions.

To initiate the simulation process, the operator must type in a set of input parameters. The insertion of the input parameters is accomplished on an interactive basis with prompting. A typical interactive sequence with responses is listed below.

1. PARAMETERS: INPUT (=I), OR DEFAULT (=D)? I
2. DIAGNOSTIC PRINTOUT? (Y OR N). N
3. ENTER MAXIMUM NUMBER OF PELS PER LINE: 1728
4. ENTER VERTICAL SAMPLING: 1
5. ENTER PARAMETER K: 4
6. ENTER ERROR PATTERN PHASE: 0
7. ENTER MINIMUM COMPRESSED LINE LENGTH: 96
8. NUMBER OF SCAN LINES TO BE PROCESSED = ? 10
9. ERROR MODE = ? (M=MANUAL, T=TAPE, N=NO ERRORS) N

After the data has been entered and the measurement parameters have been selected, the first step in the simulation process is the "ENCODE" function. This function detects color changes in the input data and constructs the appropriate code word by table look-up or algorithm. The actual code is fed to the error corrupt unit, while the number of code bits is accumulated with fill and EOL codes to provide the output total number of data bits, to compute the Compression Factors, CF_3 and CF_4.

The error corruption step combines the transmission error data with the encoded data. At each point in the image where an error occurs, the corresponding bit in the encoded signal is reversed and fed to the

decode function. The decoder basically performs the inverse function of the encoder, generating a series of lines of image pels. There are two parts of the decoding function which are not obvious and require clarification: (1) what the decoder does when an error occurs (2) what the decoder does when a line is missing. The operation of the decoder under these two conditions is described in Section 4.

The output of the Decode function feeds the "WRITAP" or "CONVERT" functions for writing the error corrupted image on magnetic tape. It is also fed to a subtraction function which compares the decoded image with the original image. Pels which are in error are fed to the "ERRMES" subroutine which counts all the pels in the image which are in error. This subroutine also counts the number of transmission error bits which corrupted the encode signal. Finally, the "ERRMES" subroutine computes the ESF by dividing the number of incorrect pels by the number of transmitted bits in error.

Figure 3-1 shows that the simulation process provides a printout of all the computed performance data as well as a summary tabulation of the input parameters.

For more details on the computer programs, refer to Section 3.2 for a description of the program structure and to the Appendices for flow charts and program listings.

The reader should note that most of the software prepared under this contract is suitable for simulating any compression algorithm. The only subroutines which must be written specifically for a particular coding technique are the encode and decode subroutines.

3.2 Program Structure

The following section describes the structure of the computer program written to simulate the Modified READ code. The program conforms to the general structure shown in Figure 3-2 which illustrates the hierarchy of the functions/subroutines that make up each simulation program. A brief description of each of the functions/subroutines follows:

MODREAD

The MODREAD program controls the decoding process and the error recovery procedure for getting back in sync when an error is detected. As can be seen from Figure 3-2, the simulation process is "decode driven;" that is, the main program controls the decode process which decodes a buffered line of compressed data. When the contents of the buffer have been used up, a new line of data is encoded. The MODREAD program also controls parameter input, measurement of errors, and reports computed results.

GETLE

The GETLE subroutine retrieves a number of requested bits from the coded line and delivers the bits packed into a word (right justified). End-of-line codes (EOL) or line synchronization signals (LSS) are detected. If the number of coded bits requested by the calling program is not available, the ENCDE subroutine is called to provide them.

ENCDE

This subroutine supplies a line of compressed data. Color transitions on an input line are detected bit-by-bit. Both one-dimensional and two-dimensional lines are encoded depending on the parameter K. The code word is generated by table look-up, or algorithm, as appropriate, and added to the coded line buffer via CODELN and/or CODENG.

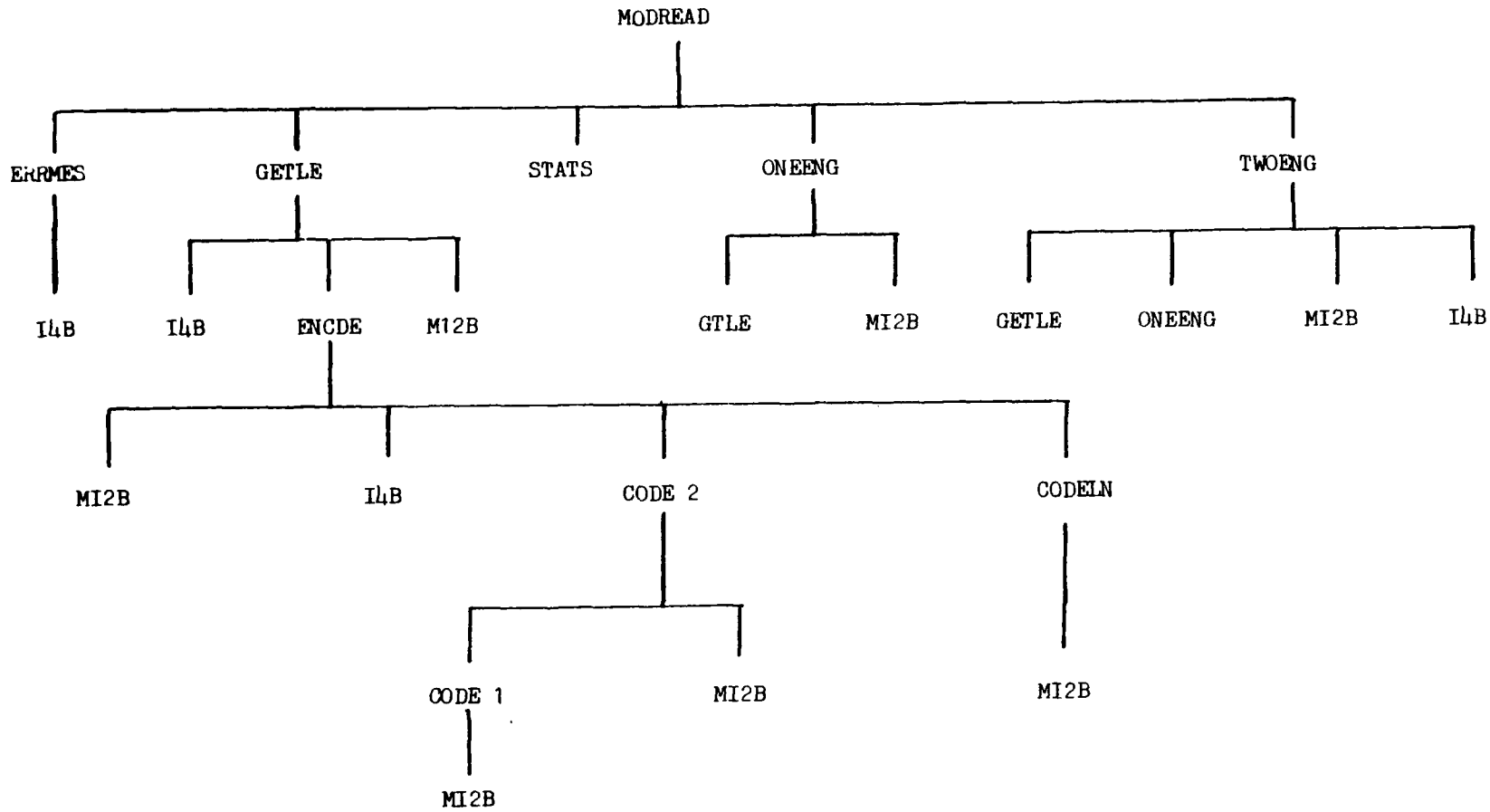


FIGURE 3-2 SUBROUTINE HIERARCHY

CODELN

The subroutine CODELN is called by ENCDE to look up the Modified Huffman Code (MHC) corresponding to a given run length and color, and add the code word to the coded line buffer.

CODENG

The subroutine CODENG performs a similar function for the two-dimensional case. Based on a particular feature, the appropriate code word is generated by table look-up or algorithm and added to the coded line buffer. All code tables for both one-dimensional and two-dimensional codes are stored in labelled common which is initialized by a BLOCK DATA subprogram.

ONEENG

The ONEENG subroutine decodes the MHC. It extracts a set of n bits (n=3 initially) from the coded line and looks for a match with all code words of length n, increasing n until a match is found or the code table is exhausted. When and if a match is found, the indicated bits are constructed on the output line. Any errors detected in the decoding process, such as no match to code table, or line too long, are flagged.

TWOENG

This subroutine performs the same function as ONEENG for the two-dimensional line.

MI2B and I4B

The subprograms MI2B and I4B are used to pack and unpack a set of bits into (or from) an array of words.

4.0 Error Detection/Correction Procedure

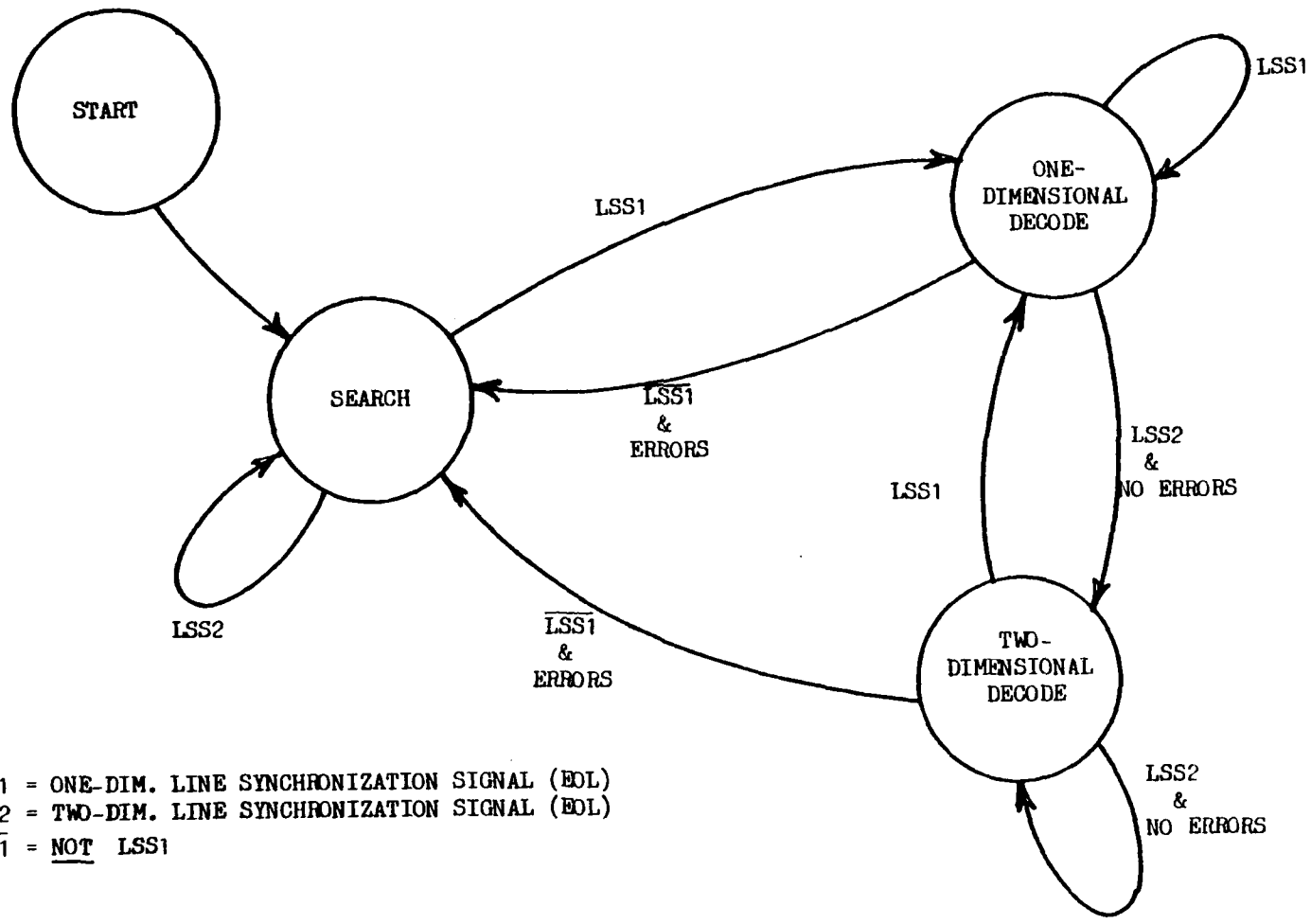
In Reference 7.0, the following error checking and processing procedure was specified by the CCITT for testing the proposed two-dimensional coding techniques:

1) Error checking - If decoded signals are not exactly 1728 pels/line, the line is recognized as an erroneous line.
2) Error processing - The erroneous line is replaced by the previous line and following lines are replaced by white lines until one-dimensional coding line is correctly decoded.

The error detection and correction procedures used in this simulation follow the spirit, if not the letter, of this directive. The error checking was expanded to include the detection of any condition that could not possibly occur in a correctly received transmission. Some examples of possible error conditions are:

- EOL occurs before 1728 pels have been written
- More than 2728 pels have been written before EOL is received
- No word in applicable code table matches received bit pattern
- Current line decoding references a run that does not exist in the previous line
- Pels are written to the left of the first pel on the line

Upon detection of an error condition, the decoder attempts to re-synchronize by searching for the next unique Line Synchronization Signal (LSS). The state diagram for error recovery for these algorithms is shown in Figure 4-1.



LSS1 = ONE-DIM. LINE SYNCHRONIZATION SIGNAL (EOL)
LSS2 = TWO-DIM. LINE SYNCHRONIZATION SIGNAL (EOL)
$\overline{\text{LSS1}}$ = NOT LSS1

Figure 4-1 DECODE STATE DIAGRAM

Because of transmission errors, some of the original image lines may be missing in the output, or additional lines may be in the output that were not in the original image. In order that a missing or extra line not have an undue influence on the ESF, it is important that the original and received images not get permanently out of line alignment when they are compared to determine the number of pel errors. To this end, each of the lines in the original image is assigned a serial line number, and this number continues to be associated with the same line in the received image. If a transmitted line is dropped, due to the loss of an EOL, then its line number will be missing in the output. On the other hand, if a line is broken into two or more lines in the received image, due to false EOL's, then its line number will appear more than once in the output.

If no lines are dropped or added, the line numbers of the original and received lines that are compared to detect pel errors will be equal. When a line is added or deleted, the line numbers of the compared lines will become unequal. When this occurs for the first time, the two lines with different line numbers are compared to determine the number of pel errors, which is added to the pel error total. Then, instead of proceeding to the next line in both the original and received images, the next line is used in only one of the images, with the previous line being used in the other image. The line is advanced only in that image that has the smaller line number, so as to tend to make the line numbers of the two images more equal. This continues until the line numbers are equal, after which the next line is used in both images, until another inequality is detected. This procedure provides a proper penalty for a missing or added line, but prevents this type of error from causing pel errors over the entire image below the place where it occurred.

5.0 MEASUREMENT RESULTS

During the course of the study, Delta Information Systems prepared a computer program to simulate the Modified READ code. The program was run on the Hybrid Computer Facility at the Defense Communications Engineering Center in Reston, Virginia. A total of twenty-two computer runs were performed. Twenty runs were performed at low K-factors (K = 2, 4) and two runs were at an infinite K-factor. The parameters for twenty runs performed at low K-factors are listed in Table 5-1. The test results for the Modified READ code for these twenty runs are provided in Table 5-2. For reference purposes Tables 5-3 through 5-7 are included for the same twenty test conditions for five other candidate two-dimensional coding techniques--AT&T, IBM, 3M, Japan (READ), XEROX. The compression and error sensitivity of the Modified READ code appears very satisfactory relative to the other five candidate techniques.

Table 5-8 summarizes the results of the tests which were performed at an infinite K-factor. Again the performance of the Modified READ code appears to be favorable relative to the seven other two-dimensional codes which were evaluated in previous studies. Table 5-9 summarizes the coded line length statistics for the Modified READ code.

Table 5-10 compares the performance of the Modified READ code with that of the Modified Huffman code which had been simulated in an earlier program. The results in the first and second rows indicate that the improved compression performance of the Modified READ relative to the Modified Huffman is limited for standard resolution and a complex image like CCITT document number 4. However, the improvement increases for high resolution as shown in the third row. The CF_4 for modified READ at an

TABLE 5-1 TABULATION OF TEST RUN PARAMETERS

TEST RUN	TEST DOCUMENT NUMBER	ERROR PHASE	TRANSMISSION ERROR FILE	MIN. SCAN LINE TIME (ms.)	VERTICAL RESOLUTION * (lines/mm.)
1	4	0	1	20	3.85
2	4	0	1	20	7.70
3	4	0	2	20	3.85
4	4	0	2	20	7.70
5	4	0	3	20	3.85
6	4	0	3	20	7.70
7	4	0	4	20	3.85
8	4	0	4	20	7.70
9	4	1	1	20	7.70
10	4	2	1	20	7.70
11	4	0	1	10	3.85
12	4	0	2	10	7.70
13	4	0	3	10	3.85
14	4	0	4	10	7.70
15	1	0	1	20	3.85
16	1	0	1	10	7.70
17	5	0	1	10	7.70
18	5	0	1	20	3.85
19	7	0	1	20	3.85
20	7	0	1	10	7.70

* For Resolution of 3.85 lines/mm K=2

" " " 7.70 " K=4

Table 5-2 Compression/Error Sensitivity Test Results, Modified READ Algorithm

TEST RUN	# CODED BITS	# BITS IN ERROR XMTD	BER $X10^{-3}$	# INCORRECT PELS	# CODED DATABITS	ESF	CF_3	CF_4
1	440,253	362	.822	16,377	392,817	45.2403	4.6629	5.2260
2	727,041	564	.776	33,107	623,547	58.7003	5.6472	6.5845
3	440,253	510	1.16	14,124	392,317	27.6941	4.6629	5.2260
4	727,041	510	.701	18,163	623,547	35.6137	5.6472	6.5245
5	440,253	296	.672	16,993	392,817	57.4088	4.6629	5.2260
6	727,041	682	.938	42,301	623,547	62.0249	5.6472	6.5845
7	440,253	598	1.36	12,688	392,817	21.2174	4.6629	5.2260
8	727,041	793	1.09	29,880	623,547	37.6797	5.6472	6.5845
9	727,041	564	.776	34,223	623,547	60.6791	5.6472	6.5845
10	727,041	564	.776	36,841	623,547	65.3209	5.6472	6.5845
11	417,862	278	.665	11,205	392,817	40.3057	4.9128	5.2260
12	678,525	510	.752	26,738	623,547	52.4274	6.0510	6.5845
13	417,862	290	.694	12,508	392,817	43.1310	4.9128	5.2260
14	678,525	783	1.15	24,105	623,547	30.7854	6.0510	6.5845
15	189,171	120	.634	2,472	115,238	20.6000	10.8519	17.8141
16	253,704	216	.851	10,718	176,770	49.6204	16.1831	23.2264
17	373.798	220	.589	15,480	322,284	70.3636	10.9838	12.7395
18	255,462	216	.846	7,058	210,841	32.6759	8.0359	9.7365
19	420,197	290	.69	8,779	386,889	30.2724	4.8855	5.3061
20	665,755	564	.847	24,990	620,753	44.3085	6.1670	6.6141

TABLE 5-3 TEST RESULTS, AT&T ALGORITHM

TEST RUN	NO. CODED BITS	NO. BITS IN ERROR XMTD	BER $x10^{-3}$	NO. INCORRECT PELS	NO. OF CODED DATA BITS	ESF	CF_3	CF_4
1	466,613	374	.80	19,378	415,034	51.8128	4.3995	4.9463
2	763,481	564	.73	33,756	655,807	59.8511	5.3776	6.2606
3	466,613	510	1.09	13,823	415,034	37.1039	4.3995	4.9463
4	763,481	510	.66	22,773	655,807	44.6529	5.3776	6.2606
5	466,613	326	.69	15,143	415,034	46.4509	4.3995	4.9463
6	763,481	682	.89	40,440	655,807	59.2962	5.3776	6.2606
7	466,613	626	1.34	11,529	415,034	18.4169	4.3995	4.9463
8	763,481	793	1.03	24,628	655,807	31.056	5.3776	6.2606
9	763,481	564	.73	34,127	655,807	60.5089	5.3776	6.2606
10	763,481	564	.73	36,800	655,807	65.2482	5.3776	6.2606
11	443,326	362	.81	15,862	415,034	43.8177	4.6306	4.9463
12	714,016	510	.71	27,731	655,807	54.3745	5.7502	6.2606
13	443,326	296	.66	10,432	414,034	35.2432	4.6306	4.9463
14	714,016	793	1.11	24,958	655,807	31.4729	5.7502	6.2606
15	193,573	132	.68	1,236	112,546	9.3636	10.6051	18.2402
16	258,832	216	.83	8,586	175,159	39.7500	15.8625	23.4400
17	388,419	220	.56	15,238	335,235	69.2636	10.5704	12.2473
18	267,503	220	.82	5,570	220,429	25.3182	7.6742	9.3130
19	451,171	362	.80	9,463	415,929	26.1409	4.5501	4.9356
20	709,814	564	.79	25,074	663,918	44.4574	5.7842	6.1841

TABLE 5-4 TEST RESULTS, IBM ALGORITHM

TEST RUN	# CODED BITS	# BITS IN ERROR XMTD	BER $X10^{-3}$	# INCORRECT PELS	# OF CODED DATABITS	ESF	CF_3	CF_4
1	430,215	346	.80	16,013	383,562	46.2803	4.7717	5.3521
2	727,740	564	.77	30,600	627,122	54.2553	5.6418	6.5469
3	430,215	510	1.18	15,842	383,562	31.0627	4.7717	5.3521
4	727,740	510	.70	18,102	627,122	35.4941	5.6418	6.5469
5	430,215	296	.62	9,004	383,562	30.4189	4.7717	5.3521
6	727,740	682	.93	43,186	627,122	63.3226	5.6418	6.5469
7	430,215	524	1.21	13,925	383,562	26.5744	4.7717	5.3521
8	727,740	793	1.08	24,117	627,122	30.4124	5.6418	6.5469
9	727,740	564	.77	33,899	627,122	60.1046	5.6418	6.5469
10	727,740	564	.77	37,025	627,122	65.6472	5.6418	6.5469
11	407,850	262	.64	11,329	383,562	43.2404	5.0334	5.3521
12	679,677	510	.75	25,028	627,122	49.0745	6.0407	6.5469
13	407,850	278	.68	13,920	383,562	50.0719	5.0334	5.3521
14	679,677	789	1.16	30,120	627,122	38.1749	6.0407	6.5469
15	187,619	120	.63	3,034	115,011	25.2833	10.9417	17.8493
16	255,119	216	.84	10,928	181,740	50.5926	16.0934	22.5912
17	37,810	220	.58	18,908	329,697	85.9454	10.8385	12.4530
18	254,459	216	.84	8,211	210,809	38.0139	8.0676	9.7380
19	413,042	272	.65	6,056	379,460	22.2647	4.9701	5.4100
20	672,892	564	.83	27,093	628,606	48.0372	6.1016	6.5315

TABLE 5-5 TEST RESULTS, 3M ALGORITHM

TEST RUN	# CODED BITS	# BITS IN ERROR XMTD	BER $X10^{-3}$	NUMBER INCORRECT PELS	NUMBER 2-DIM. LINES	NUMBER CODED DATABITS	ESF	CF_3	CF_4
1	441,104	362	.82	12,255	202	397,549	33.8536	4.6539	5.1638
2	757,869	564	.74	38,682	924	668,555	68.5851	5.4175	6.1412
3	441,104	510	1.15	16,253	202	397,549	31.8686	4.6539	5.1638
4	757,869	510	.67	19,435	924	668,555	38.1098	5.4175	6.1412
5	441,104	296	.67	11,977	202	397,549	40.4628	4.6539	5.1638
6	757,869	682	.89	47,401	924	668,555	69.5029	5.4175	6.1412
7	441,104	598	1.35	11,767	202	397,549	19.6772	4.6539	5.1638
8	757,869	793	1.04	31,900	924	668,555	40.2270	5.4175	6.1412
9	757,869	564	.74	32,650	924	668,555	57.8901	5.4175	6.1412
10	757,869	564	.74	34,597	924	668,555	61.3422	5.4175	6.1412
11	419,039	290	.69	12,538	250	396,899	43.2345	4.8990	5.1723
12	709,588	510	.71	29,229	1068	664,381	57.3118	5.7861	6.1798
13	419,039	290	.69	9,599	250	396,899	33.1000	4.8990	5.1723
14	709,588	793	1.11	25,898	1068	664,381	32.6582	5.7861	6.1798
15	192,484	132	.68	1,160	128	126,122	8.7879	10.6651	16.2768
16	260,382	216	.82	13,650	675	201,902	63.1944	15.7681	20.3353
17	392,062	220	.56	17,432	1446	346,058	79.2364	10.4721	11.8643
18	264,163	216	.81	7,386	375	226,815	34.1944	7.7712	9.0508
19	431,481	356	.82	8,485	262	399,497	23.8343	4.7577	5.1386
20	716,340	564	.78	21,333	1309	676,844	36.0514	5.7315	6.0660

TABLE 5-6 TEST RESULTS, READ ALGORITHM

TEST RUN	# CODED BITS	# BITS IN ERROR XMTD	BER $X10^{-3}$	# INCORRECT PELS	# STUFFING BITS	# CODED DATABITS	ESF	CF_3	CF_4
1	442,434	362	.82	21,030	7,497	390,927	58.093	4.6399	5.2513
2	727,418	564	.775	38,283	9,842	620,671	67.877	5.6442	6.6150
3	442,434	510	1.15	21,175	7,497	390,927	41.519	4.6399	5.2513
4	727,418	510	.701	21,197	9,842	620,671	41.562	5.6442	6.6150
5	442,434	296	.669	14,305	7,497	390,927	48.327	4.6399	5.2513
6	727,418	682	.937	50,035	9,842	620,671	73.36	5.6442	6.6150
7	442,434	598	1.35	19,657	7,497	390,927	32.87	4.6399	5.2513
8	727,418	793	1.09	39,569	9,842	620,671	49.89	5.6442	6.6150
9	727,418	564	.775	50,559	9,842	620,671	89.64	5.6442	6.6150
10	727,418	564	.775	40,294	9,842	620,671	71.44	5.6442	6.6150
11	419,636	290	.691	13,299	7,497	390,927	45.85	4.8920	5.2513
12	678,257	510	.751	31,561	9,842	620,671	61.88	6.0534	6.6150
13	419,636	290	.691	13,105	7,497	390,927	45.18	4.8920	5.2513
14	678,257	783	1.154	34,636	9,842	620,671	44.23	6.0534	6.6150
15	188,070	120	.638	3,538	1,654	113,956	29.48	10.915	18.0145
16	250,379	216	.862	8,591	2,171	174,838	39.77	16.398	23.4830
17	370,448	220	.593	19,920	3,623	322,307	90.545	11.083	12.738
18	253,989	216	.850	7,549	2,467	210,040	34.94	8.082	9.7737
19	423,040	290	.685	9,361	8,567	385,871	32.27	4.852	5.320
20	664,857	564	.848	25,657	12,503	616,812	45.49	6.175	6.656

TABLE 5-7 TEST RESULTS, XEROX ALGORITHM

TEST RUN	# CODED BITS	# BITS IN ERROR XMTD	BER $X10^{-3}$	# INCORRECT PELS	# OF CODED DATA BITS	ESF	CF_3	CF_4
1	468,341	374	.798	15,642	430,660	41.8235	4.3833	4.7668
2	822,790	564	.685	25,464	748,406	45.1489	4.9900	5.4860
3	468,341	510	1.089	12,336	430,660	24.1882	4.3833	4.7668
4	822,790	510	.620	14,789	748,406	28.9980	4.9900	5.4860
5	468,341	326	.696	10,753	430,660	32.9846	4.3833	4.7668
6	822,790	682	.829	29,913	748,406	43.8607	4.9900	5.4860
7	468,341	626	1.336	11,229	430,660	17.9377	4.3833	4.7668
8	822,790	899	1.093	19,133	748,406	21.2825	4.9900	5.4860
9	822,790	564	.685	25,984	748,406	46.0709	4.9900	5.4860
10	822,790	564	.685	32,316	748,406	57.2979	4.9900	5.4860
11	448,833	362	.806	14,087	430,660	38.9144	4.5738	4.7668
12	782,611	510	.652	13,119	748,406	25.7235	5.2462	5.4860
13	448,833	296	.659	11,444	430,660	38.6622	4.5738	4.7668
14	782,611	793	1.013	15,660	748,406	19.7478	5.2462	5.4860
15	198,749	132	.664	2,571	133,050	19.4773	10.3289	15.4293
16	290,172	220	.758	9,556	233,764	43.4364	14.1493	17.5636
17	447,691	362	.809	13,802	419,000	38.1271	9.1709	9.7989
18	269,544	220	.816	3,041	236,284	13.8227	7.6161	8.6881
19	448,809	362	.807	9,017	421,857	24.9088	4.5740	4.8663
20	779,185	564	.724	18,894	749,859	33.5000	5.2693	5.4753

TABLE 5-8 TEST RESULTS FOR INFINITE K-FACTOR*

VERT. RESOLUTION	ALGORITHM	NO. CODED BITS	NO. BITS IN ERROR XMTD	BER X 10^{-3}	NO. INCORRECT PELS	NO. CODED DATA BITS	ESF	CF_3	CF_4
STANDARD RESOL. 3.85 /pm	MOD READ	418,173	284	.679	245,256	367,006	863.57	4.9091	5.5935
	JAPAN	421,115	290	.589	249,247	363,284	859.47	4.8748	5.6509
	3M	425,179	290	.682	16,652	381,510	57.42	4.8282	5.3809
	IBM	399,045	220	.551	245,062	349,188	1,113.9	5.1444	5.8790
	XEROX	450,789	362	.803	241,216	447,770	666.34	4.554	4.616
	AT & T	402,686	238	.591	31,493	350,103	132.3	5.0979	5.8636
	BPO	416,057	272	.654	35,666	365,761	131.1	4.9341	5.6126
	FRG	399,140	220	.551	245,092	352,379	1,114.05	5.1432	5.8257
HIGH RESOL. 7.7 /pm	MOD READ	663,239	564	.850	500,803	554,185	887.95	6.1904	7.4086
	JAPAN	663,182	564	.850	504,457	550,527	894.4[illegible]	6.1910	7.4578
	3M	703,756	564	.801	96,869	613,946	171.75	5.8340	6.6874
	IBM	664,554	564	.848	501,443	569,271	889.1	6.1782	7.3412
	XEROX	733,163	564	.769	490,854	721,204	870.31	5.6000	5.6929
	AT & T	666,296	564	.846	99,838	556,114	177.0	6.1620	7.3829
	BPO	661,948	564	.852	103,623	554,167	183.7	6.2025	7.4088
	FRG	663,011	564	.851	501,407	563,965	889.02	6.1925	7.2801

*DOCUMENT NO. - 4; ERROR PHASE - 0; ERROR FILE - 1; MIN. SCAN LINE TIME - 20 ms.

TABLE 5-9 CODED LINE LENGTH STATISTICS - MODIFIED READ ALGORITHM

TEST PARAMETERS				TEST RESULTS - MODIFIED READ			
TEST DESIGNATION	TEST DOCUMENT #	MIN. SCAN. LINE TIME (ms.)	VERTICAL RESOLUTION	MIN. BITS/LINE	MAX. BITS/LINE	AVG. BITS/LINE	STANDARD DEVIATION
A	4	20	3.85	96	1,207	370.52	336.10
B	4	20	7.70	96	1,089	305.96	271.14
C	4	10	3.85	48	1,207	351.67	351.91
D	4	10	7.70	48	1,089	285.54	287.46
E	1	20	3.85	96	797	159.17	149.04
F	1	10	7.70	48	797	106.74	133.03
G	5	10	7.70	48	1,063	157.29	160.45
H	5	20	3.85	96	1,063	214.97	183.42
I	7	20	3.85	96	718	353.64	177.28
J	7	10	7.70	48	766	280.17	165.63

Table 5-10 Comparison of Modified HUFFMAN and Modified READ Codes *

TEST PARAMETERS				MODIFIED HUFFMAN				MODIFIED READ				
TEST DOC	VERT RESOL.	MSLT	ERROR FILE	BER X 10^{-3}	CF_3	CF_4	ESF	K	BER X 10^{-3}	CF_3	CF_4	ESF
4	3.85	10	1	.82	4.67	4.91	24.93	2	.66	4.91	5.23	40.31
4	3.85	20	1	.81	4.46	4.91	27.64	2	.82	4.66	5.23	45.24
								INF	.68	4.91	5.59	863.57
4	7.7	20	1	.65	4.46	4.91	29.28	4	.78	5.65	6.58	58.70
								INF	.85	6.19	7.41	887.95
4	3.85	20	2	1.1	4.46	4.91	19.23	2	1.16	4.66	5.23	27.69
4	3.85	20	3	.70	4.46	4.91	20.98	2	.67	4.66	5.23	57.41
4	3.85	20	4	1.3	4.46	4.91	14.66	2	1.36	4.66	5.23	21.22
1	3.85	20	1	.81	10.20	15.14	9.70	2	.63	10.85	17.81	20.60
								INF	.59	11.56	21.36	644.69
7	3.85	20	1	.82	4.64	4.99	22.06	2	.69	4.89	5.31	30.27
								INF	.56	5.18	5.67	799.80

* Error Phase - 0

infinite K-factor is 7.41 which is 66% better than the CF_3 of the Modified Huffman. For a non-complex image like CCITT document number 1 (see row 7 of Table 5-10) the relative improvement is greater even at low resolution. In this case, the Modified READ CF_4 at an infinite K factor exceeds the Modified Huffman CF_3 by over 100%. In general, the ESF for the Modified READ is about twice the ESF for the Modified Huffman code.

Figures 5-1 through 5-6 illustrate the subjective effects of the errors in several of the simulations listed in Table 5-10. Figures 5-1 and 5-2 simulate the output for the Modified Huffman code at standard and high resolution respectively (rows 2 and 3 of Table 5-10). Figures 5-3 and 5-4 illustrate the output images for the Modified READ code for the same relative test conditions. Notice that the legibility of the Modified READ code images is poorer than the corresponding Modified Huffman documents. This corresponds to the indication of the relative ESF measurements.

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet, et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 125 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étudiés récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen ; ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles, voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger, et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gérera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie Internationale pour l'Informatique a été installé à Toulouse en [illegible] même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux.

Figure 5-1 Modified Huffman, standard resolution (Row 2 of Table 5-10)
Photo n° 1 - Document très dense lettre 1,5mm de haut -
Restitution photo n° 9 -

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet, et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 425 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étaient récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen ; ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles, voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger, et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gèrera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie Internationale pour l'Informatique a été installé à Toulouse en février ; la même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux.

Figure 5-2 Modified Huffman, high resolution (Row 3 of Table 5-10)

Photo n° 1 - Document très dense lettre 1,5mm de haut -

Restitution photo n° 9

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet, et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 425 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étaient récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen, ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles, voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger, et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gèrera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie internationale pour l'informatique a été installé à Toulouse en février ; la même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux.

Figure 5-3 Modified READ, standard resolution (Row 2 of Table 5-10)

Photo n° 2 - Document très dense lettre 2, sinon du haut

Restitution photo n° 9

L'ordre de lancement et de réalisation des applications fait l'objet de décisions au plus haut niveau de la Direction Générale des Télécommunications. Il n'est certes pas question de construire ce système intégré "en bloc" mais bien au contraire de procéder par étapes, par paliers successifs. Certaines applications, dont la rentabilité ne pourra être assurée, ne seront pas entreprises. Actuellement, sur trente applications qui ont pu être globalement définies, six en sont au stade de l'exploitation, six autres se sont vu donner la priorité pour leur réalisation.
Chaque application est confiée à un "chef de projet", responsable successivement de sa conception, de son analyse-programmation et de sa mise en oeuvre dans une région-pilote. La généralisation ultérieure de l'application réalisée dans cette région-pilote dépend des résultats obtenus et fait l'objet d'une décision de la Direction Générale. Néanmoins, le chef de projet doit dès le départ considérer que son activité a une vocation nationale donc refuser tout particularisme régional. Il est aidé d'une équipe d'analystes-programmeurs et entouré d'un "groupe de conception" chargé de rédiger le document de "définition des objectifs globaux" puis le "cahier des charges" de l'application, qui sont adressés pour avis à tous les services utilisateurs potentiels et aux chefs de projet des autres applications. Le groupe de conception comprend 6 à 10 personnes représentant les services les plus divers concernés par le projet,et comporte obligatoirement un bon analyste attaché à l'application.

II - L'IMPLANTATION GEOGRAPHIQUE D'UN RESEAU INFORMATIQUE PERFORMANT

L'organisation de l'entreprise française des télécommunications repose sur l'existence de 20 régions. Des calculateurs ont été implantés dans le passé au moins dans toutes les plus importantes. On trouve ainsi des machines Bull Gamma 30 à Lyon et Marseille, des GE 425 à Lille, Bordeaux, Toulouse et Montpellier, un GE 437 à Massy, enfin quelques machines Bull 300 TI à programmes câblés étaient récemment ou sont encore en service dans les régions de Nancy, Nantes, Limoges, Poitiers et Rouen ; ce parc est essentiellement utilisé pour la comptabilité téléphonique.
A l'avenir, si la plupart des fichiers nécessaires aux applications décrites plus haut peuvent être gérés en temps différé, un certain nombre d'entre eux devront nécessairement être accessibles, voire mis à jour en temps réel : parmi ces derniers le fichier commercial des abonnés, le fichier des renseignements, le fichier des circuits, le fichier technique des abonnés contiendront des quantités considérables d'informations.
Le volume total de caractères à gérer en phase finale sur un ordinateur ayant en charge quelques 500 000 abonnés a été estimé à un milliard de caractères au moins. Au moins le tiers des données seront concernées par des traitements en temps réel.
Aucun des calculateurs énumérés plus haut ne permettait d'envisager de tels traitements.
L'intégration progressive de toutes les applications suppose la création d'un support commun pour toutes les informations, une véritable "Banque de données", répartie sur des moyens de traitement nationaux et régionaux, et qui devra rester alimentée, mise à jour en permanence, à partir de la base de l'entreprise, c'est-à-dire les chantiers, les magasins, les guichets des services d'abonnement, les services de personnel etc.
L'étude des différents fichiers à constituer a donc permis de définir les principales caractéristiques du réseau d'ordinateurs nouveaux à mettre en place pour aborder la réalisation du système informatif. L'obligation de faire appel à des ordinateurs de troisième génération, très puissants et dotés de volumineuses mémoires de masse, a conduit à en réduire substantiellement le nombre.
L'implantation de sept centres de calcul interrégionaux constituera un compromis entre : d'une part le désir de réduire le coût économique de l'ensemble, de faciliter la coordination des équipes d'informaticiens; et d'autre part le refus de créer des centres trop importants difficiles à gérer et à diriger,et posant des problèmes délicats de sécurité. Le regroupement des traitements relatifs à plusieurs régions sur chacun de ces sept centres permettra de leur donner une taille relativement homogène. Chaque centre "gèrera" environ un million d'abonnés à la fin du VIème Plan.
La mise en place de ces centres a débuté au début de l'année 1971 : un ordinateur IRIS 50 de la Compagnie Internationale pour l'Informatique a été installé à Toulouse en février ; la même machine vient d'être mise en service au centre de calcul interrégional de Bordeaux.

Figure 5-4 Modified READ, high resolution (Row 3 of Table 5-10)

Photo n° 1 - Document très dense lettre 1,5mm de haut -

Restitution photo n° 9

THE SLEREXE COMPANY LIMITED

SAPORS LANE - BOOLE - DORSET - BH 25 8 ER

TELEPHONE BOOLE (945 13) 51617 - TELEX 123456

Our Ref. 350/PJC/EAC 18th January, 1972.

Dr. P.N. 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

Figure 5-5 Modified READ (Row 7 of Table 5-10)

CCITTの概要

沿革

CCITTは、国際電気通信連合（ITU）の四つの常設機関（事務総局、国際周波数登録委員会、CCIR、CCITT）の一つとして、ITUの中でも、世界の国際通信上の諸問題を真先に取上げ、その解決方法を見出して行く重要な機関である。日本名は、国際電信電話諮問委員会と称する。

CCITTの前身は、CCIF（国際電話諮問委員会）とCCIT（国際電信諮問委員会）である。CCIFは、1924年にヨーロッパに「国際長距離電話通信諮問委員会」が設置され、これが1925年のパリ電信電話会議のとき、正式に、「国際電話諮問委員会」として万国電信連合の公式機関となったものである。CCITTは、同じく1925年の会議のとき、CCIFと併立するものとして設置された。

そして、CCIFは、1956年の12月に第18回総会が開催されたのち、CCITTは、同年同月に第8回総会が開催されたのち、併合されて現在のCCITTとなった。このCCITTは、CCIFとCCITTが解散した直後、第1回総会を開催し、第2回総会は、1960年にニューデリーで、第3回総会は、1964年、ジュネーブで、第4回総会は、1968年、アルゼンチンで開催された。

CCIFとCCITTが合併したのは、有線電気通信の分野、とくに伝送路について電信回線と電話回線とを技術的に分ける意味がなくなってきたこと、各国とも一体において、電信部門と電話部門は同一組織内にあること、CCIFの事務局とCCITTの事務局の合併による能率増進等がおもな理由であった。

CCITTは、上述のように、ヨーロッパ内の国ぐにによって、ヨーロッパ内の電信・電話の技術・運用・料金の基準を定め、あるいは統一をはかってきたので、現在でも、その影響を受け、会合参加国は、ヨーロッパの国が多く、ヨーロッパで出起する問題の研究が多い。たとえば、1960年のCCITT勧告の中で、技術上配慮する距離は約2,500kmであったが、これはヨーロッパ内領域を想定したものである。

しかしながら、1956年9月に敷設された大西洋横断電話ケーブルは、大陸間電話通信の自動化および半自動化への技術的可能性を与え、CCITTがこの問題を取り上げるに及び、CCITTの性格は漸次、汎世界的色彩を実質的に帯びるに至った。この汎世界的性格は第2次世界大戦後目ざましくなったアジア・アフリカ植民地の独立に伴ってITUの構成員の中にこれらの国が加わり、ITUの中に新しい意見が導入されたことにも起因して、技術面、政治面の双方から導入されてきた。CCITTの汎世界化は、1960年の第2回総会がニューデリーで開催されたことにもあらわれている。この総会までは、CCIT、CCIFのいずれにしゃ、アメリカやアジアで総会が開催されたことがなく、CCITT委員長も、ニューデリー総会の準備文書で、この点には注目すべきであるとのべている。

任務

ITUは、全権委員会議、主管庁会議を始めとして、七つの機関をもち、それぞれの機関の権限と任務は国際電気通信条約に明記されている。そこで条約を参照してみるならば、CCITTの任務は、つぎのとおりとなっている。

「国際電信電話諮問委員会（CCITT）は、電信および電話に関する技術、運用および料金の問題について研究し、および意見を表明することを任務とする。」（1965年モントルー条約第187号）

「各国際諮問委員会は、その任務の遂行に当たって、新しい国または発展の途上にある国における地域的および国際的分野にわたる電気通信の創設、発達および改善に直接関連のある問題について研究し、および意見を作成するように妥当な注意を払わなければならない。」（同第188号）

「各国際諮問委員会は、また、関係国の要請に基づき、その国内電気通信の問題について研究し、かつ、勧告を行なうことができる。」（同第189号）

上記第187号と第188号にいわれる「意見」とは、フランス語の Avis から訳したもので、英語では、「勧告（Recommendation）」となっている。CCITTの表明する意見は、国際法的には強制力をもたないものであって、この点が、条約、電信規則、電話規則等各国を拘束する力をもっているものと異なる。もっとも意見とは称しても、技術的分野では、電信規則のごとき、各国政府が承認してその内容を実施する強制規則をもたないので、実際にある機器の仕様を定める場合には、多くの国の意見が統一されたこの「意見」に従わなければ、円滑な国際通信を行なうことができない場合が多い。この意見（または勧告）は、国際通信を行なう場合を国が直面する問題について、具体的意見を表明するもので、たとえば、大陸間ケーブルで大陸間通話を半自動化しようとする場合、その信号方式や取り扱う通話の種類および料金は、どのようにするかを研究して意見を表明する。したがって、CCITTの活動は、つねに時代の最先端を行くもので、CCITTの活動方向は、そのまま世界の国際通信の活動方向であるともいえる。

この意見は、また、電信規則以下のその他の規則のごとく、数年以上の間隔をもって開催される主管庁会議というような大会議の決定をまたなくても表明することができ、また、その改正も容易であるので、現在のように進歩の早い国際通信界では、関係国の意見を統一した国際的見解としては非常に便利である。

Figure 5-6 Modified READ (Row 8 of Table 5-10)

6.0 REFERENCES

1. CCITT Contribution No. 66, "Criteria for the Evaluation of Two-Dimensional Coding Techniques for use in Digital Facsimile Terminals" Source: United States of America; Date: January 1979.
2. CCITT Contribution COM XIV - No. 70, "Report of the Meeting Held in Geneva," 11-15 Dec. 1978, Annex No. 2, Section III.
3. National Communications System Report, "Development of a Computer Program for Measuring the Compression and Error Sensitivity of Facsimile Coding Techniques," August 10, 1979.
4. CCITT Contribution COM XIV - No. 42, Japan Algorithm.
5. CCITT Contribution COM XIV - No. 74, 3M Algorithm.
6. National Communications System Report, "Measurement of Compression Factor and Error Sensitivity Factor of Five Selected Two-Dimensional Facsimile Coding Techniques," October 1979.
7. Collective Letter No. 87 from the CCITT to Members of Study Group XIV COM/TO dated 21 May 1979, page 5, section 4.0.
8. Federal Republic of Germany, "Sensibility of Redundancy Reducing Codes to Transmission Bit Errors," CCITT Study Group XIV - Contribution No. 5, February 1977.
9. CCITT Contribution COM XIV - No. 64, IBM Algorithm.
10. CCITT Contribution COM XIV - No. 84, XEROX Algorithm.
11. CCITT Contribution COM XIV - No. 81, AT&T Algorithm.
12. National Communications System Report, "Measurement of Compression Factor and Error Sensitivity Factor of Facsimile Coding Techniques submitted to the CCITT by Great Britain and Germany," October, 1979.

APPENDIX A

DRAFT RECOMMENDATION (T.4) FOR STANDARDIZATION OF GROUP 3 FACSIMILE APPARATUS FOR DOCUMENT TRANSMISSION

Recommendation T.4 (Draft)

STANDARDIZATION OF GROUP 3 FACSIMILE APPARATUS FOR DOCUMENT TRANSMISSION

The CCITT,

considering

a) that Recommendation T.2 refers to Group 1 type apparatus for ISO A4 document transmission over a telephone-type circuit in approximately six minutes;

b) that Recommendation T.3 refers to Group 2 type apparatus for ISO A4 document transmission over a telephone-type circuit in approximately three minutes;

c) that there is a demand for Group 3 apparatus which enables an ISO A4 document to be transmitted over a telephone-type circuit in approximately one minute;

d) that for a large number of applications black and white reproduction is sufficient;

e) that such a service may be requested either alternatively with telephone conversation, or when either or both stations are not attended; in both cases, the facsimile operation will follow Recommendation T.30,

(unanimously) declares the view

that Group 3 facsimile apparatus for use on the general switched telephone network and international leased circuits should be designed and operated according to the following standards :

1. Scanning track

The message area should be scanned in the same direction in the transmitter and receiver. Viewing the message area in a vertical plane, the scanning direction should be from left to right and subsequent scans should be adjacent and below the previous scan.

2. Dimensions of apparatus

The following dimensions should be used :

2.1 A normal definition standard and an optional higher definition standard of 3.85 and 7.7 line/mm respectively in vertical direction;

2.2 1728 black and white picture elements along the scanned line;

2.3 A scanning line length of 215 mm. Other scanning line lengths may be employed in which case the scanning density should be changed to maintain the correct picture proportions;

2.4 Input documents up to a minimum of ISO A4 size should be accepted.

3. Transmission time per scanning line

A total scanning line is defined as the sum of DATA bits plus any required FILL bits plus the EOL bits.

The minimum transmission times of the total scanning line should conform to the following :

3.1 20 milliseconds recommended standard,

3.2 10 milliseconds recognized option with a mandatory fall-back to the 20 milliseconds standard,

3.3 5 milliseconds recognized option with a mandatory fall-back to the 10 milliseconds option and the 20 milliseconds standard,

3.4 0 millisecond recognized option with a mandatory fall-back to the 5 milliseconds option, and the 10 milliseconds option and the 20 milliseconds standard and an optional fall-back to the 40 milliseconds option,

3.5 40 milliseconds recognized option,

3.6 10 milliseconds recognized option with a mandatory fall-back to the 20 milliseconds standard,

3.7 20 milliseconds recommended standard,

3.8 40 milliseconds recognized option.

The identification and choice of this minimum transmission time is to be made in the pre-message (Phase B) portion of the T.30 control procedure.

The maximum transmission time of any total scanning line should be less than 5 seconds.

Note: The alternatives 3.1 to 3.5 apply to both the normal and optional higher definition standard of 3.85 and 7.7 line/mm respectively in the vertical direction. The alternatives 3.6 to 3.8 apply to the normal definition standard and the times should be halved for the higher definition standard.

4. Coding Scheme

4.1 One-dimensional coding scheme

The one-dimensional run length encoding scheme recommended for Group 3 apparatus is as follows :

4.1.1 DATA

A line of DATA is composed of a series of variable length code words. Each code word represents a run length of either all white or all black. White runs and black runs alternate. A total of 1728 picture elements represent one horizontal scanning line of the document of standard A4 size. In order to insure that the receiver maintains color synchronization, all DATA lines will begin with a white run length code word. If the actual scanning line begins with a black run, a white run length of zero will be sent. Black or white run lengths, up to a maximum length of one scanning line (1728 picture elements or pels) are defined by the code words in Tables 1 and 2. The code words are of two types : Terminating Code words and Made Up Code words. Each run length is represented by either one Terminating Code word or one Make Up Code word followed by a Terminating Code word.

Run lengths in the range of 0 to 63 pels are encoded with their appropriate Terminating Code word. Note that there is a different list of code words for black and white run lengths.

Run lengths in the range of 64 to 1728 pels are encoded first by the Make Up Code word representing the run length which is equal to or shorter than that required. This is then followed by the Terminating Code word representing the difference between the required run length and the run length represented by the Make Up Code.

4.1.2 END OF LINE (EOL)

This code word follows each line of DATA. It is a unique code word that can never be found within a valid line of DATA; therefore, resynchronization after an error burst is possible.

In addition, this signal will occur prior to the first DATA line of a page.

Format: 000000000001

4.1.3 FILL

A pause may be placed in the message flow by transmitting FILL. FILL may be inserted between a line of DATA and an EOL, but never within a line of DATA. FILL must be added to insure that each line of DATA, FILL and EOL exceeds the minimum transmission time of a total scanning line established in the pre-message control procedure. The maximum length for a single line of FILL is 5 seconds, after which the receiver may disconnect.

Format: variable length string of 0's.

4.1.4 RETURN TO CONTROL (RTC)

The end of a document transmission is indicated by sending six consecutive EOL's. Following the RTC signal, the transmitter will send the post message commands in the standard T.30 framed format.

Format: 000000000001----------000000000001
(total of 6 times)

To further clarify the relationship of the signals defined herein, the Figures 1 and 2 are offered. Figure 1 shows several scan lines of data starting at the beginning of a transmitted page. Figure 2 shows the last line of a page.

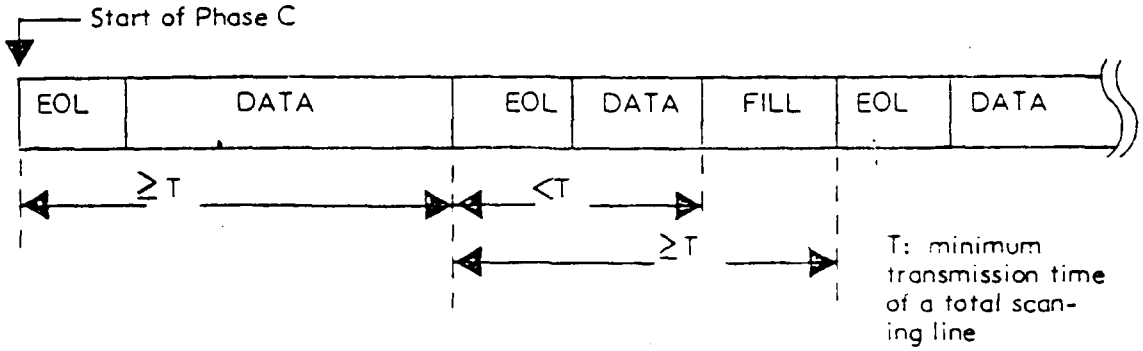


FIGURE 1

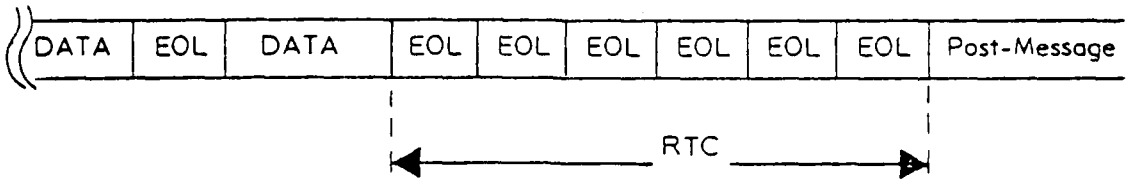


FIGURE 2

TABLE I

Terminating Codes

White Run Length	Code Word	Black Run Length	Code Word
0	00110101	0	0000110111
1	000111	1	010
2	0111	2	11
3	1000	3	10
4	1011	4	011
5	1100	5	0011
6	1110	6	0010
7	1111	7	00011
8	10011	8	000101
9	10100	9	000100
10	00111	10	0000100
11	01000	11	0000101
12	001000	12	0000111
13	000011	13	00000100
14	110100	14	00000111
15	110101	15	000011000
16	101010	16	0000010111
17	101011	17	0000011000
18	0100111	18	0000001000
19	0001100	19	00001100111
20	0001000	20	00001101000
21	0010111	21	00001101100
22	0000011	22	000001110111
23	0000100	23	000001101000
24	0101000	24	000000010111
25	0101011	25	000000011000
26	0010011	26	00001100101 0
27	0100100	27	00001100101 1
28	0011000	28	000011001100
29	00000010	29	000011001101
30	00000011	30	000001101000
31	00011010	31	000001101001
32	00011011	32	000001101010
33	00010010	33	000001101011
34	00010011	34	000011010010
35	00010100	35	000011010011
36	00010101	36	000011010100
37	00010110	37	000011010101
38	00010111	38	000011010110
39	00101000	39	000011010111
40	00101001	40	000001101100
41	00101010	41	000001101101
42	00101011	42	000011011010
43	00101101	43	000011011011
44	00101101	44	000001010100
45	00000100	45	000001010101
46	00000101	46	000001010110
47	00001010	47	000001010111
48	00001011	48	000001100100
49	01010010	49	000001100101
50	01010011	50	000001010010
51	01010100	51	000001010011
52	01010101	52	000000100100
53	00100100	53	000000110111
54	00100101	54	000000111000
55	01011000	55	000000100111
56	01011001	56	000000101000
57	01011010	57	000001011000
58	01011011	58	000001011001
59	01001010	59	000000101011
60	01001011	60	000000101100
61	00110010	61	000001011010
62	00110011	62	000001100110
63	00110100	63	000001100111

TABLE 2

Make Up Codes

White Run Lengths	Code Word	Black Run Lengths	Code Word
64	11011	64	0000001111
128	10010	128	000011001000
192	010111	192	000011001001
256	0110111	256	000001011011
320	00110110	320	000000110011
384	00110111	384	000000110100
448	01100100	448	000000110101
512	01100101	512	0000001101100
576	01101000	576	0000001101101
640	01100111	640	0000001001010
704	011001100	704	0000001001011
768	011001101	768	0000001001100
832	011010010	832	0000001001101
896	011010011	896	0000001110010
960	011010100	960	0000001110011
1024	011010101	1024	0000001110100
1088	011010110	1088	0000001110101
1152	011010111	1152	0000001110110
1216	011011000	1216	0000001110111
1280	011011001	1280	0000001010010
1344	011011010	1344	0000001010011
1408	011011011	1408	0000001010100
1472	010011000	1470	0000001010101
1536	010011001	1536	0000001011010
1600	010011010	1600	0000001011011
1664	011000	1664	0000001100100
1728	010011011	1728	0000001100101
EOL	000000000001	EOL	000000000001

Note: It is recognized that machines exist which accommodate larger paper widths whilst maintaining the standard horizontal resolution. This option has been provided for by the addition of the Make Up Code Set defined as follows:

Run Length (Black and White)	Make Up Codes
1792	00000001000
1856	00000001100
1920	00000001101
1984	000000010010
2048	000000010011
2112	000000010100
2176	000000010101
2240	000000010110
2304	000000010111
2368	000000011100
2432	000000011101
2496	000000011110
2560	000000011111

The identification and choice of either the standard code table or the extended code table is to be made in the pre-message (Phase B) portion of the T.30 control procedures.

4.2 Two dimensional coding scheme

The two-dimensional coding scheme is an optional extension of the one-dimensional coding scheme specified in Paragraph 4.1 and is as follows:

4.2.1 DATA

4.2.1.1 Parameter K

In order to limit the disturbed area in the event of transmission errors, after each line coded one-dimensionally at most K-1 successive lines shall be coded two-dimensionally. The value of K shall be set as follows:

Normal resolution standard: K=2
Higher resolution standard: K=4

Note 1: Some Administrations pointed out that for higher resolution K may optionally be set to a lower value.

Note 2: Some Administrations reserve the right to approve only such apparatus for use in the facsimile service in their respective countries which will be able to produce a visible sign on its received facsimile message indicating that two-dimensional coding has been used in the transmission process.

4.2.1.2 One dimensional coding

This conforms with the description of DATA, Paragraph 4.1.1.

4.2.1.3 Two-dimensional coding

This is a line-by-line coding method in which the position of each changing picture element on the current or coding line is coded with respect to the position of a corresponding reference element situated on either the coding line or the reference line which lies immediately above the coding line. After the coding line has been coded it becomes the reference line for the next coding line.

a) Definition of changing picture elements (see Figure 3)

A changing element is defined as an element whose "color" (i.e. black or white) is different from that of the previous element along the same scan line.

a_o The reference or starting changing element on the coding line. At the start of the coding line a_o is set on an imaginary white changing element situated just before the first element on the line. During the coding of the coding line, the position of a_o is defined by the previous coding mode. (See Paragraph 4.2.1.3,b)

a_1 The next changing element to the right of a_o on the coding line.

a_2 The next changing element to the right of a_1 on the coding line.

b_1 The first changing element on the reference line to the right of a_0 and of opposite color to a_0.

b_2 The next changing element to the right of b_1 on the reference line.

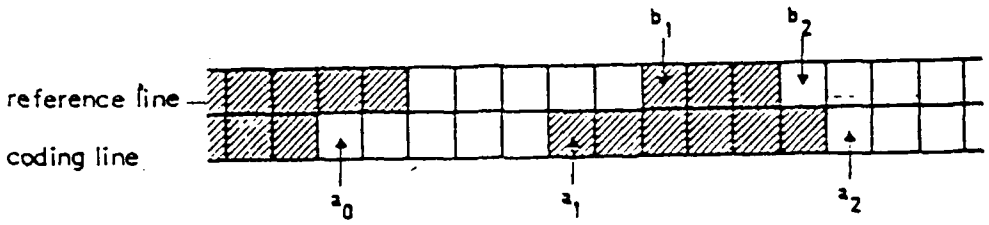


FIGURE 3 - Changing Picture Elements

b) Coding Modes

One of three coding modes are chosen according to the coding procedure described in Paragraph 4.2.1.3,c to code the position of each changing element along the coding line. Examples of the three coding modes are given in Figures 4, 5 and 6.

(i) Pass mode

This mode is identified when the position of b_2 lies to the left of a_1.

When this mode has been coded, a_o is set on the element of the coding line below b_2 in preparation for the next coding. (i.e. on a_o')

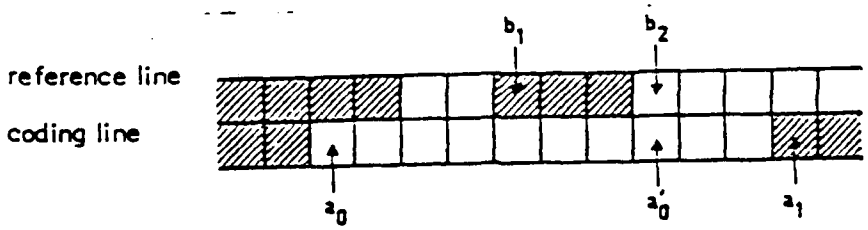


FIGURE 4 - Pass mode

However, the state where b_2 occurs just above a_1, as shown in Figure 5, is not considered as a Pass mode.

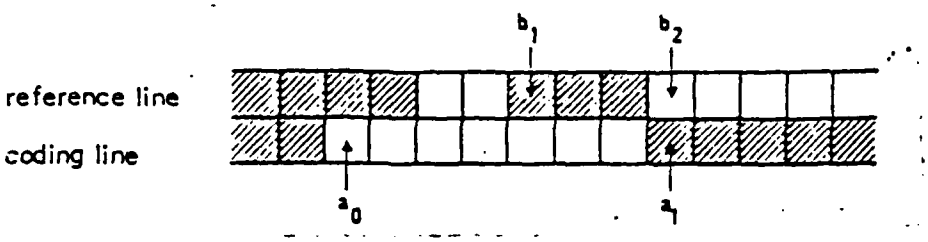


FIGURE 5 - An example not corresponding to a Pass mode

(ii) Vertical mode

When this mode is identified, the position of a_1 is coded relative to the position of b_1. The relative distance a_1b_1 can take on one of seven values V(0), $V_R(1)$, $V_R(2)$, $V_R(3)$, $V_L(1)$, $V_L(2)$ and $V_L(3)$, each of which is represented by a separate code word. The subscripts R and L indicate that a_1 is to the right or left respectively of b_1 and the number in brackets indicates the value of the distance a_1b_1. After vertical mode coding has occurred, the position of a_o is set on a_1. (see Figure 6)

(iii) *Horizontal mode*

When this mode is identified, both the run-lengths a_oa_1 and a_1a_2 are coded using the code words H + $M(a_oa_1)$ + $M(a_1a_2)$. H is the flag code word '001' taken from the two-dimensional code table (Table 3). $M(a_oa_1)$ and $M(a_1a_2)$ are code words which represent the length and "color" of the runs a_oa_1 and a_1a_2 respectively and are taken from the appropriate white or black one-dimensional code tables (Tables 1 and 2). After a horizontal mode coding, the position of a_o is set on a_2. (see Figure 6)

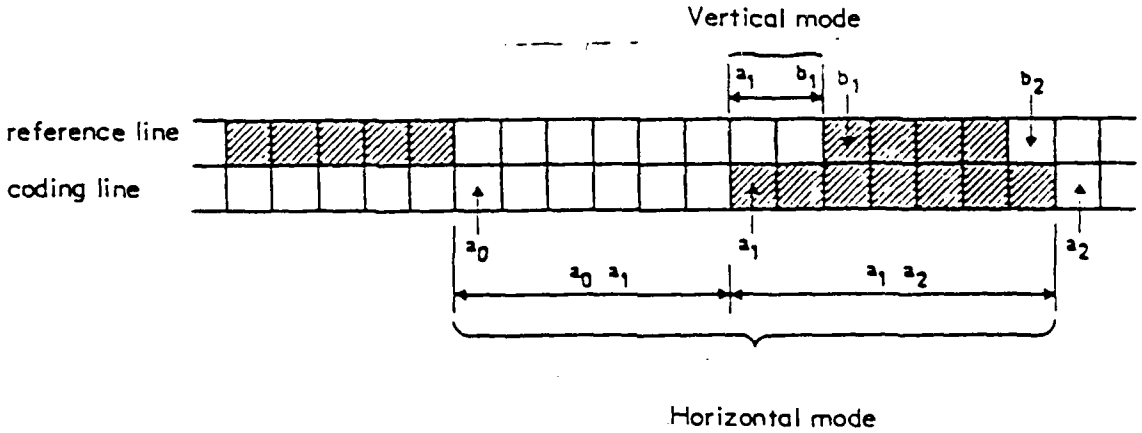


FIGURE 6 - Vertical mode and Horizontal mode

c) *Coding Procedure*

The coding procedure identifies the coding mode that is to be used to code each changing element along the coding line. When one of the three coding modes has been identified, an appropriate code word is selected from the code table given in Table 3. The coding procedure is formally defined by the flow diagram given in Figure 9.

Note: It does not affect compatibility to restrict the use of Pass mode in the encoder to a single Pass mode. Variations of the algorithm which do not affect compatibility should be the subject of further study.

Step 1

i) If a pass mode is identified this is coded using the code word '0001' (Table 3). After this processing, picture element a_o' just under b_2 is regarded as the new starting picture element a_o for the next coding. (see Figure 4)

ii) If a pass mode is not detected then proceed to Step 2.

Step 2

i) Determine the absolute value of the relative distance a_1b_1.

ii) If $|a_1b_1| \leq 3$, as shown in Table 3, a_1b_1 is coded by Vertical mode, after which position a_1 is regarded as the new starting picture element a_o for the next coding.

iii) If $|a_1b_1| > 3$, as shown in Table 3, following Horizontal mode code '001', a_oa_1 and a_1a_2 are respectively coded by one-dimensional coding. After this processing position a_2 is regarded as the new starting picture element a_o for the next coding.

TABLE 3

Two-dimensional code table

MODE	ELEMENTS TO BE CODED		NOTATION	CODE WORD
PASS	b_1, b_2		P	0001
HORIZONTAL	a_0a_1, a_1a_2		H	001 + $M(a_0a_1)$ + $M(a_1a_2)$ Note 1
VERTICAL	a_1 just under b_1	$a_1b_1 = 0$	V(0)	1
	a_1 to the right of b_1	$a_1b_1 = 1$	$V_R(1)$	011
		$a_1b_1 = 2$	$V_R(2)$	000011
		$a_1b_1 = 3$	$V_R(3)$	0000011
	a_1 to the left of b_1	$a_1b_1 = 1$	$V_L(1)$	010
		$a_1b_1 = 2$	$V_L(2)$	000010
		$a_1b_1 = 3$	$V_L(3)$	0000010
EXTENSION	2-D (extensions) 1-D (extensions)			Note 2 0000001xxx 000000001xxx

Note 1 Code M( ) of Horizontal mode represents the code words in Tables 1 and 2, Paragraph 4.1.1.

Note 2 It is suggested that Uncompressed mode is recognized as an optional extension of the two-dimensional coding scheme for Group 3 apparatus. The bit assignment for the xxx bits is 111 for the Uncompressed mode of operation. The code words with this bit assignment are used to switch into Uncompressed mode. Table 4 gives an example of a code table for Uncompressed mode and the exit from it.

Note 3 Further study is needed to define other unspecified xxx bit assignments and their use for any further extensions.

TABLE 4

Uncompressed mode code table

Image Pattern	Code Word
1	1
01	01
001	001
0001	0001
0000	00001
00000	000001

Exit	Image Pattern	Code Word
		0000001T
	0	00000001T
	00	000000001T
	000	0000000001T
	0000	00000000001T

T denotes a tag bit which tells the color of the next run. (black = 1, white = 0)

d) Processing the first and last picture elements in a line

(i) Processing the first picture element

The first starting picture element a_o on each coding line is imaginarily set at a position just before the first picture element, and is regarded as a white picture element. (4.2.1.3,a)

The first run length on a line $a_o a_1$ is replaced by $a_o a_1 - 1$. Therefore, if the first run is black and is deemed to be coded by horizontal mode coding, then the first code word $M(a_o a_1^-)$ corresponds to a white run of zero length. (see Figure 10, Example 5)

(ii) Processing the last picture element

The coding of the coding line continues until the position of the imaginary changing element situated just after the last actual element has been coded. This may be coded as a_1 or a_2. Also, if b_1 and/or b_2 are not detected at any time during the coding of the line, they are positioned on the imaginary changing element situated just after the last actual picture element on the reference line.

4.2.2 LINE SYNCHRONIZATION CODE WORD

To the end of every coded line is added the end-of-line (EOL) code word '000000000001'. The EOL codeword is followed by a single tag bit which indicates whether one- or two-dimensional coding is used for the next line.

Format:

EOL + 1 : one-dimensional coding of next line
EOL + 0 : two-dimensional coding of next line

4.2.3 FILL

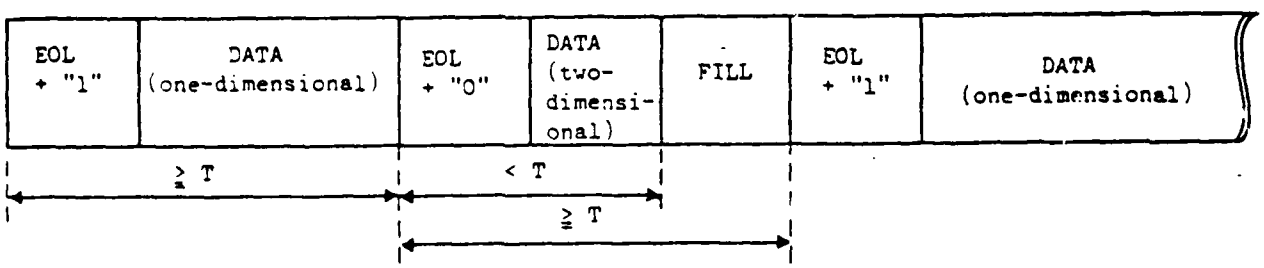
FILL is inserted between a line of DATA and the line synchronization signal, EOL + tag bit, but is not inserted in DATA.

Format : variable length string of 0's.

4.2.4 RETURN TO CONTROL (RTC)

The format used is six consecutive line synchronization code words, i.e., 6 X (EOL+1).

To further clarify the relationship of the signals defined herein, Figures 7 and 8 are offered in the case of K = 2. Figure 7 shows several scan lines of data starting at the beginning of a transmitted page. Figure 8 shows the last several lines of a page.



T: minimum transmit time of a total scanning line

FIGURE 7 - Message transmission (first part of page)

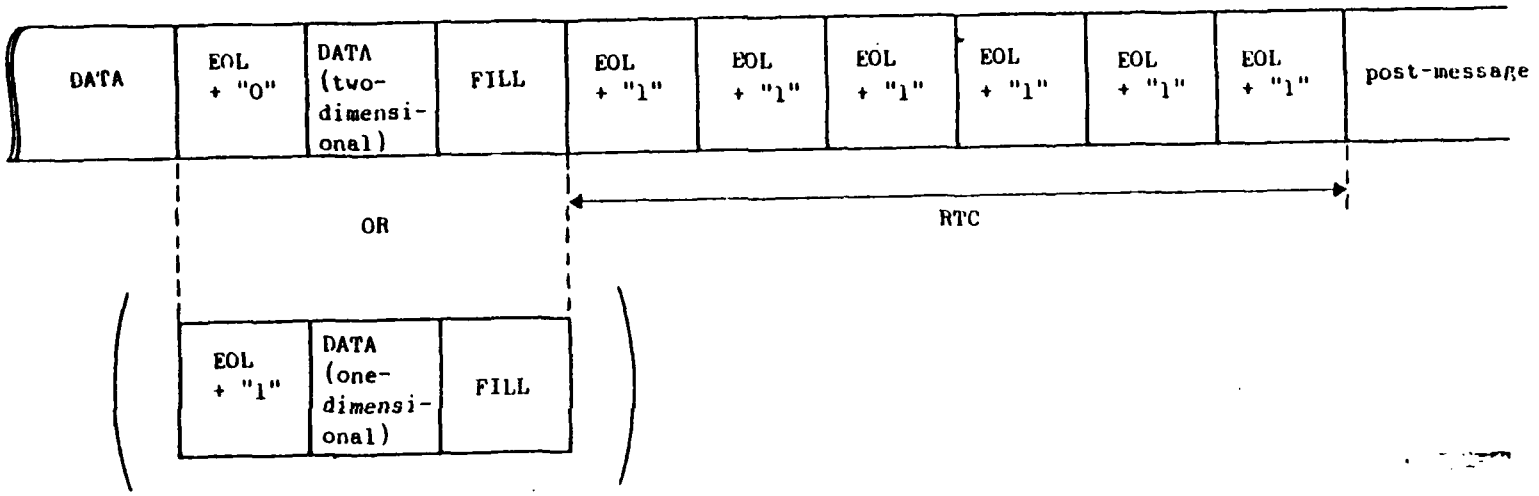


FIGURE 8 - Message transmission (last part of page)

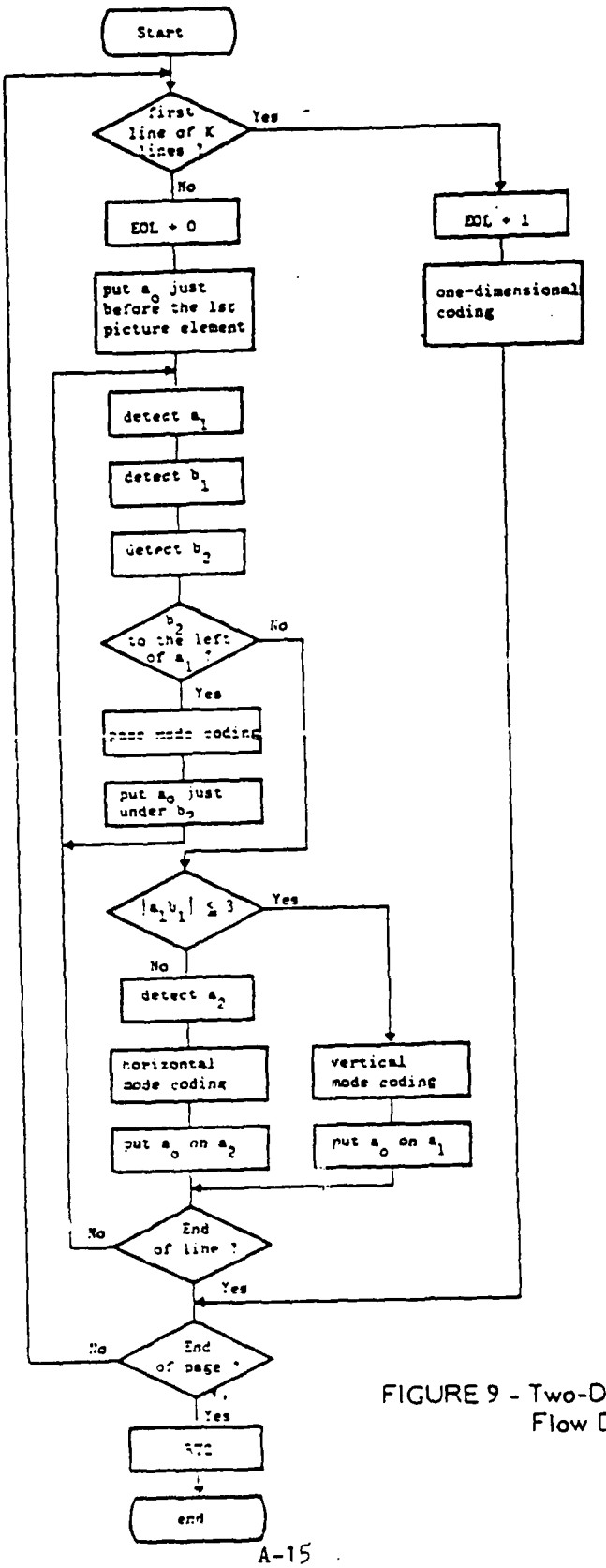


FIGURE 9 - Two-Dimensional Coding Flow Diagram

4.2.5 Coding Examples

Figure (10) shows coding examples of the first part of scanning lines and Figure (11) coding examples of the last part, while Figure (12) shows other coding examples. The notations P, H and V in the figures are, as shown in Table 3, the symbols for Pass mode, Horizontal mode and Vertical mode respectively. The picture elements marked with black spots indicate the changing picture elements to be coded.

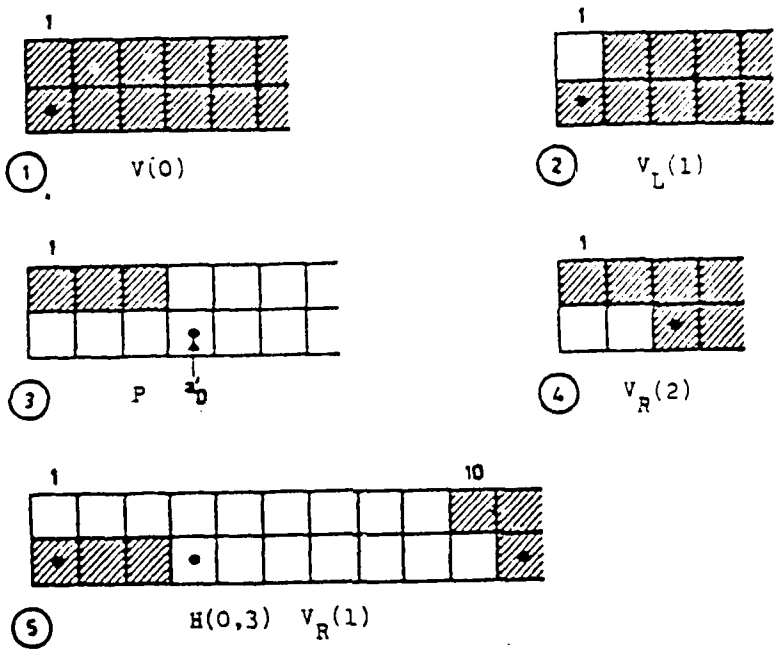


FIGURE 10 - Coding examples : first part of scanning line

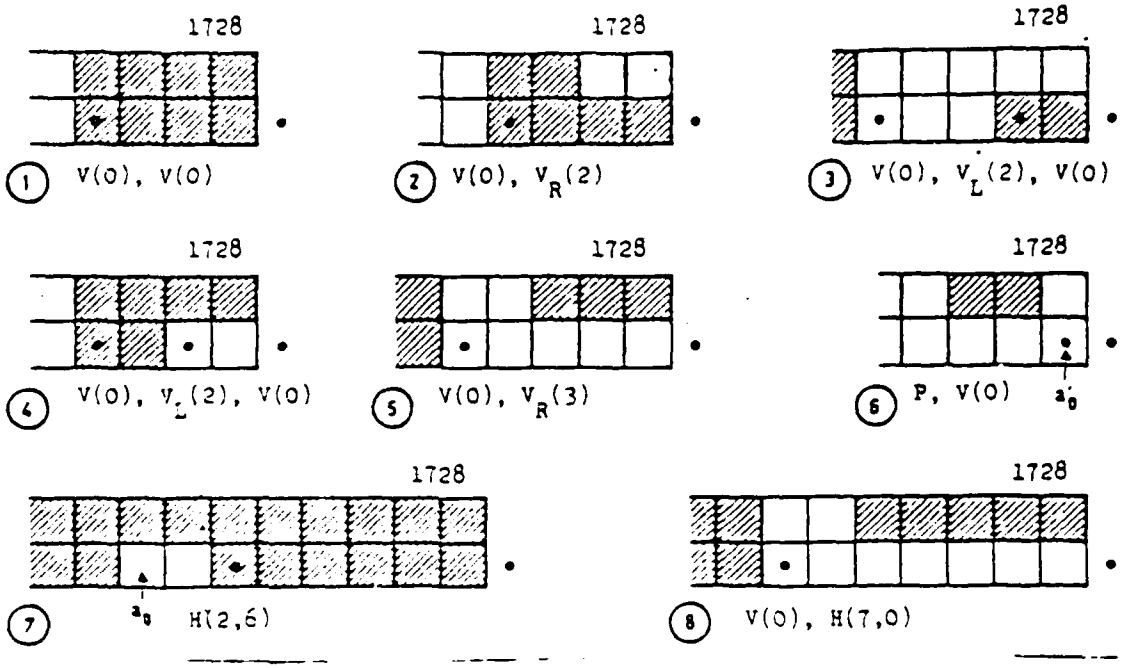


FIGURE 11 - Coding examples : last part of scanning line

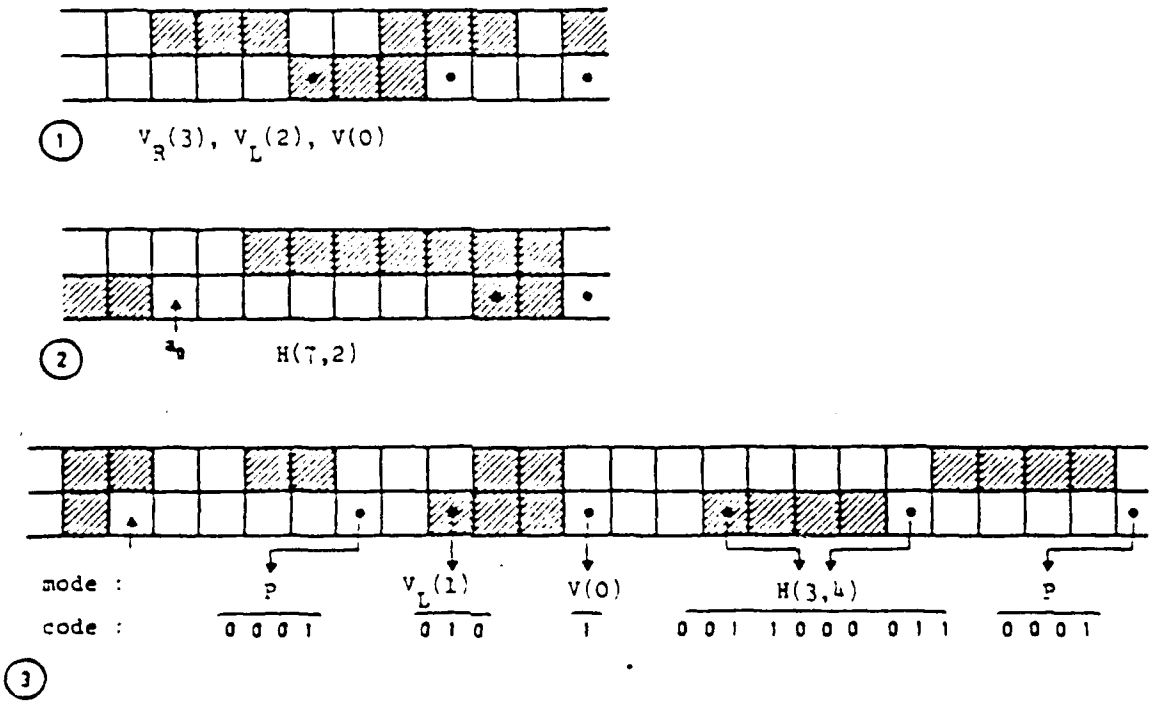


FIGURE 12 - Coding examples

5. Modulation and demodulation

Groups 3 apparatus operating on the general switched telephone network shall utilize the modulation, scrambler, equalization and timing signals defined in CCITT Recommendation V.27ter, specifically the preamble, sections 2, 3, 8, 10, 11 and the Appendix.

5.1 The training signal to be used shall be the long training sequence with protection against talker echo. (reference V.27ter, section 2.5.1 Table 3/V.27ter)

5.2 The data signalling rates to be used are 4800 bit/s and 2400 bit/s as defined in Recommendation V.27ter.

Note 1 Some Administrations pointed out that it would not be possible to guarantee the service at a data signalling rate higher than 2400 bit/s.

Note 2 It should be noted that there are equipments in service using, inter alia, other modulation methods.

Note 3 Where quality of communication service can successfully support higher speed operation, such as may be possible on leased circuits or high quality switched circuits, Group 3 apparatus may optionally utilize the modulation, scrambler, equalization and timing signals defined in CCITT Recommendation V.29, specifically sections 1, 2, 3, 4, 9, 10, 12 and 13. Under this option the data should be non-multiplexed and limited to the rates of 9600 bit/s and 7200 bit/s.

6. Power at the transmitter

The average power should be adjustable from -15dBm to 0dBm but the equipment should be so designed that there is no possibility of this adjustment being tampered with by an operator.

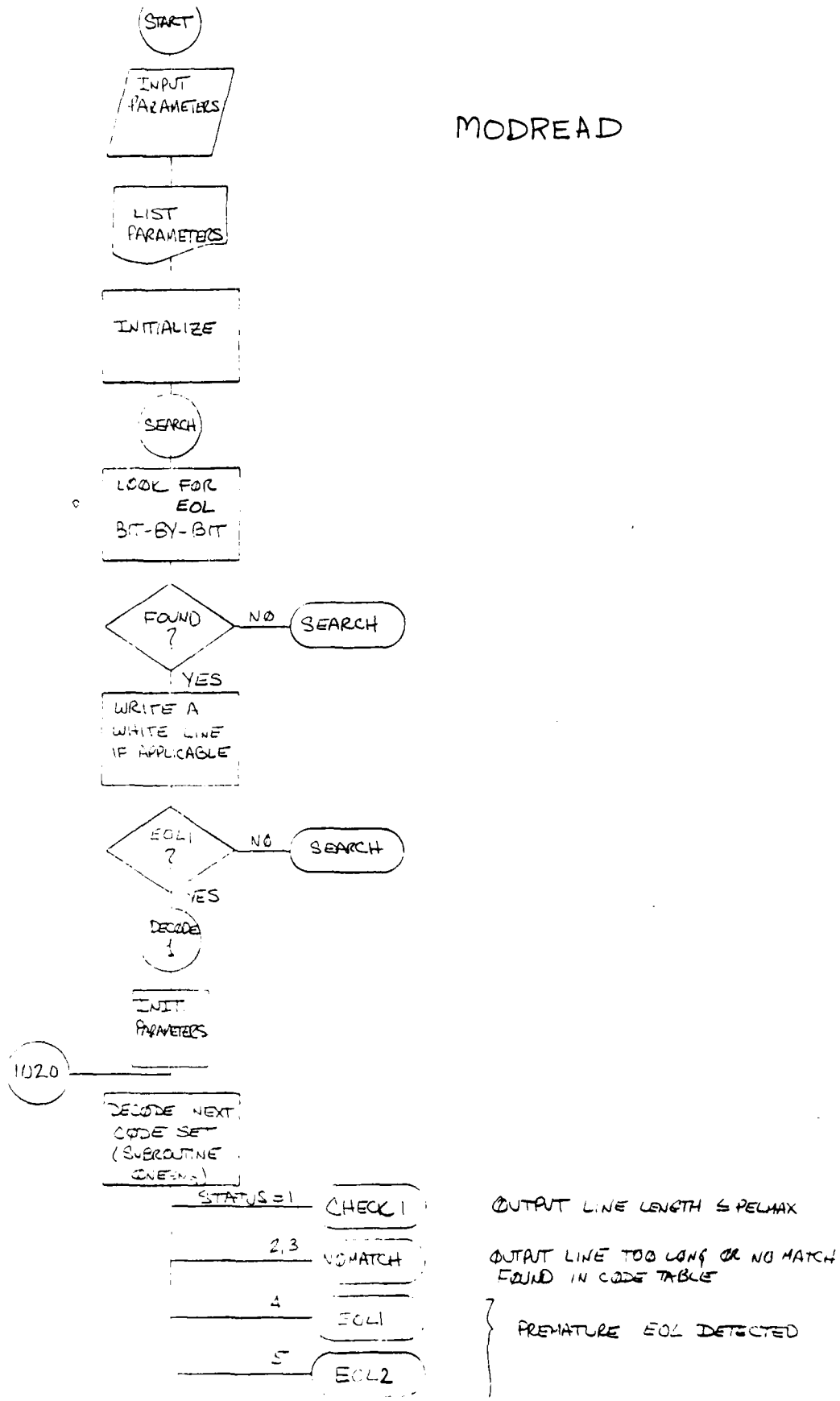
7. Power at the receiver input

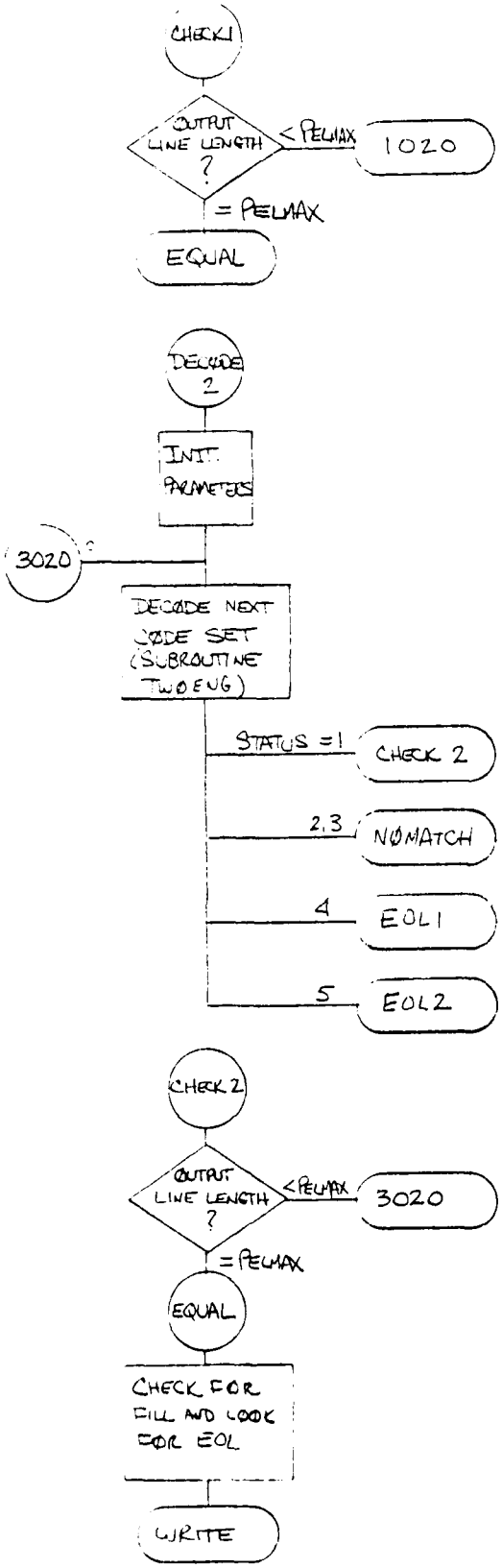
The receiving apparatus should be capable of functioning correctly when the received signal level is within the range of 0dBm to -40dBm. No control of receiver sensitivity should be provided for operator use.

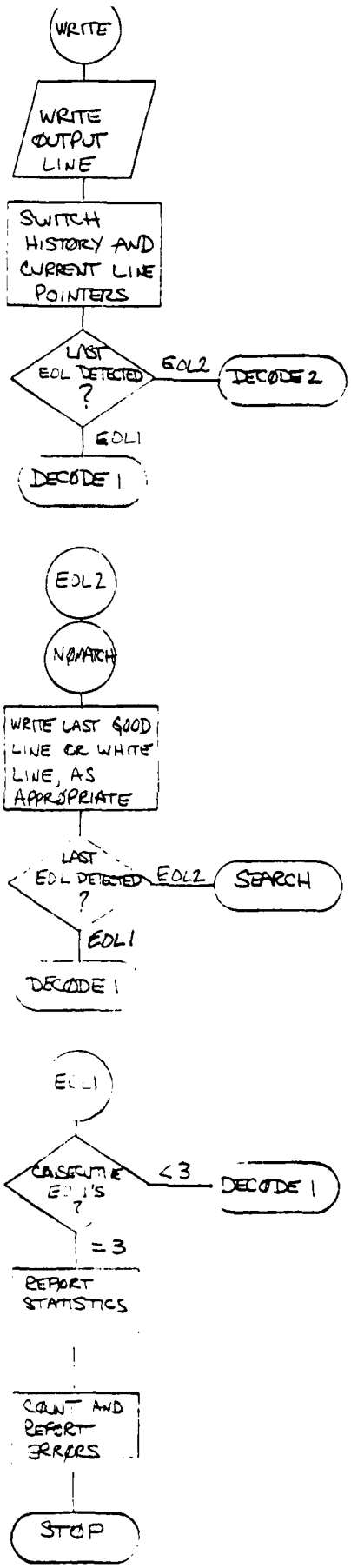
APPENDIX B

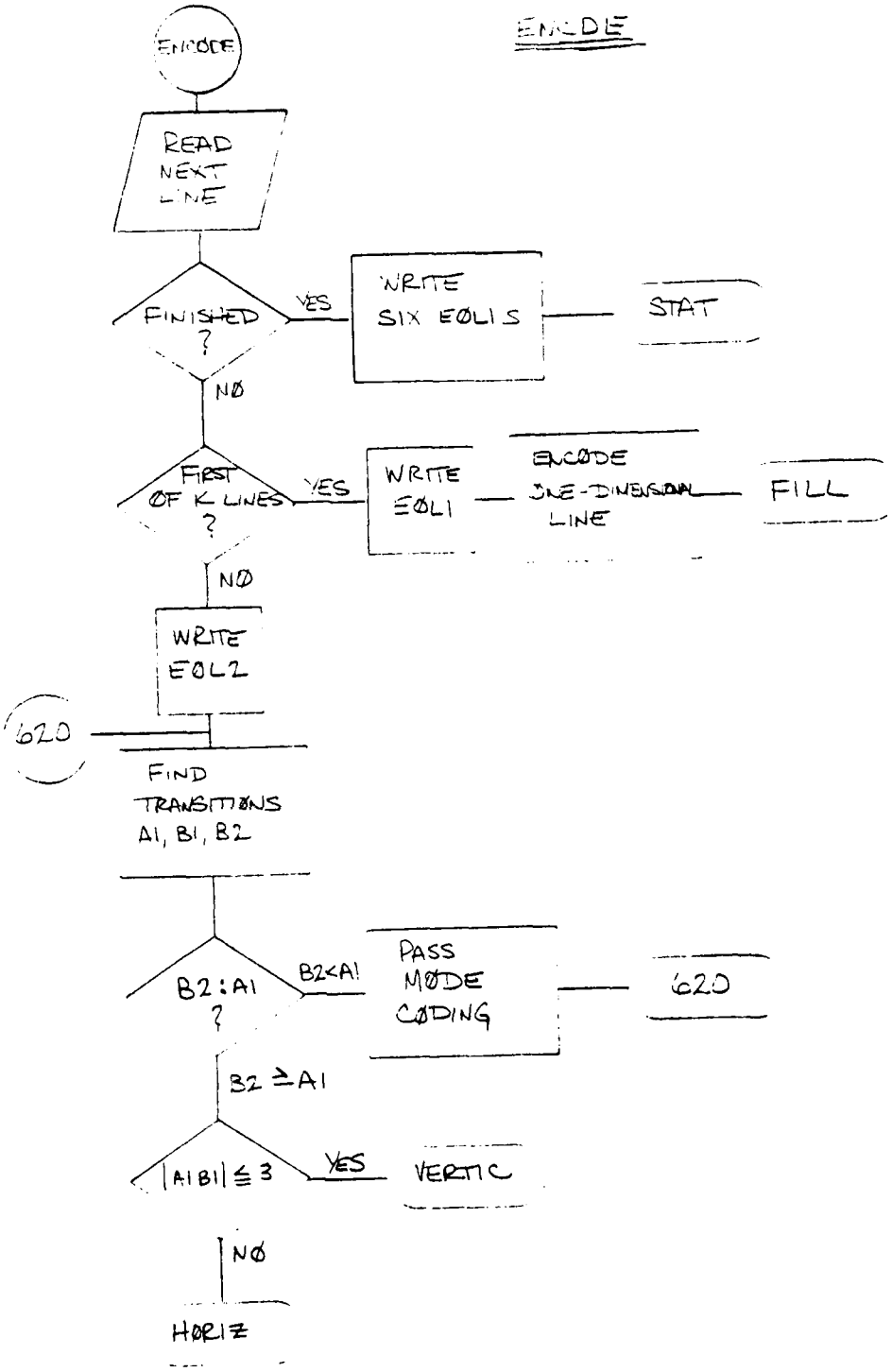
FLOW CHARTS FOR MODIFIED READ PROGRAM

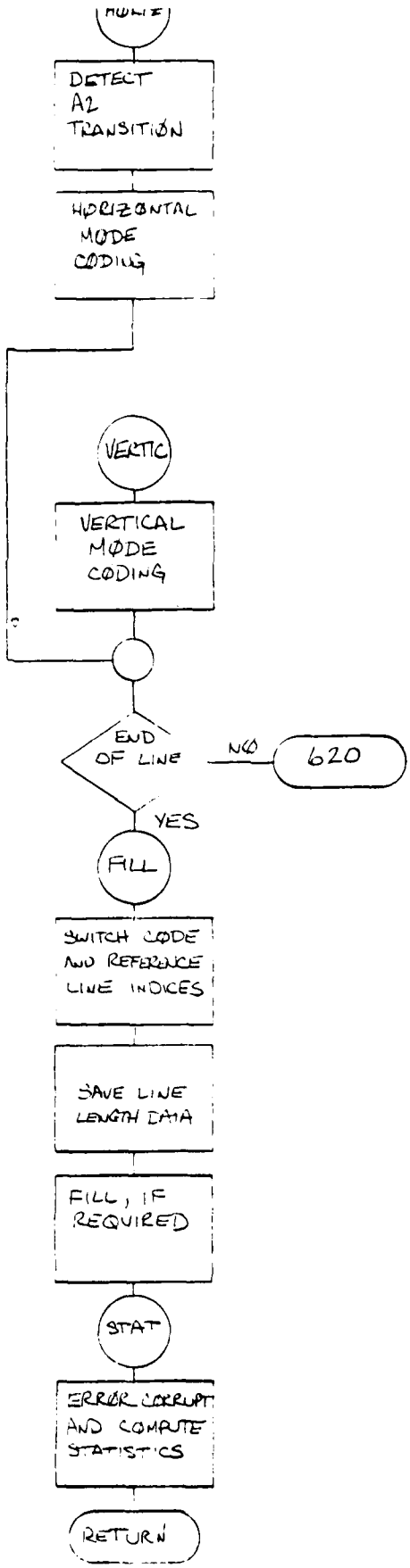
MODREAD


START
INPUT PARAMETERS
LIST PARAMETERS
INITIALIZE
SEARCH
LOOK FOR EOL BIT-BY-BIT
FOUND ?
NO
SEARCH
YES
WRITE A WHITE LINE IF APPLICABLE
EOL1 ?
NO
SEARCH
YES
DECODE 1
INIT PARAMETERS
1020
DECODE NEXT CODE SET (SUBROUTINE ONEDIM)
STATUS = 1
CHECK 1
OUTPUT LINE LENGTH ≤ PELMAX
2,3
NOMATCH
OUTPUT LINE TOO LONG OR NO MATCH FOUND IN CODE TABLE
4
EOL1
5
EOL2
PREMATURE EOL DETECTED

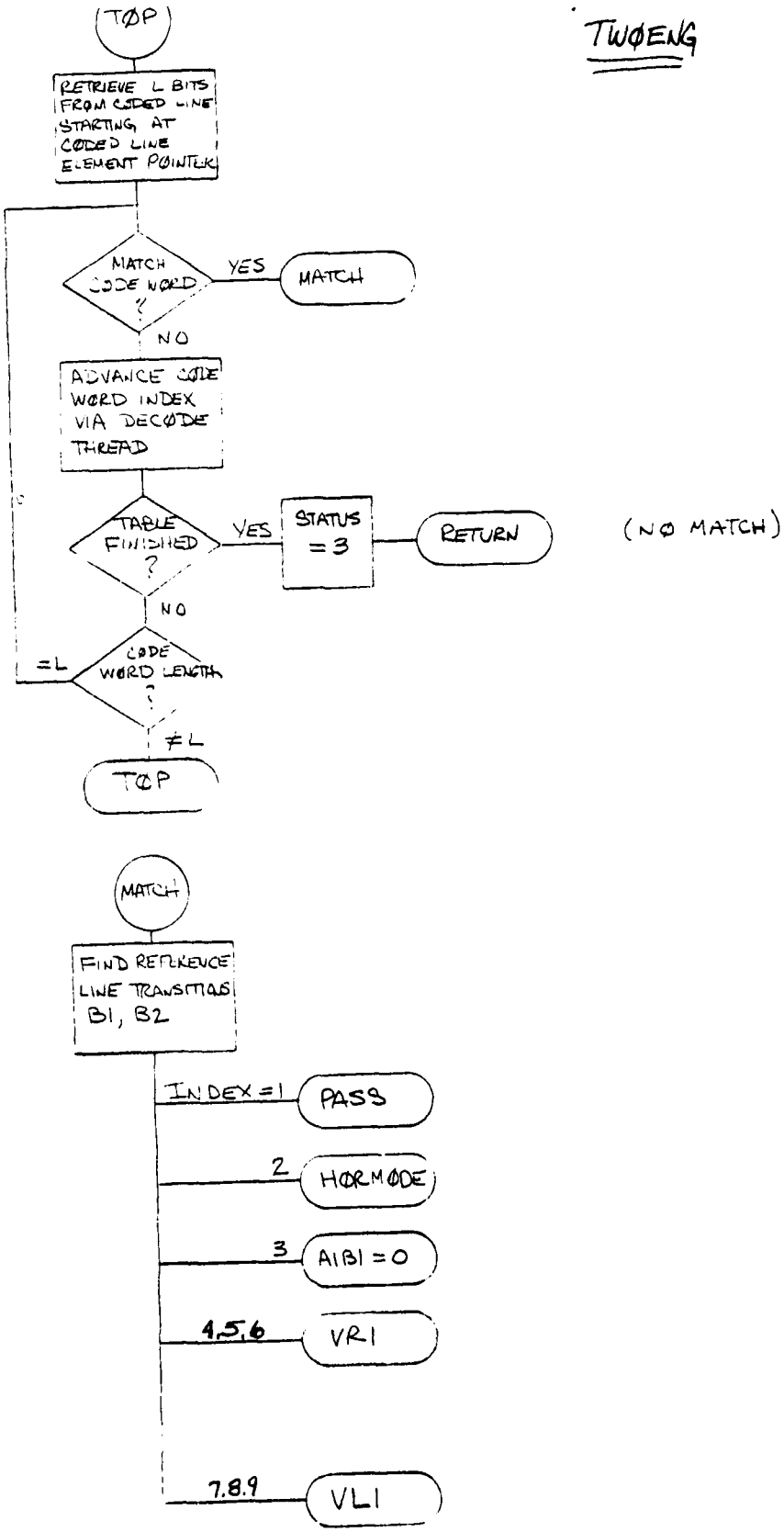

CHECK1
OUTPUT LINE LENGTH ?
< PELMAX
1020
= PELMAX
EQUAL
DECODE 2
INIT. PARMETERS
3020
DECODE NEXT CODE SET (SUBROUTINE TWOENG)
STATUS = 1
CHECK 2
2,3
NOMATCH
4
EOL1
5
EOL2
CHECK 2
OUTPUT LINE LENGTH ?
< PELMAX
3020
= PELMAX
EQUAL
CHECK FOR FILL AND LOOK FOR EOL
WRITE

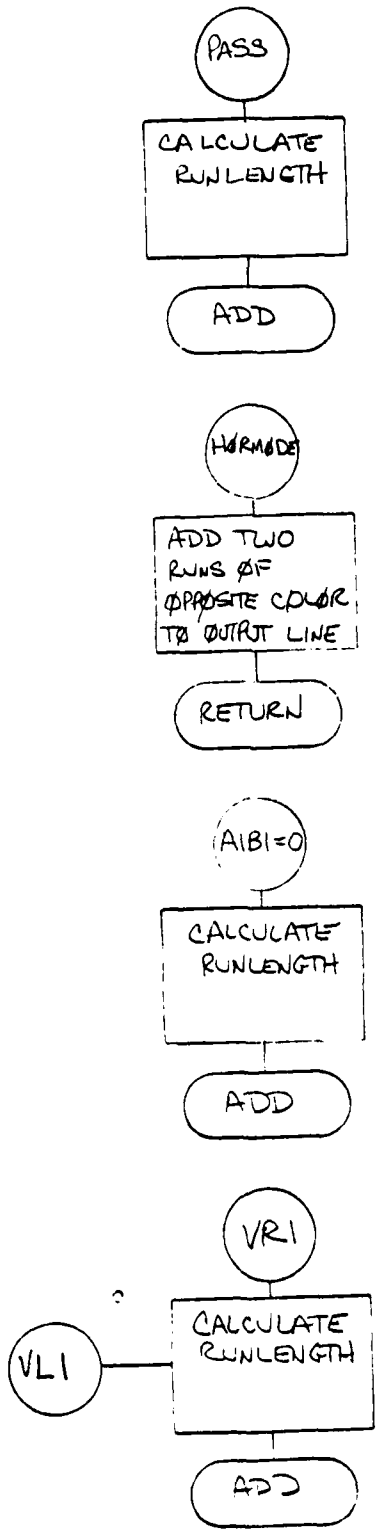

WRITE
WRITE OUTPUT LINE
SWITCH HISTORY AND CURRENT LINE POINTERS
LAST EOL DETECTED ?
EOL2
DECODE 2
EOL1
DECODE 1
EOL2
NOMATCH
WRITE LAST GOOD LINE OR WHITE LINE, AS APPROPRIATE
LAST EOL DETECTED ?
EOL2
SEARCH
EOL1
DECODE 1
EOL1
CONSECUTIVE EOL'S ?
< 3
DECODE 1
= 3
REPORT STATISTICS
COUNT AND REPORT ERRORS
STOP


ENCODE
ENCODE
READ NEXT LINE
FINISHED ?
YES
WRITE SIX EØLIS
STAT
NØ
FIRST ØF K LINES ?
YES
WRITE EØL1
ENCØDE ONE-DIMENSIONAL LINE
FILL
NØ
WRITE EØL2
620
FIND TRANSITIØNS A1, B1, B2
B2 : A1 ?
B2<A1
PASS MØDE CØDING
620
B2 ≥ A1
|A1 B1| ≤ 3
YES
VERTIC
NØ
HØRIZ


DETECT A2 TRANSITION
HORIZONTAL MODE CODING
VERTIC
VERTICAL MODE CODING
END OF LINE
NO
620
YES
FILL
SWITCH CODE AND REFERENCE LINE INDICES
SAVE LINE LENGTH DATA
FILL, IF REQUIRED
STAT
ERROR CORRUPT AND COMPUTE STATISTICS
RETURN

TWØENG


TØP
RETRIEVE L BITS FROM CODED LINE STARTING AT CODED LINE ELEMENT POINTER
MATCH CODE WORD ?
YES
MATCH
NO
ADVANCE CODE WORD INDEX VIA DECØDE THREAD
TABLE FINISHED ?
YES
STATUS = 3
RETURN
(NØ MATCH)
NO
CODE WORD LENGTH ?
=L
≠L
TØP
MATCH
FIND REFERENCE LINE TRANSITIONS B1, B2
INDEX = 1
PASS
2
HØRMØDE
3
A1B1 = 0
4,5,6
VR1
7,8,9
VL1


PASS
CALCULATE RUNLENGTH
ADD
HØRMØDE
ADD TWO RUNS ØF ØPPØSITE CØLØR TØ ØUTPUT LINE
RETURN
A1B1=0
CALCULATE RUNLENGTH
ADD
VR1
VL1
CALCULATE RUNLENGTH
ADD

APPENDIX C

CODE LISTING FOR THE MODIFIED READ PROGRAM

PROGRAM NAME	FUNCTION	PAGE
MODREAD	Main Program	C-1
CODELN	Line Code Subroutine of "Encode" Subroutine	C-18
ERRMES	Error Measurement Subroutine	C-19
MI2B	Packing Subroutine	C-21
IL4B	Unpacking Subroutine	C-22
STATS	Computer Statistics of Coded Lines	C-23

```
START OF DCEC UPRINT PROGRAM          DSNAME=F3802.MODREAD.FORT
C     PROGRAM MODREAD                                                   00000010
C     <6/02/80    ADDED VARIABLE FILEND TO LOGIC COMMON BLOCK>          00000011
C     <6/02/80    ADDED VARIABLE FILEND TO LOGICAL STATEMENT>           00000012
C     <6/02/80    ADDED FILEND ASSIGNMENT STATEMENT IN INITIALIZATIONS> 00000013
C     <6/02/80    ADDED PASSING PARAMETER TO ERRMES CALL-DIAG>          00000014
C     <6/02/80    ADDED PASSING PARAMETER TO CODELN CALL-CODE>          00000015
C     <6/02/80    CHANGED ERRORS DIMENSION FROM 9000 TO 10000>          00000016
C                                                                       00000017
      IMPLICIT INTEGER(A-Z)                                             00000020
      REAL CF3,CF4,ERRATE                                               00000030
C******* LABELED COMMON /G32BIT/ *******                                00000040
C                                                                       00000050
      COMMON /G32BIT/MASK(32),COMASK(32),LIBIT(32),LZBIT(32)            00000060
      INTEGER MASK,COMASK,LIBIT,LZBIT                                   00000070
C                                                                       00000080
      COMMON/BUFF/PELBUF(60,2),CDBUF(240),OTBUF(60,2),                  00000090
     *            STFBUF(240), STAT(3000)                               00000100
      COMMON/HUFF/CODE(3,92,2),CODERD(3,11)                             00000110
      COMMON/ERAY/ERRORS(10000)                                         00000120
C********************** FILE DEFINITIONS ********************          00000130
C                                                                       00000140
      COMMON/FILES/TERM,LPFIL,PELFIL,CTFIL,ERFIL                        00000150
C                                                                       00000160
C******************** LABELLED COMMON VARIABLES ********************** 00000170
C                                                                       00000180
      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPMAX,ERRMOD,LINMAX,K             00000190
      COMMON/PVAR/INLNNO,OTLNNO,OTFLW,INELP,CDELP,OTELP,CDELW,          00000200
     *            CDELCT,INELCT,TCDATA,TCDEL,ERRPNT,ERRCFF,ERRLIM,      00000210
     *            ERRCNT,INLNCT,CONSEC,ONECNT,LNNOBF,KCNT,              00000220
     *            INCOD,INSEF,OTCOD,OTSEF,STFBIT                        00000230
      COMMON/ICHAR/DD,II,MM,TT,NN,YY                                    00000240
      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,    00000250
     AFILEND                                                            00000251
      LOGICAL SEARCH,DIAG,SYNC,WRITE,LEFT,CHCOL,ONE,WHITE,FILEND        00000260
C                                                                       00000270
C   READ INPUT PARAMETERS                                               00000280
   90 WRITE(6,100)                                                      00000290
  100 FORMAT('$PARAMETERS: INPUT(=I), OR DEFAULT(=D)?')                 00000300
      READ(5,110,ERR=90) INSW                                           00000310
  110 FORMAT(A1)                                                        00000320
      IF (INSW.EQ.DD) GO TO 315                                         00000330
      IF (INSW.NE.II) GO TO 90                                          00000340
C                                                                       00000350
C   READ DIAGNOSTIC SWITCH                                              00000360
C                                                                       00000370
  114 WRITE(6,115)                                                      00000380
  115 FORMAT('$DIAGNOSTIC PRINTOUT? (Y OR N): ')                        00000390
      READ(5,110) INSW                                                  00000400
      IF(INSW.EQ.YY) GO TO 116                                          00000410
      IF(INSW.EQ.NN) GO TO 120                                          00000420
      GO TO 114                                                         00000430
  116 CONTINUE                                                          00000440
      DIAG=.TRUE.                                                       00000450
C                                                                       00000460
C   READ MAXIMUM NUMBER OF PELS PER LINE                                00000470
C                                                                       00000480
  120 CONTINUE                                                          00000490
      WRITE(6,130)                                                      00000500
  130 FORMAT('$ENTER MAXIMUM NUMBER OF PELS PER LINE: ')                00000510
      READ(5,140,ERR=120) PELMAX                                        00000520
  140 FORMAT(I4)                                                        00000530
      IF(PELMAX.GE.1.AND.PELMAX.LE.1728) GO TO 160                      00000540
      WRITE(6,150) PELMAX                                               00000550
  150 FORMAT('0NUMBER OUT OF RANGE (=',I6,')')                          00000560
      GO TO 120                                                         00000570
C                                                                       00000580
C   READ VERTICAL SAMPLING                                              00000590
C                                                                       00000600
  160 CONTINUE                                                          00000610
      WRITE(6,170)                                                      00000620
  170 FORMAT('$ENTER VERTICAL SAMPLING: ')                              00000630
      READ(5,180,ERR=160) VRES                                          00000640
  180 FORMAT(I2)                                                        00000650
      IF(VRES.GE.1.AND.VRES.LE.10) GO TO 190                            00000660
      WRITE(6,150) VRES                                                 00000670
      GO TO 160                                                         00000680
C                                                                       00000690
C   READ PARAMETER K                                                    00000700
C                                                                       00000710
  190 CONTINUE                                                          00000720
      WRITE(6,192)                                                      00000730
```

```
  190 FORMAT('$ENTER PARAMETER K: ')
      READ(5,140,ERR=190) K
      IF(K.GE.1.AND.K.LE.3000) GO TO 200
      WRITE(6,150) K
      GO TO 190
C
C   READ ERROR PATTERN PHASE
C
  200 CONTINUE
      WRITE(6,210)
  210 FORMAT('$ENTER ERROR PATTERN PHASE: ')
      READ(5,220,ERR=200) EPHASE
  220 FORMAT(I1)
      IF(EPHASE.GE.0.AND.EPHASE.LE.3) GO TO 240
      WRITE(6,150) EPHASE
      GO TO 200
C
C   READ MINIMUM COMPRESSED LINE LENGTH
C
  240 CONTINUE
      WRITE(6,250)
  250 FORMAT('$ENTER MINIMUM COMPRESSED LINE LENGTH: ')
      READ(5,140,ERR=240) CMPMAX
      IF(CMPMAX.GE.0.AND.CMPMAX.LE.1728) GO TO 320
      WRITE(6,150) CMPMAX
      GO TO 240
C
C   READ NUMBER OF SCAN LINES TO BE PROCESSED
  320 CONTINUE
      WRITE(6,330)
  330 FORMAT('$NUMBER OF SCAN LINES TO BE PROCESSED=? ')
      READ(5,140,ERR=320) LINMAX
      IF(LINMAX.GE.1.AND.LINMAX.LE.3000) GO TO 280
      WRITE(6,150) LINMAX
      GO TO 320
C
C   READ ERROR MODE
C
  280 CONTINUE
      WRITE(6,290)
  290 FORMAT('$ERROR MODE=? (M=MANUAL,T=TAPE,N=NO ERRORS)')
      READ(5,110,ERR=280) ERRMOD
      IF(ERRMOD.EQ.MM) GO TO 300
      IF(ERRMOD.EQ.TT) GO TO 315
      IF(ERRMOD.NE.NN) GO TO 280
      GO TO 350
C
C   READ ERROR LOCATIONS
C
  300 CONTINUE
      ERRLIM=1
  305 READ(5,140) ERRORS(ERRLIM)
      IF(ERRORS(ERRLIM).EQ.9999) GO TO 310
      ERRLIM=ERRLIM+1
      GO TO 305
  310 CONTINUE
      ERRLIM=ERRLIM-1
      GO TO 350
C
C   READ ERROR TAPE FILE AND OPEN
C
  315 CONTINUE
C
      ERRLIM=1
      READ(3,318,END=317) ERRORS(ERRLIM)
      ERRLIM=ERRLIM+1
  316 READ(3,318,END=317) ERRORS(ERRLIM)
  318 FORMAT(I16)
      ERRORS(ERRLIM)=ERRORS(ERRLIM)+ERRORS(ERRLIM-1)
      ERRLIM=ERRLIM+1
      GO TO 316
  317 ERRLIM=ERRLIM-1
C
  350 CONTINUE
C
  360 CONTINUE
C   WRITE INPUT PARAMETERS
C
      WRITE(6,400) PELMAX,VRES,K,EPHASE,CMPMAX,LINMAX
  400 FORMAT('1'INPUT PARAMETERS:'/
     *       '0MAXIMUM NUMBER OF PELS PER LINE=',I6/
     *       '0VERTICAL SAMPLING: N=',I4/
```

```
     *        '0PARAMETER K =',I4/                                          00001550
     *        '0ERROR PATTERN PHASE =',I4/                                  00001570
     *        '0MINIMUM COMPRESSED LINE LENGTH =',I4,' BITS'/               00001580
     *        '0NUMBER OF SCAN LINES TO BE PROCESSED =',I6)                 00001590
      IF(ERRMOD.EQ.NN) WRITE(6,410)                                         00001600
  410 FORMAT ('0NO ERRORS INSERTED')                                        00001610
      IF(ERRMOD.EQ.MM) WRITE(6,140) (ERRORS(I),I=1,ERPLIM)                  00001620
      IF(ERRMOD.EQ.TT) WRITE(6,420) ERRLIM                                  00001630
  420 FORMAT(I12,' ERRORS OBTAINED FROM ERROR TAPE')                        00001640
C******************** BEGIN PROGRAM ************************************** 00001650
C                                                                           00001660
C    INITIALIZE                                                             00001670
C                                                                           00001680
      TCDEL=0                                                               00001690
      TCDATA=0                                                              00001700
      ERRPNT=1                                                              00001710
      ERRCNT=0                                                              00001720
      INLNCT=0                                                              00001730
      ERROFF=EPHASE*1024                                                    00001740
      CDELCT=32                                                             00001750
      OTELP=1                                                               00001760
      CDELP=32+1                                                            00001770
      CONSEC=1                                                              00001780
      INREF=1                                                               00001790
      INCOD=2                                                               00001800
      OTREF=1                                                               00001810
      OTCOD=2                                                               00001820
      WHITE=.FALSE.                                                         00001830
      KCNT=1                                                                00001840
C                                                                           00001850
      DO 800 I=1,240                                                        00001860
      STEBUF(I)=0                                                           00001870
      CDBUF(I)=0                                                            00001880
  800 CONTINUE                                                              00001890
      DO 850 I=1,60                                                         00001900
      OTBUF(I,OTREF)=0                                                      00001910
      OTBUF(I,OTCOD)=0                                                      00001920
      PELBUF(I,INREF)=0                                                     00001930
      PELBUF(I,INCOD)=0                                                     00001940
  850 CONTINUE                                                              00001950
      SEARCH=.TRUE.                                                         00001960
      SYNC=.FALSE.                                                          00001970
      WRITE=.FALSE.                                                         00001980
C     <6/02/80   ADDED FOLLOWING STATEMENT>                                 00001981
C                                                                           00001982
      FILEND = .FALSE.                                                      00001983
C                                                                           00001990
C    SEARCH MODE: LOOK FOR EOL1 BIT-BY-BIT                                  00002000
C                                                                           00002010
  900 CONTINUE                                                              00002020
      CALL GETLE(13,MODE,LBITS,L)                                           00002030
      GO TO (910,930,930,920),MODE                                          00002040
      STOP 900                                                              00002050
  910 CONTINUE                                                              00002060
C                                                                           00002070
C    EOL NOT FOUND; ADVANCE POINTER AND TRY AGAIN                           00002080
C                                                                           00002090
      CDELP=CDELP+1                                                         00002100
      GO TO 900                                                             00002110
  920 CONTINUE                                                              00002120
      STOP 920                                                              00002130
  930 CONTINUE                                                              00002140
C                                                                           00002150
C    EOL FOUND                                                              00002160
C                                                                           00002170
      SEARCH=.FALSE.                                                        00002180
      CDELP=CDELP+L                                                         00002190
      IF(WRITE) GO TO 935                                                   00002200
      WRITE=.TRUE.                                                          00002210
      GO TO 960                                                             00002220
  935 CONTINUE                                                              00002230
C                                                                           00002240
C    SET OUTPUT DECODE LINE TO 0 AND WRITE OUT                              00002250
      DO 950 I=1,60                                                         00002260
      OTBUF(I,OTCOD)=0                                                      00002270
  950 CONTINUE                                                              00002280
      WRITE(2) OTLNNO,PELMAX,(OTBUF(I,OTCOD),I=1,60)                        00002290
      OTLANO=LNNOBF                                                         00002300
  960 CONTINUE                                                              00002310
      IF(MODE-2)965,1000,900                                                00002320
  965 STOP 965                                                              00002330
 1000 CONTINUE                                                              00002340
```

```
C
C     PERFORM ONE-DIMENSIONAL DECODE OF A COMPLETE LINE
C     FIRST,SET OUTPUT BUFFER TO WHITE
C     (ONLY BLACK RUNS WILL BE INSERTED)
C
      DO 1010 I=1,60
      OTBUF(I,OTCOD)=0
 1010 CONTINUE
C
      INDEX=3
      COLOR=1
      OTELP=1
C
 1020 CONTINUE
      CALL ONEENG(INDEX,COLOR,STATUS,L)
      GO TO (1030,1070,1070,1035,1040),STATUS
C                 1    2    3    4    5
      STOP 1000
C
C     RUN ADDED; CHECK LENGTH OF OUTPUT LINE
C
 1030 CONTINUE
      ONE=.TRUE.
      IF(OTELP-1-PELMAX) 1031,1032,1050
 1031 CONTINUE
      IF(CHCOL)COLOR=MOD(COLOR+2,2)+1
      INDEX=3
      GO TO 1020
 3000 CONTINUE
C
C     PERFORM TWO-DIMENSIONAL DECODE
C
C
C     FIRST,SET OUTPUT BUFFER TO WHITE
C     (ONLY BLACK RUNS WILL BE INSERTED)
C
      DO 3010 I=1,60
      OTBUF(I,OTCOD)=0
 3010 CONTINUE
C
      INDEX=3
      COLOR=1
      OTELP=1
C
 3020 CONTINUE
      CALL TWOENG(INDEX,COLOR,STATUS,L)
      GO TO (3030,1070,1070,1035,1040),STATUS
C                 1    2    3    4    5
      STOP 3000
C
C     RUN ADDED; LOOK FOR NEXT RUN
C
 3030 CONTINUE
      ONE=.FALSE.
      IF(OTELP-1-PELMAX) 3031,1032,1050
 3031 CONTINUE
      IF(CHCOL)COLOR=MOD(COLOR+2,2)+1
      INDEX=3
      GO TO 3020
C
C
C     LINE LENGTH=PELMAX; CHECK FOR FILL AND LOOK FOR EOL
C
 1032 CONTINUE
      ZERO=-1
 1033 CONTINUE
      ZERO=ZERO+1
      CALL GETLE(1,MODE,LBITS,L)
C
      GO TO (1034,1050,1050,1050),MODE
C
C     CHECK FOR FILL
C
 1034 CONTINUE
C
      CDELP=CDELP+L
      IF(LBITS.EQ.0) GO TO 1033
      IF(ZERO.LE.10) GO TO 1070
C
C     EOL FOUND; CHECK TYPE
C
      CALL GETLE(1,MODE,LBITS,L)
```

```
      IF(LBITS.EQ.1) MODE=2
      IF(LBITS.EQ.0) MODE=3
      GO TO (1070,1060,1060,1080),MODE
C
C   PREMATURE EOL DETECTED
C
C
C   EOL1 DETECTED
C
 1035 CONTINUE
      CDELP=CDELP+L
      STATUS=4
      IF(LTELP.LE.1) CONSEC=CONSEC+1
      IF(CONSEC-2)1080,1000,2000
C
C   EOL2 DETECTED
C
 1040 CONTINUE
      CDELP=CDELP+L
      STATUS=5
C
      GO TO 1080
C
C   PROBLEMS,PROBLEMS
C
 1050 STOP 1050
C
C   LINE LENGTH CORRECT, EOL DETECTED PROPERLY; WRITE OUTPUT LINE
C
 1060 CONTINUE
      CDELP=CDELP+L
      WRITE(2)OTLNNO,PELMAX,(OTBUF(I,OTCOD),I=1,60)
      OTLNNO=LNNOBF
      CONSEC=1
      IF(ONE) SYNC=.TRUE.
      TEMP=OTREF
      OTREF=OTCOD
      OTCOD=TEMP
      IF(MODE.EQ.2) GO TO 1000
      GO TO 3000
C
C   LINE TOO LONG OR NO MATCH
C
 1070 CONTINUE
      WRITE=.FALSE.
C
C   LINE SHORT
C
 1080 CONTINUE
      IF(.NOT.SYNC) GO TO 1090
C
C   WRITE LAST GOOD LINE
C
      WRITE(2) OTLNNO,PELMAX,(OTBUF(I,OTREF),I=1,60)
      SYNC=.FALSE.
      GO TO 1110
 1090 CONTINUE
C
C   WRITE A WHITE LINE
C
      DO 1100 I=1, 60
 1100 OTBUF(I,OTCOD)=0
      WRITE(2) OTLNNO,PELMAX,(OTBUF(I,OTCOD),I=1,60)
 1110 OTLNNO=LNNOBF
      IF(STATUS.EQ.4) GO TO 1000
      SEARCH=.TRUE.
      GO TO 900
C
C   END OF MESSAGE
C
 2000 CONTINUE
      WRITE(6,2010) CONSEC
 2010 FORMAT('0END OF MESSAGE DETECTED (',I2,' EOL''S)')
C
C   REPORT COMPRESSION FACTOR, ERROR SENSITIVITY FACTOR,BIT ERROR RATE
C
      ERRATE=FLOAT(ERCOUT)/FLOAT(TCDEL)
      WRITE(6,2020) TCDEL,TCDATA,STERIT,INLNCT,ERRATE
 2020 FORMAT('0TOTAL NUMBER OF CODED BITS = ',I8/
     *       '0TOTAL NUMBER OF CODED DATA BITS = ',I8/
     *       '0TOTAL NUMBER OF 2-DIM LINES = ',I8/
     *       '0TOTAL NUMBER OF INPUT LINES PROCESSED = ',I8/
```

```
     *        '0BIT ERROR RATE = ',G14.6)
C
      CALL STATS(STAT,INLNCT,DIAG)
      CF3=FLOAT(PELMAX)*FLOAT(INLNCT)/FLOAT(TCDEL)
      CF4=FLOAT(PELMAX)*FLOAT(INLNCT)/FLOAT(TCDATA)
C
      WRITE(6,2030) CF3,CF4
 2030 FORMAT('0COMPRESSION FACTOR FOR G3 MACHINE (CF3) =',F8.4/
     *        '0COMPRESSION FACTOR FOR G4 MACHINE (CF4) =',F8.4)
C
C     <6/02/80   ADDED PASSING PARAMETER TO ERRMES CALL-DIAG>
C
      CALL ERRMES(PELBUF,CTBUF,PELMAX,VRES,ERRCNT,DIAG)
C
      STOP
      END
      SUBROUTINE GETLE(LBITS,MODE,WRD,L)
      IMPLICIT INTEGER(A-Z)
C******* LABELED COMMON /G32BIT/ *******
C
      COMMON /G32BIT/MASK(32),COMASK(32),LIBIT(32),LZBIT(32)
      INTEGER MASK,COMASK,LIBIT,LZBIT
C
      COMMON/BUFF/PELBUF(60,2),CDBUF(240),CTBUF(60,2),
     *            STEBUF(240), STAT(3000)
      COMMON/BUFF/CODE(3,92,2),CODERD(3,11)
      COMMON/ERRAY/ERRORS(10000)
C******************** LABELLED COMMON VARIABLES ********************
C
      COMMON/IVAR/PELMAX,VRES,ERPHASE,CMPMAX,ERRMOD,LINMAX,K
      COMMON/EVAR/INLNNO,CTLNNO,CTELW,INELP,CDELP,CTELP,CDELW,
     *            CDELCT,INELCT,TCDATA,TCDEL,ERRCNT,ERRCEF,ERRLIM,
     *            ERRCNT,INLNCT,CONSEC,CNECNT,LNNOBF,KCNT,
     *            INCOD,INREF,CTCOD,CTREF,STEBIT
      COMMON/ICHAR/DD,II,MM,TT,NN,YY
      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,
     AFILEND
      LOGICAL SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,FILEND
C************************** BEGIN PROGRAM ***********************
C
      MODE=4
C
C    RETRIEVE NEXT BIT FROM CDBUF
C
 100  CONTINUE
C
C    ENCODE A NEW LINE IF NECESSARY
C
      IF(LBITS+CDELP-1.LE.CDELCT) GO TO 200
      IF(CDELCT-CDELP+1) 170,190,180
  170 STOP 170
  180 CONTINUE
      STEBUF(1)=IAB(STEBUF,CDELP,CDELCT-CDELP+1)
  190 CONTINUE
      CDELP=32-(CDELCT-CDELP)
      CALL ENCODE
 200  CONTINUE
      WRD=IAB(STEBUF,CDELP,LBITS)
      L=LBITS
      IF(L.LT.13) GO TO 250
      IF(L.EQ.13.AND.WRD.EQ.CODERD(3,10)) GO TO 300
      IF(L.EQ.13.AND.WRD.EQ.CODERD(3,11)) GO TO 400
  250 CONTINUE
      MODE=1
      RETURN
  300 CONTINUE
      MODE=2
      RETURN
  400 CONTINUE
      MODE=3
      RETURN
      END
      SUBROUTINE ENCODE
C
      IMPLICIT INTEGER(A-Z)
C
C******* LABELED COMMON /G32BIT/ *******
C
      COMMON /G32BIT/MASK(32),COMASK(32),LIBIT(32),LZBIT(32)
      INTEGER MASK,COMASK,LIBIT,LZBIT
C
      COMMON/BUFF/PELBUF(60,2),CDBUF(240),CTBUF(60,2),
```

```
     *              STFBUF(240), STAT(3000)
      COMMON/HUFF/CODE(3,92,2),CODERD(3,11)
      COMMON/ERAY/ERRORS(10000)
C********************** FILE DEFINITIONS ********************
C
      COMMON/FILES/TERM,LPFIL,PELFIL,OTFIL,ERFIL
C
C******************** LABELLED COMMON VARIABLES ********************
C
      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
      COMMON/PVAR/INLNNO,OTLNNO,OTELW,INELP,CDELP,OTELP,CDELW,
     *            CDELCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
     *            ERRCNT,INLNCT,CONSEC,ONECNT,LNNOBF,KCNT,
     *            INCOD,INREF,OTCOD,OTREF,STFBIT
      COMMON/ICHAR/DD,II,MM,TT,NN,YY
      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,
     AFILEND
      LOGICAL SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,FILEND
C
C********************************** BEGIN PROGRAM ********************
C
C    INITIALIZE VARIABLES
C
      KCNT=KCNT-1
      CDELCT=32
      CDDATA=0
      DO 50 I=2,240
      CDBUF(I)=0
      STFBUF(I)=0
   50 CONTINUE
C
C    READ INPUT PICTURE FILE
C
  100 CONTINUE
C
C     <6/02/80    ADDED FOLLOWING STATEMENT>
C
      IF(FILEND) GO TO 120
C
      READ(1,END=120,ERR=500)
     * INLNNO,INELCT,(PELBUF(I,INCOD),I=1,60)
      IF(MOD(INLNNO-1,VRES).NE.0) GO TO 100
      IF(INELCT.LT.PELMAX) CALL EXIT
      INLNCT=INLNCT+1
C
C    LOAD OUTPUT LINE NUMBER BUFFER
C
      LNNOBF=INLNNO
      IF(SEARCH)OTLNNO=LNNOBF
C
      IF(INLNNO.LE.LINMAX) GO TO 140
C
C    WRITE SIX EOL1'S
C
  120 CONTINUE
C
C     <6/02/80    ADDED FOLLOWING STATEMENT>
C
      FILEND = .TRUE.
C
      DO 130 I=1,6
      CALL CODENG(10,0,0,0,0,CDELCT,CDDATA)
  130 CONTINUE
      DO 135 I=1,6
      STFBUF(I)=CDBUF(I)
  135 CONTINUE
      GO TO 400
C
C    FIRST OF K LINES?
C
  140 CONTINUE
C
      IF(MOD(INLNCT-1,K).NE.0) GO TO 600
C    ONE-DIMENSIONAL CODING
C    WRITE ONE EOL1
C
      CALL CODENG(10,0,0,0,0,CDELCT,CDDATA)
C
      POLAR=1
C
C    TEST COLOR OF FIRST ELEMENT
C
```

```
      IF(IAB(PELBUF(1,INCOD),1,1).EQ.0) GO TO 150
C
C   FIRST ELEMENT BLACK: ENCODE 0-LENGTH WHITE RUN
C
      CALL CODELN(0,1,CDELCT,CDDATA,CODE)
      POLAR=2
C
C   CALCULATE RUN LENGTH AND ENCODE
C
  150 CONTINUE
      RUN=0
      DO 200 I=1,PELMAX
      PEL=IAB(PELBUF(1,INCOD),I,1)+1
      IF(PEL.EQ.POLAR) GO TO 180
      CALL CODELN(RUN,POLAR,CDELCT,CDDATA,CODE)
      IF(.NOT.DIAG) GO TO 170
      WRITE(6,160) RUN,POLAR,CDELCT,CDDATA
  160 FORMAT(4I8)
  170 CONTINUE
      RUN=1
      POLAR=MOD(POLAR+2,2)+1
      GO TO 200
  180 CONTINUE
      RUN=RUN+1
  200 CONTINUE
      CALL CODELN(RUN,POLAR,CDELCT,CDDATA,CODE)
      IF(.NOT.DIAG) GO TO 210
      WRITE(6,160) RUN,POLAR,CDELCT,CDDATA
      GO TO 210
C
C   TWO-DIMENSIONAL CODING
C
  600 CONTINUE
      STFBIT=STFBIT+1
C
C   WRITE ONE EOL2
C
      CALL CODENG(11,0,0,0,0,CDELCT,CDDATA)
C
C   SET A0 TO LEFT EDGE-1 AND POLARITY=WHITE
C
      A0=0
      POL=0
      LEFT=.TRUE.
C
C   DETECT A1
C
  620 CONTINUE
      I=A0+1
      IF(I.GT.PELMAX) GO TO 640
  630 CONTINUE
      PEL=IAB(PELBUF(1,INCOD),I,1)
      IF(PEL.NE.POL) GO TO 640
      I=I+1
      IF(I.LE.PELMAX) GO TO 630
  640 CONTINUE
      A1=I
C
C   DETECT B1
C
      I=A0+1
      IF(I.GT.PELMAX) GO TO 665
      IF(LEFT) GO TO 645
      PELM1=IAB(PELBUF(1,INREF),A0,1)
      GO TO 650
  645 PELM1=0
  650 CONTINUE
      PEL=IAB(PELBUF(1,INREF),I,1)
      IF(PEL.NE.PELM1) GO TO 670
 660  CONTINUE
      PELM1=PEL
      I=I+1
      IF(I.LE.PELMAX) GO TO 650
 665  CONTINUE
      B1=I
      GO TO 710
 670  CONTINUE
      IF(PEL.NE.POL) GO TO 690
      GO TO 660
 690  CONTINUE
      B1=I
      POL=PEL
```

```
C
C     DETECT B2
C
      I=B1+1
      IF(I.GT.PELMAX) GO TO 710
  700 CONTINUE
      PEL=I4B(PELBUF(1,INPEF),I,1)
      IF(PEL.NE.POL) GO TO 720
      I=I+1
      IF(I.LE.PELMAX) GO TO 700
  710 CONTINUE
      B2=I
      GO TO 730
  720 CONTINUE
      B2=I
      POL=PEL
  730 CONTINUE
      IF(.NOT.LEFT) POLAR=I4B(PELBUF(1,INCOD),A0,1)+1
      IF(.NOT.LEFT) GO TO 740
      POLAR=1
      A0=1
      LEFT=.FALSE.
  740 CONTINUE
C
C     TEST FOR PASS MODE
C
      IF(B2.GE.A1) GO TO 750
C
C     PASS MODE CODING (CAN'T END A LINE IN PASS MODE;NEW A0 MUST HAVE
C                       SAME POLARITY AS B2)
C
      CALL CODENG(1,0,0,0,0,CDELCT,CDDATA)
      A0=B2
      GO TO 620
  750 CONTINUE
C
      MAB=IABS(A1-B1)
      IF(MAB-3) 835,835,799
C
C     CODE BY HORIZONTAL MODE;FIRST DETECT A2
C
  799 CONTINUE
      I=A1+1
      IF(I.GT.PELMAX) GO TO 810
C
C     CALCULATE POLARITY OF A1
C
      POL=I4B(PELBUF(1,INCOD),A1,1)
  800 CONTINUE
      PEL=I4B(PELBUF(1,INCOD),I,1)
      IF(PEL.NE.POL) GO TO 820
      I=I+1
      IF(I.LE.PELMAX) GO TO 800
  810 A2=PELMAX+1
      GO TO 830
  820 CONTINUE
      A2=I
  830 CONTINUE
      CALL CODENG(2,POLAR,A0,A1,A2,CDELCT,CDDATA)
      A0=A2
      GO TO 960
C
C     CODE BY VERTICAL MODE
C
  835 CONTINUE
      IF(A1-B1) 850,840,840
C
  840 CALL CODENG(A1-B1+3,0,0,0,0,CDELCT,CDDATA)
      GO TO 950
  850 CONTINUE
      CALL CODENG(B1-A1+6,0,0,0,0,CDELCT,CDDATA)
  950 CONTINUE
      A0=A1
C
C     TEST FOR END OF LINE
C
  960 CONTINUE
      IF(A0.GT.PELMAX) GO TO 210
      POL=I4B(PELBUF(1,INCOD),A0,1)
      GO TO 620
  210 CONTINUE
C
```

```
C     SWITCH CODE & REFERENCE LINES
C
      TEMP=INREF
      INREF=INCOD
      INCOD=TEMP
C
      CDELW=(CDELCT+32-1)/32
      DO 300 I=2,CDELW
      STEBUF(I)=CDBUF(I)
  300 CONTINUE
C   SAVE LINE LENGTH(DATA BITS + EOL)
C
      STAT(INLNCT)=CODATA+13
C
C   CHECK CODED LINE LENGTH
C
      FILL=CMPMAX-(CDELCT-32)
      IF(FILL) 400,400,250
C
C   CODE LINE TOO SHORT; FILL IT TO CMPMAX
  250 CONTINUE
      CDELCT=CDELCT+FILL
C
C   ACCUMULATE STATISTICS AND ERROR CORRUPT
C
  400 CONTINUE
      IF(ERRMOD.EQ.NN) GO TO 390
C
C   ERROR CORRUPT
C
  350 CONTINUE
      ERRBIT=ERRORS(ERRPNT)-ERROFF-TCDEL
      IF(ERRBIT.LE.0) GO TO 360
      IF(ERRBIT.GT.CDELCT-32) GO TO 390
C
C   ERROR IN RANGE OF CODED LINE; CHANGE APPROPRIATE BIT
C
      BIT=I48(STEBUF,ERRBIT+32,1)
      BIT=MOD(BIT+1,2)
      CALL MI28(BIT,STEBUF,ERRBIT+32,1)
      ERRCNT=ERRCNT+1
C
C   INCREMENT ERROR LIST POINTER
C
  360 CONTINUE
      ERRPNT=ERRPNT+1
      IF(ERRPNT.LE.ERRLIM) GO TO 350
C
C   ERROR LIST EXHAUSTED
C
      ERRPNT=ERRPNT-1
      WRITE(6,370) ERRPNT,ERRORS(ERRPNT)
  370 FORMAT('0ERROR LIST EXHAUSTED AT',I10,'TH ERROR;'/
     *       ' LAST ERROR OCCURRED AT',I10,' BITS')
      ERRMOD=NN
C
C   COMPUTE STATISTICS
C
  390 CONTINUE
      TCDEL=TCDEL+CDELCT-32
      TCDATA=TCDATA+CODATA
      IF(DIAG) WRITE(6,160) INLNCT, CODATA
C
      IF (.NOT.DIAG) GO TO 460
      CDELW=(CDELCT+32-1)/32
      WRITE(6,450) (CDBUF(I),I=1,CDELW)
      WRITE(6,450) (STEBUF(I),I=1,CDELW)
  450 FORMAT(6Z12)
  460 CONTINUE
      RETURN
C
  500 CONTINUE
      CALL EXIT
C
      E N D
      SUBROUTINE CODENG(MODE,POLAR,A,B,C,CDELCT,CODATA)
      IMPLICIT INTEGER(A-Z)
      COMMON/BUFF/PELBUF(60,2),CDBUF(240),DTBUF(60,2),
     *            STEBUF(240), STAT(3000)
      COMMON/BUFF/CODE(3,92,2),CODERD(3,11)
      COMMON/ERAY/ERRORS(10000)
C
```

```
C***************** BEGIN PROGRAM *********************
C
      CALL MI2B(CODERD(3,MODE),CDBUF,CDELCT+1,CODERD(1,MODE))
      CDELCT=CDELCT+CODERD(1,MODE)
      GO TO (100,200,100,100,100,100,100,100,100,800,800) ,MODE
C
C   MODE      1   2   3   4   5   6   7   8   9  10  11
C
      STOP 129
C
C   PASS MODE(1),VERTICAL MODE:A1B1=0(3),A1B1=1(4,7),=2(5,8),=3(6,9)
C
 100  CONTINUE
      CDDATA=CDDATA+CODERD(1,MODE)
      RETURN
C
C   HORIZONTAL MODE(2)
C
  200 CONTINUE
      CDDATA=CDDATA+CODERD(1,MODE)
      CALL CODELN(B-A,POLAR,CDELCT,CDDATA,CODE)
      NEWPOL=MOD(POLAR+2,2)+1
      CALL CODELN(C-B,NEWPOL,CDELCT,CDDATA,CODE)
      RETURN
C
C   ADD ECL1 OR ECL2 TO LINE (10,11)
C
 800  CONTINUE
      RETURN
      END
      SUBROUTINE ONEENG(INDEX,COLOR,STATUS,L)
      IMPLICIT INTEGER(A-Z)
C******* LABELED COMMON /G32BIT/ *******
C
      COMMON /G32BIT/MASK(32),COMASK(32),LIBIT(32),LZBIT(32)
      INTEGER MASK,COMASK,LIBIT,LZBIT
C
      COMMON/BUFF/BELBUF(60,2),CDBUF(240),CTBUF(60,2),
     *            STEBUF(240), STAT(3000)
      COMMON/HUFF/CODE(3,92,2),CODERD(3,11)
      COMMON/ERAY/ERRORS(10000)
C********************** FILE DEFINITIONS ********************
C
      COMMON/FILES/TERM,LPFIL,PELFIL,CTFIL,ERFIL
C
C******************** LABELLED COMMON VARIABLES ********************
C
      COMMON/IVAR/BELMAX,VDES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
      COMMON/RVAR/INLNND,CTLNND,OTELW,INELP,CDELP,OTELP,CDELW,
     *            CDELCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
     *            ERRCNT,INLNCT,CONSEC,ONECNT,LNNOBF,KCNT,
     *            INCOD,INREF,CTCOD,OTREF,STEBIT
      COMMON/ICHAR/DD,II,MM,TT,NN,YY
      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,
     AFILEND
      LOGICAL SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,FILEND
C
C   BEGIN DECODE LOOP; RETRIEVE NEXT CODE WORD LENGTH (L)
C
 1000 CONTINUE
 1002 LENBIT=CODE(1,INDEX,COLOR)
      CALL GETLE(LENBIT,MODE,LBITS,L)
      IF(DIAG) WRITE(6,1003) LENBIT,MODE,LBITS,L
 1003 FORMAT(2I6,Z8,I6)
      GO TO (1040,1200,1205,1190), MODE
      STOP 1040
 1040 CONTINUE
      IF(LBITS.EQ.CODE(3,INDEX,COLOR)) GO TO 1100
C
C   NO MATCH; ADVANCE CODE WORD INDEX VIA DECODE THREAD
C
      INDEX=CODE(2,INDEX,COLOR)
      IF(INDEX.GE.93) GO TO 1190
      IF(CODE(1,INDEX,COLOR).EQ.LENBIT) GO TO 1040
C
C   CODE WORD LONGER; FROM THE TOP
C
      GO TO 1002
C
C   MATCH FOUND
C
 1100 CONTINUE
```

```
      CDELP=CDELP+L
C
C   NOT AN EOL
C
C
C   TEST FOR MAKE UP OR TERMINATING CODE
C
      RUNLEN=INDEX-1
      IF(INDEX.GE.65) RUNLEN=(INDEX-64)*64
      IF(RUNLEN.EQ.0) GO TO 1160
      IF(CCLOR.EQ.1) GO TO 1155
      IF(RUNLEN.LT.0) STOP 1100
C
C   ADD BLACK RUN TO OUTPUT BUFFER
C
      DO 1150 I=1,RUNLEN
      CALL MI2B(CCLOR-1,OTBUF(1,OTCOD),OTELP,1)
      OTELP=OTELP+1
      IF(OTELP-1.GT.PELMAX) GO TO 1180
 1150 CONTINUE
      GO TO 1160
C
C   ADD WHITE RUN TO OUTPUT BUFFER (BY DEFAULT)
C
 1155 CONTINUE
      OTELP=OTELP+RUNLEN
      IF(OTELP-1.GT.PELMAX) GO TO 1180
C
C OUTPUT LINE LESS THAN OR EQUAL TO MAX SPECIFIED
C
 1160 CONTINUE
      IF(INDEX.LT.65) GO TO 1170
      INDEX=3
      GO TO 1000
C
C   RUN ADDED TO OUTPUT LINE; LENGTH LESS THAN OR EQUAL TO PELMAX (1)
C
 1170 CONTINUE
      CHCCL=.TRUE.
      STATUS=1
      RETURN
C
C   RUN ADDED UNTIL PELMAX EXCEEDED; LINE TOO LONG (2)
C
 1180 CONTINUE
      IF(DIAG) WRITE(6,1185) (OTBUF(I,OTCOD),I=1,60)
 1185 FORMAT(6Z10)
      STATUS=2
      RETURN
C
C   NO MATCH FOUND IN CODE TABLE (3)
C
 1190 CONTINUE
      STATUS=3
      RETURN
C
C   EOL1 DETECTED (4)
C
 1200 CONTINUE
      STATUS=4
      RETURN
C
C   EOL2 DETECTED (5)
C
 1205 CONTINUE
      STATUS=5
      RETURN
      END
      SUBROUTINE TWOENG(INDEX,COLOR,STATUS,L)
      IMPLICIT INTEGER(A-Z)
C******* LABELED COMMON /G32BIT/ *******
C
      COMMON /G32BIT/MASK(32),COMASK(32),LIBIT(32),LZBIT(32)
      INTEGER MASK,COMASK,LIBIT,LZBIT
C
      COMMON/BUFF/PELBUF(60,2),CDBUF(240),OTBUF(60,2),
     *            STFBUF(240), STAT(3000)
      COMMON/HUFF/CODE(3,92,2),CODERD(3,11)
      COMMON/ERRY/ERRORS(10000)
C*********************** FILE DEFINITIONS ********************
C
      COMMON/FILES/TERM,LPFIL,PELFIL,OTFIL,ERFIL
```

```
C
C******************** LABELLED COMMON VARIABLES ********************
C
      COMMON/IVAR/PELMAX,VRES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
      COMMON/RVAR/INLNNO,CTLNNO,CTELW,INELP,CDELP,CTELP,CDELW,
     *            CDELCT,INELCT,TCDATA,TCDEL,ERRPNT,ERROFF,ERRLIM,
     *            ERRCNT,INLNCT,CONSEC,ONECNT,LNNOBF,KCNT,
     *            INCOD,INREF,CTCOD,CTREF,STFBIT
      COMMON/ICHAR/DD,II,MM,TT,NN,YY
      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,
     AFILEND
      LOGICAL SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,FILEND
C
C   BEGIN DECODE LOOP; RETRIEVE NEXT CODE WORD LENGTH (L)
C
 1000 CONTINUE
 1002 LENBIT=CODERD(1,INDEX)
      CALL GETLE(LENBIT,MODE,LBITS,L)
      IF(DIAG) WRITE(6,1003) LENBIT,MODE,LBITS,L
 1003 FORMAT(2I6,Z12,I6)
      GO TO (1040,1200,1205,1190), MODE
      STOP 1040
 1040 CONTINUE
      IF(LBITS.EQ.CODERD(3,INDEX)) GO TO 1100
C
C   NO MATCH; ADVANCE CODE WORD INDEX VIA DECODE THREAD
C
      INDEX=CODERD(2,INDEX)
      IF(INDEX.GE.12) GO TO 1190
      IF(CODERD(1,INDEX).EQ.LENBIT) GO TO 1040
C
C   CODE WORD LONGER; FROM THE TOP
C
      GO TO 1002
C
C   MATCH FOUND
C
 1100 CONTINUE
      CDELP=CDELP+L
C
C   NOT AN EOL
C
C
C   FIND B1 AND B2
C
      A0=CTELP
      IF(CTELP.EQ.1) A0=0
      PCL=COLOR-1
C
C   DETECT B1
C
      I=A0+1
      IF(I.GT.PELMAX) GO TO 65
      PELM1=0
      IF(A0.EQ.0) GO TO 50
      PELM1=I4B(CTBUF(1,CTREF),A0,1)
   50 CONTINUE
      PEL=I4B(CTBUF(1,CTREF),I,1)
      IF(PEL.NE.PELM1) GO TO 70
   60 CONTINUE
      PELM1=PEL
      I=I+1
      IF(I.LE.PELMAX) GO TO 50
   65 CONTINUE
      B1=I
      GO TO 92
   70 CONTINUE
      IF(PEL.NE.PCL) GO TO 90
      GO TO 60
   90 CONTINUE
      B1=I
      PCL=PEL
C
C   DETECT B2
C
      I=B1+1
      IF(I.GT.PELMAX) GO TO 92
   91 CONTINUE
      PEL=I4B(CTBUF(1,CTREF),I,1)
      IF(PEL.NE.PCL) GO TO 92
      I=I+1
      IF(I.LE.PELMAX) GO TO 91
```

```
   92 CONTINUE
      B2=I
      GO TO (100,200,300,400,400,400,600,600,600),INDEX
      STOP 100
C
C    PASS MODE
C
 100  CONTINUE
      RUNLEN=B2-CTELP
      CHCOL=.FALSE.
      GO TO (1155,1145),COLOR
C
C    HORIZONTAL MODE
C
 200  CONTINUE
      ENTRY=3
      CALL ONEENG(ENTRY,COLOR,STATE,L)
      GO TO (210,1180,1190,1200,1205),STATE
 210  CONTINUE
      COLOR=MOD(COLOR+2,2)+1
      ENTRY=3
      CALL ONEENG(ENTRY,COLOR,STATE,L)
      GO TO (220,1180,1190,1200,1205),STATE
 220  CONTINUE
      CHCOL=.TRUE.
      GO TO 1160
C
C    VERTICAL MODE A1B1=0
C
 300  CONTINUE
      RUNLEN=B1-CTELP
      CHCOL=.TRUE.
      GO TO (1155,1145),COLOR
C
C    VERTICAL MODE VR1 A1B1=1,2,3
C
 400  CONTINUE
      RUNLEN=B1-CTELP+INDEX-3
      CHCOL=.TRUE.
      GO TO (1155,1145),COLOR
C
C
C    VERTICAL MODE LEFT VL1 A1B1=1,2,3
C
 600  CONTINUE
      RUNLEN=B1-CTELP-(INDEX-6)
      CHCOL=.TRUE.
      GO TO (1155,1145),COLOR
C
C    ADD BLACK RUN TO OUTPUT BUFFER
C
 1145 CONTINUE
      IF(RUNLEN) 1190,1160,1147
 1147 CONTINUE
      DO 1150 I=1,RUNLEN
      CALL MI28(COLOR-1,OTBUF(1,OTCOD),CTELP,1)
      CTELP=CTELP+1
      IF(CTELP-1.GT.PELMAX) GO TO 1180
 1150 CONTINUE
      GO TO 1160
C
C    ADD WHITE RUN TO OUTPUT BUFFER (BY DEFAULT)
C
 1155 CONTINUE
      IF(RUNLEN.LT.0) GO TO 1190
      CTELP=CTELP+RUNLEN
      IF(CTELP-1.GT.PELMAX) GO TO 1180
C
C    RUN ADDED TO OUTPUT LINE; LENGTH LESS THAN OR EQUAL TO PELMAX (1)
C
 1160 CONTINUE
      STATUS=1
      RETURN
C
C    RUN ADDED UNTIL PELMAX EXCEEDED; LINE TOO LONG (2)
C
 1180 CONTINUE
      IF(DIAG) WRITE(6,1185) (OTBUF(I,OTCOD),I=1,60)
 1185 FORMAT(6Z10)
      STATUS=2
      RETURN
C
```

```
C    NO MATCH FOUND IN CODE TABLE (3)
C
 1190 CONTINUE
      STATUS=3
      RETURN
C
C    ECL1 DETECTED (4)
C
 1200 CONTINUE
      STATUS=4
      RETURN
C
C    ECL2 DETECTED (5)
C
 1205 CONTINUE
      STATUS=5
      RETURN
      E N D
      BLOCK DATA
C
      IMPLICIT INTEGER(A-Z)
C********************** FILE DEFINITIONS ******************
C
      COMMON/FILES/TERM,LPFIL,PELFIL,CTFIL,ERFIL
C
      COMMON/BUFF/PELBUF(60,2),CDBUF(240),CTBUF(60,2),
     *            STFBUF(240), STAT(3000)
      COMMON/HUFF/CODE(3,92,2),CODERD(3,11)
      COMMON/ERRAY/ERRORS(10000)
C******************** LABELLED COMMON VARIABLES ********************
C
      COMMON/IVAR/PELMAX,VLES,EPHASE,CMPMAX,ERRMOD,LINMAX,K
      COMMON/PVAR/INLNNO,CTLNNO,OTELW,INELP,CDELP,CTELP,CDELW,
     *            CDELCT,INELCT,TC[illegible]A,TCDEL,ERRPNT,ERROFF,ERRLIM,
     *            ERRCNT,INLNCT,CON[illegible],ONECNT,LNNCBF,KCNT,
     *            INCOD,INREF,CTCOD,[illegible]REF,STFBIT
      COMMON/ICHAR/DD,II,MM,TT,NN,YY
      COMMON/LOGIC/SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,
     AFILEND
      LOGICAL SEARCH,DIAG,SYNC,WRITE,ZERO,LEFT,CHCOL,ONE,WHITE,FILEND
C
      DATA TERM,LPFIL,PELFIL,CTFIL,ERFIL/5,6,1,2,3/
      DATA DD,II,MM,TT,NN,YY/'D','I','M','T','N','Y'/
      DATA PELMAX,VLES,EPHASE,CMPMAX,ERRMOD,LINMAX/1728,2,0,96,'T',3000/
      DATA K/2/
      DATA DIAG/.FALSE./
C
       DATA CODE(1,  1,1),CODE(2,  1,1),CODE(3,  1,1)/ 8, 70,Z0035/
       DATA CODE(1,  2,1),CODE(2,  2,1),CODE(3,  2,1)/ 6, 90,Z0007/
       DATA CODE(1,  3,1),CODE(2,  3,1),CODE(3,  3,1)/ 4,  4,Z0007/
       DATA CODE(1,  4,1),CODE(2,  4,1),CODE(3,  4,1)/ 4,  5,Z0008/
       DATA CODE(1,  5,1),CODE(2,  5,1),CODE(3,  5,1)/ 4,  6,Z0008/
       DATA CODE(1,  6,1),CODE(2,  6,1),CODE(3,  6,1)/ 4,  7,Z000C/
       DATA CODE(1,  7,1),CODE(2,  7,1),CODE(3,  7,1)/ 4,  8,Z000E/
       DATA CODE(1,  8,1),CODE(2,  8,1),CODE(3,  8,1)/ 4,  9,Z000F/
       DATA CODE(1,  9,1),CODE(2,  9,1),CODE(3,  9,1)/ 5, 10,Z0013/
       DATA CODE(1, 10,1),CODE(2, 10,1),CODE(3, 10,1)/ 5, 11,Z0014/
       DATA CODE(1, 11,1),CODE(2, 11,1),CODE(3, 11,1)/ 5, 12,Z0007/
       DATA CODE(1, 12,1),CODE(2, 12,1),CODE(3, 12,1)/ 5, 65,Z0008/
       DATA CODE(1, 13,1),CODE(2, 13,1),CODE(3, 13,1)/ 6, 14,Z0006/
       DATA CODE(1, 14,1),CODE(2, 14,1),CODE(3, 14,1)/ 6, 15,Z0003/
       DATA CODE(1, 15,1),CODE(2, 15,1),CODE(3, 15,1)/ 6, 16,Z0034/
       DATA CODE(1, 16,1),CODE(2, 16,1),CODE(3, 16,1)/ 6, 17,Z0035/
       DATA CODE(1, 17,1),CODE(2, 17,1),CODE(3, 17,1)/ 6, 18,Z002A/
       DATA CODE(1, 18,1),CODE(2, 18,1),CODE(3, 18,1)/ 6, 19,Z0029/
       DATA CODE(1, 19,1),CODE(2, 19,1),CODE(3, 19,1)/ 7, 20,Z0027/
       DATA CODE(1, 20,1),CODE(2, 20,1),CODE(3, 20,1)/ 7, 21,Z000C/
       DATA CODE(1, 21,1),CODE(2, 21,1),CODE(3, 21,1)/ 7, 22,Z0009/
       DATA CODE(1, 22,1),CODE(2, 22,1),CODE(3, 22,1)/ 7, 23,Z0017/
       DATA CODE(1, 23,1),CODE(2, 23,1),CODE(3, 23,1)/ 7, 24,Z0003/
       DATA CODE(1, 24,1),CODE(2, 24,1),CODE(3, 24,1)/ 7, 25,Z0004/
       DATA CODE(1, 25,1),CODE(2, 25,1),CODE(3, 25,1)/ 7, 26,Z0028/
       DATA CODE(1, 26,1),CODE(2, 26,1),CODE(3, 26,1)/ 7, 27,Z002B/
       DATA CODE(1, 27,1),CODE(2, 27,1),CODE(3, 27,1)/ 7, 28,Z0013/
       DATA CODE(1, 28,1),CODE(2, 28,1),CODE(3, 28,1)/ 7, 29,Z0024/
       DATA CODE(1, 29,1),CODE(2, 29,1),CODE(3, 29,1)/ 7, 68,Z0018/
       DATA CODE(1, 30,1),CODE(2, 30,1),CODE(3, 30,1)/ 8, 31,Z0002/
       DATA CODE(1, 31,1),CODE(2, 31,1),CODE(3, 31,1)/ 8, 32,Z0003/
       DATA CODE(1, 32,1),CODE(2, 32,1),CODE(3, 32,1)/ 8, 33,Z001A/
       DATA CODE(1, 33,1),CODE(2, 33,1),CODE(3, 33,1)/ 8, 34,Z0018/
       DATA CODE(1, 34,1),CODE(2, 34,1),CODE(3, 34,1)/ 8, 35,Z0012/
       DATA CODE(1, 35,1),CODE(2, 35,1),CODE(3, 35,1)/ 8, 36,Z0013/
```

```
DATA CODE(1, 36,1),CODE(2, 36,1),CODE(3, 36,1)/ 8, 37,Z0014/      00014160
DATA CODE(1, 37,1),CODE(2, 37,1),CODE(3, 37,1)/ 8, 38,Z0015/      00014170
DATA CODE(1, 38,1),CODE(2, 38,1),CODE(3, 38,1)/ 8, 39,Z0016/      00014180
DATA CODE(1, 39,1),CODE(2, 39,1),CODE(3, 39,1)/ 8, 40,Z0017/      00014190
DATA CODE(1, 40,1),CODE(2, 40,1),CODE(3, 40,1)/ 8, 41,Z0028/      00014200
DATA CODE(1, 41,1),CODE(2, 41,1),CODE(3, 41,1)/ 8, 42,Z0029/      00014210
DATA CODE(1, 42,1),CODE(2, 42,1),CODE(3, 42,1)/ 8, 43,Z002A/      00014220
DATA CODE(1, 43,1),CODE(2, 43,1),CODE(3, 43,1)/ 8, 44,Z002B/      00014230
DATA CODE(1, 44,1),CODE(2, 44,1),CODE(3, 44,1)/ 8, 45,Z002C/      00014240
DATA CODE(1, 45,1),CODE(2, 45,1),CODE(3, 45,1)/ 8, 46,Z002D/      00014250
DATA CODE(1, 46,1),CODE(2, 46,1),CODE(3, 46,1)/ 8, 47,Z0004/      00014260
DATA CODE(1, 47,1),CODE(2, 47,1),CODE(3, 47,1)/ 8, 48,Z0005/      00014270
DATA CODE(1, 48,1),CODE(2, 48,1),CODE(3, 48,1)/ 8, 49,Z000A/      00014280
DATA CODE(1, 49,1),CODE(2, 49,1),CODE(3, 49,1)/ 8, 50,Z000B/      00014290
DATA CODE(1, 50,1),CODE(2, 50,1),CODE(3, 50,1)/ 8, 51,Z0052/      00014300
DATA CODE(1, 51,1),CODE(2, 51,1),CODE(3, 51,1)/ 8, 52,Z0053/      00014310
DATA CODE(1, 52,1),CODE(2, 52,1),CODE(3, 52,1)/ 8, 53,Z0054/      00014320
DATA CODE(1, 53,1),CODE(2, 53,1),CODE(3, 53,1)/ 8, 54,Z0055/      00014330
DATA CODE(1, 54,1),CODE(2, 54,1),CODE(3, 54,1)/ 8, 55,Z0024/      00014340
DATA CODE(1, 55,1),CODE(2, 55,1),CODE(3, 55,1)/ 8, 56,Z0025/      00014350
DATA CODE(1, 56,1),CODE(2, 56,1),CODE(3, 56,1)/ 8, 57,Z0058/      00014360
DATA CODE(1, 57,1),CODE(2, 57,1),CODE(3, 57,1)/ 8, 58,Z0059/      00014370
DATA CODE(1, 58,1),CODE(2, 58,1),CODE(3, 58,1)/ 8, 59,Z005A/      00014380
DATA CODE(1, 59,1),CODE(2, 59,1),CODE(3, 59,1)/ 8, 60,Z005B/      00014390
DATA CODE(1, 60,1),CODE(2, 60,1),CODE(3, 60,1)/ 8, 61,Z004A/      00014400
DATA CODE(1, 61,1),CODE(2, 61,1),CODE(3, 61,1)/ 8, 62,Z004B/      00014410
DATA CODE(1, 62,1),CODE(2, 62,1),CODE(3, 62,1)/ 8, 63,Z0032/      00014420
DATA CODE(1, 63,1),CODE(2, 63,1),CODE(3, 63,1)/ 8, 64,Z0033/      00014430
DATA CODE(1, 64,1),CODE(2, 64,1),CODE(3, 64,1)/ 8, 65,Z0034/      00014440
DATA CODE(1, 65,1),CODE(2, 65,1),CODE(3, 65,1)/ 5, 66,Z0015/      00014450
DATA CODE(1, 66,1),CODE(2, 66,1),CODE(3, 66,1)/ 5, 67,Z0012/      00014460
DATA CODE(1, 67,1),CODE(2, 67,1),CODE(3, 67,1)/ 6,  2,Z0017/      00014470
DATA CODE(1, 68,1),CODE(2, 68,1),CODE(3, 68,1)/ 7, 30,Z0037/      00014480
DATA CODE(1, 69,1),CODE(2, 69,1),CODE(3, 69,1)/ 8,  1,Z0036/      00014490
DATA CODE(1, 70,1),CODE(2, 70,1),CODE(3, 70,1)/ 8, 71,Z0037/      00014500
DATA CODE(1, 71,1),CODE(2, 71,1),CODE(3, 71,1)/ 8, 72,Z0064/      00014510
DATA CODE(1, 72,1),CODE(2, 72,1),CODE(3, 72,1)/ 8, 73,Z0065/      00014520
DATA CODE(1, 73,1),CODE(2, 73,1),CODE(3, 73,1)/ 8, 74,Z0068/      00014530
DATA CODE(1, 74,1),CODE(2, 74,1),CODE(3, 74,1)/ 8, 75,Z0067/      00014540
DATA CODE(1, 75,1),CODE(2, 75,1),CODE(3, 75,1)/ 9, 76,Z00CC/      00014550
DATA CODE(1, 76,1),CODE(2, 76,1),CODE(3, 76,1)/ 9, 77,Z00CD/      00014560
DATA CODE(1, 77,1),CODE(2, 77,1),CODE(3, 77,1)/ 9, 78,Z00D2/      00014570
DATA CODE(1, 78,1),CODE(2, 78,1),CODE(3, 78,1)/ 9, 79,Z00D3/      00014580
DATA CODE(1, 79,1),CODE(2, 79,1),CODE(3, 79,1)/ 9, 80,Z00D4/      00014590
DATA CODE(1, 80,1),CODE(2, 80,1),CODE(3, 80,1)/ 9, 81,Z00D5/      00014600
DATA CODE(1, 81,1),CODE(2, 81,1),CODE(3, 81,1)/ 9, 82,Z00D6/      00014610
DATA CODE(1, 82,1),CODE(2, 82,1),CODE(3, 82,1)/ 9, 83,Z00D7/      00014620
DATA CODE(1, 83,1),CODE(2, 83,1),CODE(3, 83,1)/ 9, 84,Z00D8/      00014630
DATA CODE(1, 84,1),CODE(2, 84,1),CODE(3, 84,1)/ 9, 85,Z00D9/      00014640
DATA CODE(1, 85,1),CODE(2, 85,1),CODE(3, 85,1)/ 9, 86,Z00DA/      00014650
DATA CODE(1, 86,1),CODE(2, 86,1),CODE(3, 86,1)/ 9, 87,Z00DB/      00014660
DATA CODE(1, 87,1),CODE(2, 87,1),CODE(3, 87,1)/ 9, 88,Z00D8/      00014670
DATA CODE(1, 88,1),CODE(2, 88,1),CODE(3, 88,1)/ 9, 89,Z00D9/      00014680
DATA CODE(1, 89,1),CODE(2, 89,1),CODE(3, 89,1)/ 9, 91,Z009A/      00014690
DATA CODE(1, 90,1),CODE(2, 90,1),CODE(3, 90,1)/ 6, 13,Z0018/      00014700
DATA CODE(1, 91,1),CODE(2, 91,1),CODE(3, 91,1)/ 9, 92,Z009B/      00014710
DATA CODE(1, 92,1),CODE(2, 92,1),CODE(3, 92,1)/13, 93,Z0003/      00014720
DATA CODE(1,  1,2),CODE(2,  1,2),CODE(3,  1,2)/10, 65,Z0037/      00014730
DATA CODE(1,  2,2),CODE(2,  2,2),CODE(3,  2,2)/ 3,  6,Z0002/      00014740
DATA CODE(1,  3,2),CODE(2,  3,2),CODE(3,  3,2)/ 2,  4,Z0003/      00014750
DATA CODE(1,  4,2),CODE(2,  4,2),CODE(3,  4,2)/ 2,  5,Z0002/      00014760
DATA CODE(1,  5,2),CODE(2,  5,2),CODE(3,  5,2)/ 3,  2,Z0003/      00014770
DATA CODE(1,  6,2),CODE(2,  6,2),CODE(3,  6,2)/ 4,  7,Z0003/      00014780
DATA CODE(1,  7,2),CODE(2,  7,2),CODE(3,  7,2)/ 4,  8,Z0002/      00014790
DATA CODE(1,  8,2),CODE(2,  8,2),CODE(3,  8,2)/ 5,  9,Z0003/      00014800
DATA CODE(1,  9,2),CODE(2,  9,2),CODE(3,  9,2)/ 6, 10,Z0005/      00014810
DATA CODE(1, 10,2),CODE(2, 10,2),CODE(3, 10,2)/ 6, 11,Z0004/      00014820
DATA CODE(1, 11,2),CODE(2, 11,2),CODE(3, 11,2)/ 7, 12,Z0004/      00014830
DATA CODE(1, 12,2),CODE(2, 12,2),CODE(3, 12,2)/ 7, 13,Z0005/      00014840
DATA CODE(1, 13,2),CODE(2, 13,2),CODE(3, 13,2)/ 7, 14,Z0007/      00014850
DATA CODE(1, 14,2),CODE(2, 14,2),CODE(3, 14,2)/ 8, 15,Z0004/      00014860
DATA CODE(1, 15,2),CODE(2, 15,2),CODE(3, 15,2)/ 8, 16,Z0007/      00014870
DATA CODE(1, 16,2),CODE(2, 16,2),CODE(3, 16,2)/ 9, 17,Z0018/      00014880
DATA CODE(1, 17,2),CODE(2, 17,2),CODE(3, 17,2)/10, 18,Z0017/      00014890
DATA CODE(1, 18,2),CODE(2, 18,2),CODE(3, 18,2)/10, 19,Z0018/      00014900
DATA CODE(1, 19,2),CODE(2, 19,2),CODE(3, 19,2)/10,  1,Z0008/      00014910
DATA CODE(1, 20,2),CODE(2, 20,2),CODE(3, 20,2)/11, 21,Z0067/      00014920
DATA CODE(1, 21,2),CODE(2, 21,2),CODE(3, 21,2)/11, 22,Z0068/      00014930
DATA CODE(1, 22,2),CODE(2, 22,2),CODE(3, 22,2)/11, 23,Z006C/      00014940
DATA CODE(1, 23,2),CODE(2, 23,2),CODE(3, 23,2)/11, 24,Z0037/      00014950
DATA CODE(1, 24,2),CODE(2, 24,2),CODE(3, 24,2)/11, 25,Z0028/      00014960
DATA CODE(1, 25,2),CODE(2, 25,2),CODE(3, 25,2)/11, 26,Z0017/      00014970
```

```
      DATA CODE(1, 26,2),CODE(2, 26,2),CODE(3, 26,2)/11, 27,Z0018/
      DATA CODE(1, 27,2),CODE(2, 27,2),CODE(3, 27,2)/12, 28,Z00CA/
      DATA CODE(1, 28,2),CODE(2, 28,2),CODE(3, 28,2)/12, 29,Z00CB/
      DATA CODE(1, 29,2),CODE(2, 29,2),CODE(3, 29,2)/12, 30,Z00CC/
      DATA CODE(1, 30,2),CODE(2, 30,2),CODE(3, 30,2)/12, 31,Z00CD/
      DATA CODE(1, 31,2),CODE(2, 31,2),CODE(3, 31,2)/12, 32,Z0068/
      DATA CODE(1, 32,2),CODE(2, 32,2),CODE(3, 32,2)/12, 33,Z0069/
      DATA CODE(1, 33,2),CODE(2, 33,2),CODE(3, 33,2)/12, 34,Z006A/
      DATA CODE(1, 34,2),CODE(2, 34,2),CODE(3, 34,2)/12, 35,Z005B/
      DATA CODE(1, 35,2),CODE(2, 35,2),CODE(3, 35,2)/12, 36,Z00D2/
      DATA CODE(1, 36,2),CODE(2, 36,2),CODE(3, 36,2)/12, 37,Z00D3/
      DATA CODE(1, 37,2),CODE(2, 37,2),CODE(3, 37,2)/12, 38,Z00D4/
      DATA CODE(1, 38,2),CODE(2, 38,2),CODE(3, 38,2)/12, 39,Z00D5/
      DATA CODE(1, 39,2),CODE(2, 39,2),CODE(3, 39,2)/12, 40,Z00D6/
      DATA CODE(1, 40,2),CODE(2, 40,2),CODE(3, 40,2)/12, 41,Z00D7/
      DATA CODE(1, 41,2),CODE(2, 41,2),CODE(3, 41,2)/12, 42,Z006C/
      DATA CODE(1, 42,2),CODE(2, 42,2),CODE(3, 42,2)/12, 43,Z006D/
      DATA CODE(1, 43,2),CODE(2, 43,2),CODE(3, 43,2)/12, 44,Z00DA/
      DATA CODE(1, 44,2),CODE(2, 44,2),CODE(3, 44,2)/12, 45,Z00DB/
      DATA CODE(1, 45,2),CODE(2, 45,2),CODE(3, 45,2)/12, 46,Z0054/
      DATA CODE(1, 46,2),CODE(2, 46,2),CODE(3, 46,2)/12, 47,Z0055/
      DATA CODE(1, 47,2),CODE(2, 47,2),CODE(3, 47,2)/12, 48,Z0056/
      DATA CODE(1, 48,2),CODE(2, 48,2),CODE(3, 48,2)/12, 49,Z0057/
      DATA CODE(1, 49,2),CODE(2, 49,2),CODE(3, 49,2)/12, 50,Z0064/
      DATA CODE(1, 50,2),CODE(2, 50,2),CODE(3, 50,2)/12, 51,Z0065/
      DATA CODE(1, 51,2),CODE(2, 51,2),CODE(3, 51,2)/12, 52,Z0052/
      DATA CODE(1, 52,2),CODE(2, 52,2),CODE(3, 52,2)/12, 53,Z0053/
      DATA CODE(1, 53,2),CODE(2, 53,2),CODE(3, 53,2)/12, 54,Z0024/
      DATA CODE(1, 54,2),CODE(2, 54,2),CODE(3, 54,2)/12, 55,Z0037/
      DATA CODE(1, 55,2),CODE(2, 55,2),CODE(3, 55,2)/12, 56,Z0038/
      DATA CODE(1, 56,2),CODE(2, 56,2),CODE(3, 56,2)/12, 57,Z0027/
      DATA CODE(1, 57,2),CODE(2, 57,2),CODE(3, 57,2)/12, 58,Z0028/
      DATA CODE(1, 58,2),CODE(2, 58,2),CODE(3, 58,2)/12, 59,Z0058/
      DATA CODE(1, 59,2),CODE(2, 59,2),CODE(3, 59,2)/12, 60,Z0059/
      DATA CODE(1, 60,2),CODE(2, 60,2),CODE(3, 60,2)/12, 61,Z002B/
      DATA CODE(1, 61,2),CODE(2, 61,2),CODE(3, 61,2)/12, 62,Z002C/
      DATA CODE(1, 62,2),CODE(2, 62,2),CODE(3, 62,2)/12, 63,Z0058/
      DATA CODE(1, 63,2),CODE(2, 63,2),CODE(3, 63,2)/12, 64,Z0066/
      DATA CODE(1, 64,2),CODE(2, 64,2),CODE(3, 64,2)/12, 66,Z0067/
      DATA CODE(1, 65,2),CODE(2, 65,2),CODE(3, 65,2)/10, 20,Z000F/
      DATA CODE(1, 66,2),CODE(2, 66,2),CODE(3, 66,2)/12, 67,Z00C8/
      DATA CODE(1, 67,2),CODE(2, 67,2),CODE(3, 67,2)/12, 68,Z00C9/
      DATA CODE(1, 68,2),CODE(2, 68,2),CODE(3, 68,2)/12, 69,Z005B/
      DATA CODE(1, 69,2),CODE(2, 69,2),CODE(3, 69,2)/12, 70,Z0033/
      DATA CODE(1, 70,2),CODE(2, 70,2),CODE(3, 70,2)/12, 71,Z0034/
      DATA CODE(1, 71,2),CODE(2, 71,2),CODE(3, 71,2)/12, 72,Z0035/
      DATA CODE(1, 72,2),CODE(2, 72,2),CODE(3, 72,2)/13, 73,Z006C/
      DATA CODE(1, 73,2),CODE(2, 73,2),CODE(3, 73,2)/13, 74,Z006D/
      DATA CODE(1, 74,2),CODE(2, 74,2),CODE(3, 74,2)/13, 75,Z004B/
      DATA CODE(1, 75,2),CODE(2, 75,2),CODE(3, 75,2)/13, 76,Z004B/
      DATA CODE(1, 76,2),CODE(2, 76,2),CODE(3, 76,2)/13, 77,Z004C/
      DATA CODE(1, 77,2),CODE(2, 77,2),CODE(3, 77,2)/13, 78,Z004D/
      DATA CODE(1, 78,2),CODE(2, 78,2),CODE(3, 78,2)/13, 79,Z0072/
      DATA CODE(1, 79,2),CODE(2, 79,2),CODE(3, 79,2)/13, 80,Z0073/
      DATA CODE(1, 80,2),CODE(2, 80,2),CODE(3, 80,2)/13, 81,Z0074/
      DATA CODE(1, 81,2),CODE(2, 81,2),CODE(3, 81,2)/13, 82,Z0075/
      DATA CODE(1, 82,2),CODE(2, 82,2),CODE(3, 82,2)/13, 83,Z0076/
      DATA CODE(1, 83,2),CODE(2, 83,2),CODE(3, 83,2)/13, 84,Z0077/
      DATA CODE(1, 84,2),CODE(2, 84,2),CODE(3, 84,2)/13, 85,Z0052/
      DATA CODE(1, 85,2),CODE(2, 85,2),CODE(3, 85,2)/13, 86,Z0053/
      DATA CODE(1, 86,2),CODE(2, 86,2),CODE(3, 86,2)/13, 87,Z0054/
      DATA CODE(1, 87,2),CODE(2, 87,2),CODE(3, 87,2)/13, 88,Z0055/
      DATA CODE(1, 88,2),CODE(2, 88,2),CODE(3, 88,2)/13, 89,Z005A/
      DATA CODE(1, 89,2),CODE(2, 89,2),CODE(3, 89,2)/13, 90,Z005B/
      DATA CODE(1, 90,2),CODE(2, 90,2),CODE(3, 90,2)/13, 91,Z0064/
      DATA CODE(1, 91,2),CODE(2, 91,2),CODE(3, 91,2)/13, 93,Z0065/
      DATA CODE(1, 92,2),CODE(2, 92,2),CODE(3, 92,2)/13, 93,Z0003/
      DATA CODERD(1,1),CODERD(2,1),CODERD(3,1)/  4,5,Z1/
      DATA CODERD(1,2),CODERD(2,2),CODERD(3,2)/  3,4,Z1/
      DATA CODERD(1,3),CODERD(2,3),CODERD(3,3)/  1,2,Z1/
      DATA CODERD(1,4),CODERD(2,4),CODERD(3,4)/  3,7,Z3/
      DATA CODERD(1,5),CODERD(2,5),CODERD(3,5)/  6,8,Z3/
      DATA CODERD(1,6),CODERD(2,6),CODERD(3,6)/  7,9,Z3/
      DATA CODERD(1,7),CODERD(2,7),CODERD(3,7)/  3,1,Z2/
      DATA CODERD(1,8),CODERD(2,8),CODERD(3,8)/  6,6,Z2/
      DATA CODERD(1,9),CODERD(2,9),CODERD(3,9)/  7,10,Z2/
      DATA CODERD(1,10),CODERD(2,10),CODERD(3,10)/  13,11,Z3/
      DATA CODERD(1,11),CODERD(2,11),CODERD(3,11)/  13,12,Z2/
C
      E N D
0        END OF DCEC UPPRINT PROGRAM          LINES PRINTED=  1391
```

```
START OF DCEC UPRINT PROGRAM          DSNAME=N0026.CODELN.FORT
      SUBROUTINE CODELN(LENGTH,POLAR,CDELCT,CDDATA,CODE)
C
C     <2/14/80 ADDED PASSING PARAMETER - CODE>
C     <2/14/80 ADDED INTEGER CODE(3,92,2)>
C     <2/14/80 DELETED COMMON/HUFF/CODE(3,92,2),CODE(3,11)>
C     <2/14/80 DELETED COMMON/ERAY/ERRORS(10000)>
C
      IMPLICIT INTEGER(A-Z)
      INTEGER CODE(3,92,2)
      COMMON/BUFF/RELBUF(60,2),CDBUF(240),CTBUF(60,2),
     *            STEBUF(240),STAT(3000)
C
C******************************** BEGIN PROGRAM *********************
C
C   INITIALIZE MAKE UP CODE, MAKE UP CODE LENGTH
C
      MCODE=0
      MLENG=0
C
C   CHECK INPUTS
C
      IF(POLAR.LT.1.OR.POLAR.GT.2) CALL EXIT
      IF(LENGTH.LT.0.OR.LENGTH.GT.1728) CALL EXIT
C
      IF(LENGTH.LE.63) GO TO 10
C
C   CALCULATE MAKE UP CODE INDEX, CODE, LENGTH
C    AND WRITE TO CODE LINE
C
      INDEX=LENGTH/64+64
      MCODE=CODE(3,INDEX,POLAR)
      MLENG=CODE(1,INDEX,POLAR)
      CALL MI2B(MCODE,CDBUF,CDELCT+1,MLENG)
      CDELCT=CDELCT+MLENG
      CDDATA=CDDATA+MLENG
C
C   CALCULATE TERMINATING CODE INDEX, CODE, LENGTH
C    AND ADD TO CODE LINE
C
   10 CONTINUE
      INDEX=MOD(LENGTH,64)+1
      TCODE=CODE(3,INDEX,POLAR)
      TLENG=CODE(1,INDEX,POLAR)
      CALL MI2B(TCODE,CDBUF,CDELCT+1,TLENG)
      CDELCT=CDELCT+TLENG
      CDDATA=CDDATA+TLENG
C
      RETURN
      END
0          END OF DCEC UPRINT PROGRAM          LINES PRINTED=    49
```

```
START OF DCEC UPRINT PROGRAM          DSNAME=N0026.ERRMES.FORT
      SUBROUTINE ERRMES(PELBUF,OTBUF,PELMAX,VRES,ERRCNT,DIAG)
C
C     <2/14/80 ADDED PASSING PARAMETER - DIAG>
C     <2/14/80 DELETED COMMON/LOGIC/SEARCH,DIAG>
C     <2/14/80 CHANGED LOGICAL SEARCH,DIAG TO LOGICAL DIAG>
C
      IMPLICIT INTEGER(A-Z)
      REAL ESF
C******* LABELED COMMON /G32BIT/ *******
C
      COMMON /G32BIT/MASK(32),COMASK(32),LIBIT(32),LZBIT(32)
      INTEGER MASK,COMASK,LIBIT,LZBIT
C
C*********************** FILE DEFINITIONS ********************
C
      COMMON/FILES/TERM,LPFIL,PELFIL,OTFIL,ERFIL
C
      DIMENSION PELBUF(60), OTBUF(60)
      LOGICAL DIAG
C
C*************************** BEGIN PROGRAM ****************************
C
      REWIND 1
      REWIND 2
      ERRCR=0
      OTELW=(PELMAX+32-1)/32
      OTLNCT=0
C
C   READ AN ERROR FREE LINE
C
  100 CONTINUE
      READ(1,END=600,ERR=800) INLNNO,INELCT,PELBUF
      IF(MOD(INLNNO-1,VRES).NE.0) GO TO 100
C
C   READ AN ERROR-CORRUPTED LINE
C
  200 CONTINUE
      READ(2,END=500,ERR=800) OTLNNO,OTELCT,OTBUF
      OTLNCT=OTLNCT+1
  300 CONTINUE
C
C   COUNT DIFFERENCES BETWEEN TRANSMITTED AND RECEIVED LINES
C
      DO 450 I=1,OTELW
      IF(OTBUF(I).EQ.PELBUF(I)) GO TO 450
      IF(.NOT.DIAG) GO TO 420
      WRITE(6,410) INLNNO,OTLNNO,I,PELBUF(I),OTBUF(I)
  410 FORMAT(3I8,2Z12)
  420 CONTINUE
      DO 440 J=1,32
      IF(IAB(OTBUF(I),J,1).NE.IAB(PELBUF(I),J,1)) ERRCR=ERRCR+1
  440 CONTINUE
  450 CONTINUE
C
C     <2/28/80 CHANGED 200,100,550 TO 200,100,480>
      IF(OTLNNO-INLNNO) 200,100,480
C
C     <2/28/80  ADDED NEXT 3 LINES>
  480 READ(1,END=600,ERR=800) INLNNO,INELCT,PELBUF
      IF(MOD(INLNNO-1,VRES).NE.0) GO TO 480
      GO TO 300
C
C   ERROR LINE NUMBER GREATER THAN GOOD LINE NUMBER;
C   COUNT DIFFERENCES BETWEEN GOOD AND ALL WHITE LINE
C
  500 CONTINUE
      DO 550 I=1,OTELW
      IF(PELBUF(I).EQ.0) GO TO 550
      IF(.NOT.DIAG) GO TO 520
      WRITE(6,410) INLNNO,OTLNNO,I,PELBUF(I),OTBUF(I)
  520 CONTINUE
      DO 540 J=1,32
      IF(IAB(PELBUF(I),J,1).NE.0) ERRCR=ERRCR+1
  540 CONTINUE
  550 CONTINUE
C
  580 READ(1,END=600,ERR=800) INLNNO,INELCT,PELBUF
      IF(MOD(INLNNO-1,VRES).NE.0) GO TO 580
C
C     <2/26/80 CHANGED GO TO 300 TO GO TO 500>
      GO TO 500
```

```
C
C   CALCULATE ERROR SENSITIVITY FACTOR
C
  600 CONTINUE
      ESF=0.
      IF(ERRCNT.LE.0) GO TO 650
      ESF=FLOAT(ERROR)/FLOAT(ERRCNT)
  650 CONTINUE
C
      WRITE(6,700) ERROR,ERRCNT,ESF,CTLNCT
  700 FORMAT('0NUMBER OF INCORRECT PELS =',I10/
     *       '0NUMBER OF BITS IN ERROR TRANSMITTED =',I10/
     *       '0ERROR SENSITIVITY FACTOR =',F12.4/
     *       '0TOTAL NUMBER OF OUTPUT LINES PROCESSED = ',I8)
C
      RETURN
  800 CONTINUE
      STOP 800
      END
C          END OF DCEC UPRINT PROGRAM              LINES PRINTED=    100
```

AD-A092 395 DELTA INFORMATION SYSTEMS INC JENKINTOWN PA F/G 17/2
MEASUREMENT OF COMPRESSION FACTOR AND ERROR SENSITIVITY FACTOR --ETC(U)
AUG 80 S URBAN, R SCHAPHORST DCA100-80-M-0145

UNCLASSIFIED NCS-TIB-80-6 NL

2 OF 2
AD
A092395

END
DATE
FILMED
1 81
DTIC

```
START OF DCEC UPRINT PROGRAM          DSNAME=F3802.MI2B.ASM
         TITLE 'MI2B - BIT STUFFING ROUTINE'                                00001000
MI2B     CSECT                                                              00002000
         SPACE                                                              00003000
*        THIS ROUTINE STORES THE NB LOW-ORDER BITS OF THE WORD              00004000
*        IVAL IN THE ARRAY IBA, STARTING WITH BIT JB.  JB MAY BE            00005000
*        ARBITRARILY LARGE.                                                 00006000
         SPACE                                                              00007000
*        CALLING SEQUENCE --                                                00008000
         SPACE                                                              00009000
*                CALL MI2B(IVAL,IBA,JB,NB)                                  00010000
         SPACE                                                              00011000
         SAVE  (14,9),,*                                                    00012000
         USING MI2B,9                                                       00013000
         LR    9,15                        LOAD BASE REGISTER               00014000
         LA    15,SA                       OUR SAVE AREA ADDRESS            00015000
         ST    13,SA+4                     SAVE CALLERS ADDRESS             00016000
         ST    15,8(,13)                   SAVE OURS IN CALLERS             00017000
         LR    13,15                       R13 NOW POINTS TO OURS           00018000
         LM    1,4,0(1)                    A(IVAL,IBA,JB,NB)                00019000
         L     3,0(,3)                     JB                               00020000
         LTR   5,3                         JB-TH BIT                        00021000
         BNP   ABEND20                     NEGATIVE, STOP 10                00022000
         BCTR  3,0                         JB-1                             00023000
         N     3,=XL4'1F'                  0-31                             00024000
         SPACE                                                              00025000
*        DETERMINE WORD BOUNDARY FOR 64-BIT WORD TO LOAD                    00026000
         SPACE                                                              00027000
         BCTR  5,0                         JB-1                             00028000
         SRL   5,5                         (JB-1)/32                        00029000
         SLL   5,2                         ((JB-1)/32) * 4                  00030000
         LA    2,0(5,2)                    FIRST WORD IN IBA OF PAIR        00031000
         LA    8,32                                                         00032000
         S     8,0(,4)                     32-NB                            00033000
         BM    ABEND20                     NB>32                            00034000
         C     8,=XL4'1F'                  NB>0?                            00035000
         BH    ABEND20                     NO, ERROR                        00036000
         L     4,0(,1)                     IVAL                             00037000
         LM    5,6,=F'0,-1'                SECOND WORD FOR DOUBLE SHIFT     00038000
         SRL   6,0(8)                      NB ONE-BITS                      00039000
         SLL   6,0(8)                       LEFT JUSTIFIED                  00040000
         SLL   4,0(8)                      IVAL LEFT JUSTIFIED              00041000
         LM    0,1,0(2)                    TWO WORDS OF IBA                 00042000
         LM    7,8,=F'0,-1'                MASK AND FLIPPER                 00043000
         SRDL  6,0(3)                      SHIFT MASK BITS                  00044000
         SRDL  4,0(3)                       AND IVAL BITS                   00045000
         XR    6,8                         FLIP MASK BITS                   00046000
         XR    7,8                         0=1, 1=0                         00047000
         NR    6,0                         GET NON-CHANGED BITS             00048000
         NR    7,1                          FROM IBA                        00049000
         OR    6,4                           AND COMBINE WITH               00050000
         OR    7,5                            IVAL                          00051000
         STM   6,7,0(2)                    SAVE CHANGED WORDS BACK IN IBA   00052000
         L     13,SA+4                                                      00053000
         RETURN (1,9)                                                       00054000
         SPACE                                                              00055000
*        INVALID COUNT FOR NB (NEGATIVE, MORE THAN 32) OR JB (NEGATIVE)     00056000
         SPACE                                                              00057000
ABEND20  ABEND 20,DUMP                                                      00058000
SA       DC    15F'0'                                                       00059000
         END                                                                00060000
0          END OF DCEC UPRINT PROGRAM            LINES PRINTED=    50
```

```
START OF DCEC UPRINT PROGRAM          DSNAME=D3B02.I4B.ASM
          TITLE 'I4B - BIT EXTRACTION ROUTINE'                          00001000
I4B       CSECT                                                         00002000
          SPACE                                                         00003000
*         THIS ROUTINE ACTS AS AN INTEGER FUNCTION, RETURNING (IN REG0) 00004000
*         A BIT STRING, ZERO-FILLED, OF LENGTH NB, STARTING AT THE JB-TH 00005000
*         BIT OF ARRAY IBA.  JB MAY BE ARBITRARILY LARGE.               00006000
          SPACE                                                         00007000
*         CALLING SEQUENCE --                                           00008000
          SPACE                                                         00009000
*                NBITS = I4B(IBA,JB,NB)                                 00010000
          SPACE                                                         00011000
          SAVE  (14,5),,*                                               00012000
          USING I4B,5                                                   00013000
          LR    5,15                    LOAD BASE REGISTER              00014000
          LA    15,SA                   OUR SAVE AREA ADDRESS           00015000
          ST    13,SA+4                 SAVE CALLERS ADDRESS            00016000
          ST    15,8(,13)               SAVE OURS IN CALLERS            00017000
          LR    13,15                   R13 NOW POINTS TO OURS          00018000
          LM    1,3,0(1)                A(IBA,JB,NB)                    00019000
          L     2,0(,2)                 JB - LOCATION OF BITS           00020000
          LTR   4,2                     JB-TH BIT                       00021000
          BNP   ABEND10                 NEGATIVE, ABEND 10              00022000
          BCTR  2,0                     JB-1                            00023000
          N     2,=XL4'1F'              0-31                            00024000
          SPACE                                                         00025000
*         DETERMINE WORD BOUNDARY FOR 64-BIT WORD TO LOAD               00026000
          SPACE                                                         00027000
          BCTR  4,0                     JB-1                            00028000
          SRL   4,5                     (JB-1)/32                       00029000
          SLL   4,2                     ((JB-1)/32) * 4                 00030000
          L     0,0(4,1)                FIRST WORD                      00031000
          L     1,4(4,1)                SECOND WORD                     00032000
          SLDL  0,0(2)                  SHIFT OUT HIGH-ORDER GARBAGE    00033000
          LA    2,32                    COMPUTE RESIDUAL NUMBER TO SHIFT 00034000
          S     2,0(,3)                 32-NB TO SHIFT                  00035000
          BM    ABEND10                 NB>32                           00036000
          C     2,=XL4'1F'              NB>0?                           00037000
          BH    ABEND10                 NO, ERROR.                      00038000
          SRL   0,0(2)                  RIGHT JUSTIFY                   00039000
          L     13,SA+4                                                 00040000
          RETURN (1,5)                  RETURN WITH VALUE IN REG 0      00041000
          SPACE                                                         00042000
*         INVALID COUNT FOR NB (NEGATIVE, MORE THAN 32) OR JB (NEGATIVE) 00043000
          SPACE                                                         00044000
ABEND10   ABEND 10,DUMP                                                 00045000
SA        DC    18F'0'                                                  00046000
          END                                                           00047000
0           END OF DCEC UPRINT PROGRAM          LINES PRINTED=    47
```

```
START OF DCEC UPPINT PROGRAM          DSNAME=N0026.STATS.FORT
      SUBROUTINE STATS(LENGTH,INLNCT,DIAG)
      IMPLICIT INTEGER(A-Z)
C
      INTEGER MTT(5),ITT(3,5),LENGTH(INLNCT)
      REAL STT(2,5),SUM,SUMSQ
      LOGICAL DIAG
C********************** FILE DEFINITIONS ******************
C
      COMMON/FILES/TERM,_PFIL,PFLFIL,CTFIL,FPFIL
C
      DATA MTT/0,24,48,96,192/
C
C*********************BEGIN PROGRAM*********************************
C
      CALL QKSORT(LENGTH,1,1,1,INLNCT)
      DO 300 I=1,5
      ITT(1,I)=10000
      ITT(2,I)=0
      SUM=0.
      SUMSQ=0.
      DO 100 J=1,INLNCT
C
C   FIND FILLED LINE LENGTH
C
      LEN=MAX0(LENGTH(J),MTT(I))
      IF(DIAG) WRITE(TERM,50) LEN
   50 FORMAT(I8)
C
C   FIND MINIMUM LINE LENGTH
C
      ITT(1,I)=MIN0(LEN,ITT(1,I))
C
C   FIND MAXIMUM LINE LENGTH
C
      ITT(2,I)=MAX0(LEN,ITT(2,I))
C
C   FIND SUM OF LENGTHS
C
      SUM=SUM+FLOAT(LEN)
      SUMSQ=SUMSQ+(FLOAT(LEN))**2
  100 CONTINUE
C
C   FIND MEDIAN LINE LENGTH
C
      MEDI=INLNCT/2
      ITT(3,I)=MAX0(LENGTH(MEDI),MTT(I))
C
C
C   FIND SAMPLE MEAN AND STANDARD DEVIATION
C
      STT(1,I)=SUM/FLOAT(INLNCT)
      STT(2,I)=SQRT((SUMSQ-(SUM**2)/FLOAT(INLNCT))/FLOAT(INLNCT-1))
  300 CONTINUE
C
      WRITE(6,400)(ITT(1,I),I=1,5)
  400 FORMAT(
     *'0                          MINIMUM TRANSMISSION TIME (4800 BPS)'//
     *'  CODED LINE'/
     *'    LENGTH                 0 MS     5 MS    10 MS    20 MS    40 MS'/
     *'  STATISTICS:'//
     *'    MINIMUM',10X,5(I8)//)
      WRITE(6,410)(ITT(2,I),I=1,5)
  410 FORMAT(
     *'    MAXIMUM',10X,5(I8)//)
      WRITE(6,415)(ITT(3,I),I=1,5)
  415 FORMAT(
     *'    MEDIAN',10X,5(I8)//)
      WRITE(6,420)(STT(1,I),I=1,5)
  420 FORMAT(
     *'    SAMPLE MEAN',8X,5(F6.2)//)
      WRITE(6,430)(STT(2,I),I=1,5)
  430 FORMAT(
     *'    STANDARD DEVIATION',2X,5(F3.2))
C
      RETURN
      END
0          END OF DCEC UPPINT PROGRAM          LINES PRINTED=    75
```