A Study

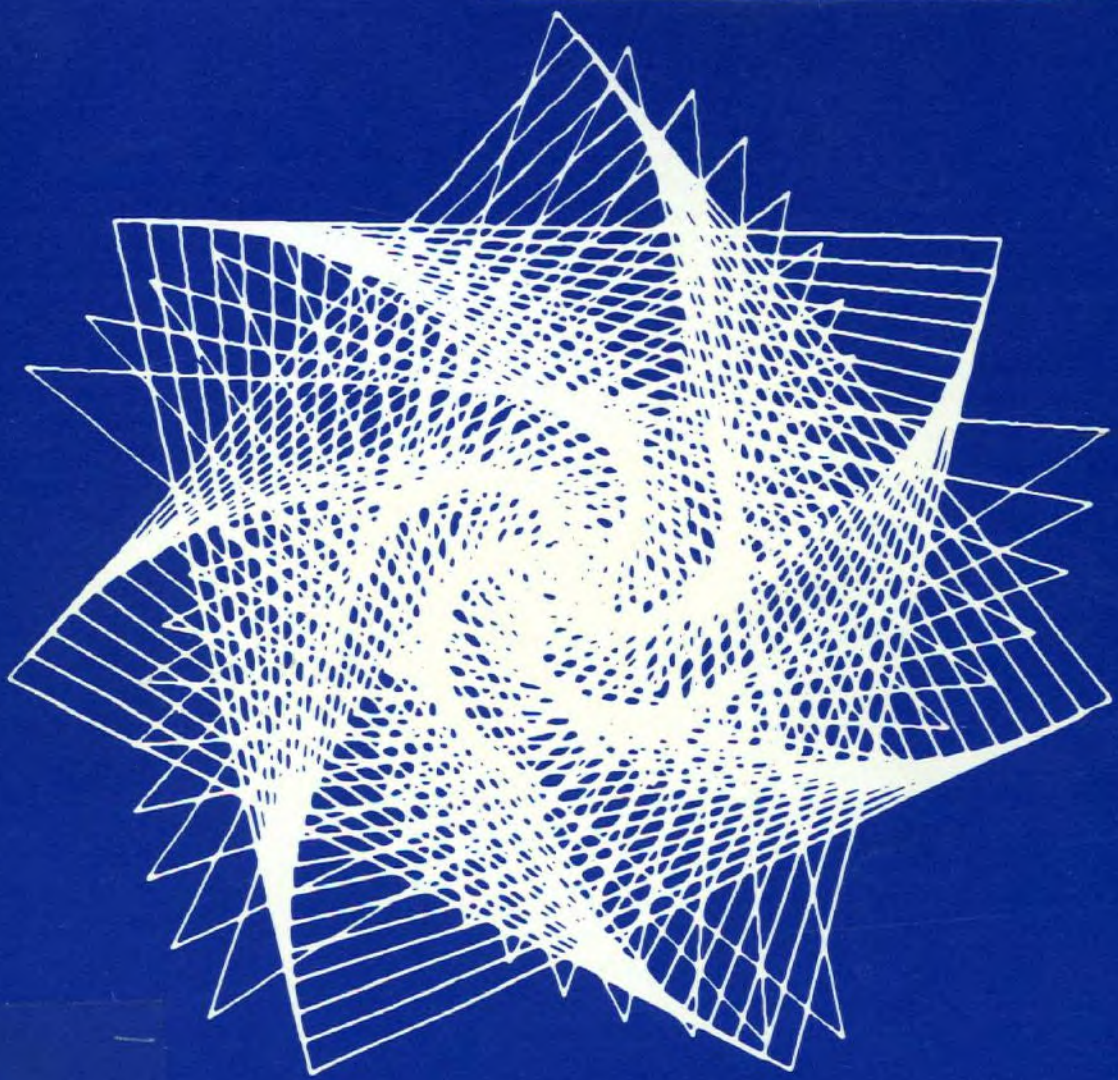
For The

Department of Communications (DOC)

New Teletext Services (NTS): Report #2

Tecnnical Requirements:

Definition and Analysis Report



orpak corporation

A Study

For The

Department of Communications (DOC)

New Teletext Services (NTS): Report #2

Technical Requirements:

Definition and Analysis Report

NORPAK Corporation

A Study

For The

Department of Communications (DOC)

New Teletext Services (NTS): Report #2

Technical Requirements:

Definition and Analysis Report

Submitted In Partial Fulfillment Of Contract

DSS File # 1ER.36001-4-1979

Authored by:

Robert J. Fitzgerald, Manager, Hardware Systems
Dr. John A. Norton, Manager, Engineering and Systems Sales

Document No.	DOC35046
Date of Issue	850325
Revision	FINAL REPORT

NORPAK Corporation

10 Hearst Way
Kanata, Ontario
Canada K2L 2P4

T A B L E O F C O N T E N T S

L I S T O F T A B L E S

1. EXECUTIVE SUMMARY

This document presents design considerations and alternatives to meet the requirements derived from an analysis of new teletext applications that could not be met under the current definition of NABTS.

The principal technical requirements derived for new teletext services are:

* error protection mechanisms to ensure that the required level of data service or product integrity is available to the end-user; and

* provider/operator control of access to services,

and to a lesser extent,

* the provision of video "film strip" capability.

These must be fulfilled optimally in consideration of the following performance-related requirements:

* optimal delivery channel usage while meeting other system function and performance needs;

* optimal terminal design reflecting the compromise between function, performance and cost (capital and operating);

* new system feature upward compatibility to ensure graceful introduction of new services without the disruption (in its full generality) of those already existing; and

* their impact and possible recommended extensions to NABTS.

While not strictly related to consideration of technical requirements of new teletext services that may necessitate extensions to NABTS, the study includes a brief look at the implications of providing effective text capability for videotex and teletext services based on NAPLPS/NABTS in select Asiatic countries.

The principal results and recommendations to result for error protection include that:

* for 7-bit data services, the Carleton code system is quite acceptable; and

* that for 8-bit data services one of Fire, Reed-Solomon or NHK code (each with their own special characteristics) will suffice, although the actual selection will likely be determined by the hardware implementation available.

The principal results and recommendations for service access control are that:

* the current NABTS addressing definition may be satisfactorily applied for group and specific addressing;

* that DES can be used for highly security sensitive 8-bit data applications in which there is sufficient error protection;

* a simple exclusive-OR scheme together with appropriate key management may be satisfactorily used for an inexpensive implementation; and

* key management be implemented using in-channel techniques, with the implication that an extension to the current definition of NABTS will be required.

For video "film strips", the principal results are that:

* only composite video encoding of static or sequences of frames is viable;

* a combination of cyclic and request-driven broadcast of video frames is optimal;

* that while animated sequences must be re-requested for re-viewing, static images may be retained or moving images "frozen" for continuous view.

The primary recommendation for adapting NAPLPS to Asiatic written text requirements are that:

* all necessary additional font symbols be stored locally in ROM; and

* for Arabic and Japanese phonetic alphabets, access to the additional symbols be via one-byte codes accessed through an additional G-set, and that for access to the Chinese, Japanese and Korean Hansi set, that access be provided through an ISO-compatible extended two-byte G-set.

2. INTRODUCTION

2.1. Background

NORPAK Corporation has been contracted by the Department of Communications to perform a study in which new teletext-based services are examined, and to make specific recommendations for any extensions that may be required of NABTS.

The only recommendations for incorporation into NABTS will be those which can be shown to be manifestly essential to either the functionality, performance, cost or some combination of these to implement a viable service.

The complete study and recommendations are be contained in four documents which address the following topics (grouped according to the contents of each document):

* analysis of applications and derivation of their technical requirements consolidated for all of the applications examined;

* quantification of the major technical requirement groups and consideration of alternative design approaches with particular emphasis on their implications with respect to the viability of the services (applications) and possible extensions to the basic NABTS specification;

* specific recommendations for extension to the current NABTS definition; and

* an examination of the possible design approach alternatives to the implementation of the new services (using the recommended extensions to NABTS) with special emphasis on the teletext sourcing and delivery plant.

2.2. Technical Requirements Summary

This document represents the second component of an overall study to address the analysis of several new teletext-based services (applications). It is possible that extensions to the current NABTS definition will be required in support of these applications.

The two principal sets of requirements are for:

* error protection mechanisms to ensure that the required level of data service or product integrity is available to the end-user;

* service provider/operator control of the user terminal to access services and to use data and software products;

These must be fulfilled optimally in consideration of the following performance related requirements:

* optimal delivery channel usage while meeting other system function and performance needs;

* optimal terminal design reflecting the compromise between function, performance and cost (capitol and operating); and

* new system feature upward compatibility to ensure graceful introduction of new services without the disruption (in its full generality) of those already existing.

While not strictly related to consideration of technical requirements of new teletext services that may necessitate extensions to NABTS, we have included as part of this study a brief look at:

* design approaches for providing video "film strip" capability;

* implications of providing effective text capability for videotex and teletext services based on NAPLPS/NABTS in select Asiatic countries (which affects the Presentation Layer of FSS in NABTS, which by definition is NAPLPS to SRM level); and

* return communication for transaction purposes, which is a requirement that is largely independent of NABTS.

In addition, there is another application requirement which will not be addressed here whatsoever:

* sound and voice synthesis, and synchronization with decoded presentation layer text and graphics.

2.3. Approach

The approach to be used for this study report will be:

* to consider separately each technical requirement category, taking the qualitative description (given in Report #1 of this study), and to add quantification where possible;

* to propose for each of these, the major technical design alternatives (or groups of alternatives);

* to perform a brief analysis of the function, performance and cost viability (pro's and con's) for each service (application) which can be identified to depend on the fultillment of the requirement, in the context of that service; and

* to make a recommendation(s) for the design approach to the requirement in consideration of the services as group.

2.4. Criteria For Evaluation

2.4.1. Optimal Channel Use

Given sufficient communication bandwidth, almost any level of function and performance are achievable. Unfortunately, this is not the reality. A global requirement relates to the optimal use of channel capacity. Many of the communication-related requirements place a burden on system response performance and receiver-adapter cost traded off against channel capacity.

2.4.2. Cost Acceptable User Terminal

Achievement of system functionality and performance is heavily embodied in the capabilities of the user terminal system. Unfortunately many technically good solutions make the user terminal system or receiver-adapter prohibitively expensive and untenable.

2.4.3. NABTS New Feature Compatibility

As these new service are introduced, they will have to coexist with the teletext plant and receiver-adapter population that may already be in place, and the specifications and standards by which they operate.

In North America, any new application services which require an extension to NABTS should be cleanly and exactly upward compatible with the current definition. To do otherwise will bring chaos and detriment to this fledging industry.

For Asiatic markets, both NABTS teletext decoders and signal plant should be relatively close derivatives of the North American product such that the adaptation is as economical and timely as possible.

3. ERROR PROTECTION

3.1. Introduction

The "Applications and Requirements Report" (#1) identified the need for enhanced error correction techniques for use in several of the new services discussed in that report. We will first summarize the logic underlying this requirement.

The information in several types of services which could be delivered through NABTS channels is error-sensitive. This includes financial information which, if incorrect, could cause actual material loss to decision-makers using that information. Another example is telesoftware which is totally intolerant of errors in the code.

Traditional teletext transmission systems, such as the NABTS as it is currently defined, have achieved an interesting balance between error correction, performance and cost. The NABTS, for example, makes use of the relatively limited Forward Error Correction (FEC) provided by the Product code, and the somewhat more powerful detection (complemented with re-cycling) capabilities provided by the Product code and (8,4) extended Hamming code.

These error detection and correction capabilities are harnessed as follows:

* information is divided into records, each of which consists of a number of packets;

* each packet is protected by the Hamming code on prefix and header bytes, and the Product code on information bytes;

* the NABTS receiver, upon receiving a record, corrects as many errors in each packet as it can, through use of the Hamming and Product codes;

* any packets which contain errors which these codes can detect but cannot correct are identified as containing decoding failures;

* the NABTS receiver waits for subsequent retransmission of the same record, and extracts the required packets from a retransmission in order to replace the packets containing the decoding failures; and

* this procedure is repeated until a correct copy of the record is fully reconstructed.

This process provides relatively good performance under many circumstances, and does not require any special hardware for error-correction in the receiving terminal. For this reason we

can say that a reasonable cost/performance balance has been reached for traditional teletext services carried through the NABTS channel.

For business services, however, the revenue which the service provider derives from a service is directly proportional to the amount of content carried. Therefore, it cannot be assumed that there will necessarily be retransmissions of each record. The use of one or two retransmissions of eacn record corresponds to a code rate of less than .5 or .33, respectively. This is not a very effective error protection mechanism for the following reasons:

* even if a code rate of .5 or .33 were acceptable, much more powerful codes than this can be designed, as we shall see in a subsequent section; and

* the performance of this scheme is limited by the detection capability of the Product code which can detect all patterns of two and some patterns of multiple errors occurring in a packet.

Since there are some patterns of multiple errors which will escape detection by the Product code, these will not be corrected by subsequent retransmissions of the record unless an additional majority voting mechanism is built into the error correction procedure.

In summary, enhanced error protection is required for two reasons:

* first, because the new teletext services contain data which is very sensitive to errors; and

* secondly, because the retransmission scheme commonly used in existing teletext services is inappropriate for services where maximizing throughput is of paramount importance, and where the cost of providing more powerful error correction in the receiving terminal is a less important factor than it would be in a consumer terminal.

3.2. Error Environment

In this section we present a summary of the principal mechanisms which generate errors in a teletext channel, and the characteristics of the errors which are caused. Before proceeding to a statement of the requirements which must be met by the new error-correction facility, we will examine the nature of these errors and some of the strategies which can be used for avoiding them.

In the following discussion, some assumptions are made. In particular, the "typical" record is assumed to contain approximately 1,ØØØ bytes of information. A message may consist

of a series of such records; for example, a large telesoftware program may contain 100 or more records. This subdivision of messages into relatively fixed size records is introduced here simply for ease of analysis.

3.2.1. Independent Errors

The first type of errors which occur in teletext channels are those known as independent errors, and are typically due to additive Gaussian noise. This class of errors is characteristic of many satellite channels and some "good" VHF/UHF broadcast channels.

Independent errors are handled quite well by the Product code which is currently defined in the NABTS. For example, the probability that one or more packets in a 1,000 byte record encoded using the Product code will exhibit decoding failure is 0.9x10E-2 at a raw bit error rate of 1.0x10E-4. This moderate bit error rate can typically be achieved with a relatively low video signal-to-Gaussian-noise ratio, such as 27 dB or less. For this reason independent errors are not the primary motivation for introducing enhanced error-correction capabilities. However, whatever error-correction mechanisms are included for other purposes will clearly have the effect of reducing the threshold for adequate operation in the presence of Gaussian noise.

3.2.2. Burst Errors

The second important class of errors is burst errors caused by impulse noise. Impulse noise is also often known as man-made noise, and can be produced by a number of sources such as automobile ignition, fluorescent lights, electric arcing in machinery and switching noise. Its effect is important in the VHF band, and it therefore affects broadcast VHF and CATV channels. Its effect on UHF channels is considerably less important.

The "burst length" is defined, for our purposes, to be the number of bits between the first error and the last error (inclusively) in a single packet. A burst of length N, therefore does not necessarily contain N bit errors, but rather contains some variable number of errors which are spread over N bits.

Very little information is currently available on the frequency of occurrence of bursts as a function of the burst length. However, Yamada (reference 7) provides some statistics based on measurements made in Japan by NHK. These indicate that approximately 91% of bursts due to impulse noise have a length less than 34 bits. Clearly then, a code which is capable of correcting 34 bit bursts or 34 independent errors would provide a significant improvement in performance. Unfortunately, however, this reference does not provide statistics on a finer scale for burst lengths between 1 and 34 bits. It is therefore not

possible to deduce the performance of codes which can correct less than 34 independent errors. In fact, we can assume that a code which is capable of correcting considerably fewer than 34 random errors might provide performance approximately equivalent to one which can correct a 34-bit burst, since most 34 bit bursts will contain less than 34 actual bit errors.

3.2.3. Signal Distortion Induced Errors

The third important class of errors is due to signal distortions such as multipath and group delay. This class of errors can be viewed either as bursty or independent. Since multipath produces a cyclic group delay characteristic, these forms of distortion are studied together here. In general, they are manifested as either an amplitude or a delay inequality (or both) across the frequency band, with the effect that certain bits are more likely to contain errors than others.

In many cases the characteristic is similar to that of a low-pass filter. In these circumstances, any bit which is preceded by one or more bits of the opposite polarity has a higher probability of error than a bit which is preceded by bits of the same polarity. Given the assumption of pseudorandom data, one quarter of the bits in a packet are subject to intersymbol interference from two preceding bits of opposite polarity, and therefore have a higher probability of error. Intersymbol interference extending over three bits would affect 1/8th of the bits in the packet, and so on.

Because the bit errors are spread randomly throughout the packet they can be viewed as independent. Clearly they are not, in fact, genuinely independent, since the mechanism in itself is inherently pattern-sensitive. However, treating them as independent errors may well prove more fruitful than viewing them as burst errors since, in many cases, the length of the burst may be comparable to that of the packet.

3.2.4. Packet Erasures

Multipath and impulse noise can also cause anomalous burst errors if their effect is strong enough to cause the receiver to lose synchronization. When this occurs each of the bits in the packet may be mis-sampled and those which already have a higher probability of error will be even more prone to error. This mechanism may, in fact, cause total packet erasures.

A packet erasure is defined as the complete loss of a packet, such that a receiver can detect its absence but cannot compensate for it through any packet-level procedures. Packet erasures can occur for a number of reasons. Multipath, group delay or impulse noise can cause the number of bit errors in a packet to exceed the error-correction capabilities of the code, either through loss of synchronization or a very long burst of errors; or

framing and addressing information at the beginning of the packet can be severely corrupted, such that the receiver never recognizes the presence of the packet and therefore does not receive it. For each of these two cases there are two different strategies which can be used.

If the data portion of the packet is unusable, the receiver can take one of the two following courses of action: It can make no further attempt to correct the data, and either use it with errors or reject the record to which the packet belongs; or, if the service provider has included a secondary code which operates over groups of packets, it may be possible to correct the erasure.

One class of such codes, known as Product codes, can be described as follows:

* visualize a group of packets as a rectangular matrix in which each row represents one packet;

* parity bits are added to the end of each packet forming horizontal code words; and

* the bottom row, that is the last packet of the group, consists of parity bits which operate on vertical code words which contain symbols selected from each of the packets in the group.

An example of such a code is the Bundle code developed by Carleton University, which consists of horizontal and vertical Reed-Solomon codes.

If the framing code or addressing information in the prefix of a packet is destroyed by errors, the situation is somewhat different. If a very powerful code is applied to each packet, one could include the prefix in each horizontal code word. This would allow even severe errors in the prefix to be corrected and therefore the probability of a packet being erased due to the corrupted prefix would be minimized. However, in typical implementations, the prefix is used as a filtering mechanism to reduce the volume of data which the receiver has to process in real time. For this reason, the prefix must normally be processed in real time in hardware, and a decision is made on whether to accept the packet well before the entire packet has been received.

In a VBI-only teletext system this would not be a problem since the receiver would have a full field-time in which to process one VBI's data, and real-time processing of the prefix would not be required. However, in full-channel transmissions packet address matching must be performed in hardware. One strategy would be to use a packet matching circuit which is more tolerant of bit errors than the commonly used incomplete decoder of the extended (8,4) Hamming code. If the circuit matched on bytes containing up to two bit errors instead of the one bit error tolerated by

the complete decoder, the probability of rejecting packets due to errors in the prefix would be significantly reduced. This would also mean that some additional spurious packets could be received due to false Hamming correction. However, the packet-level FEC code should be able to reject these. The latter operation of course does not have to take place in real time, since it occurs once a record has been captured by the receiver.

An alternate means of reconstructing packets which have been erased due to decoding failure in the prefix, is to use a nested code or Product code as described previously.

There are also certain burst avoidance strategies which can be adopted to supplement the power of any of the above error correction codes. If a code is in use which allows complete reconstruction of erased packets, then packets from various services should, if possible, be multiplexed such that there is no more than one packet per TV field per service. In this way, a burst of errors due to impulse noise (which will normally affect only one TV field) will only corrupt one packet per service, which can be corrected by the erasure capability of the code.

Another approach which could be used to good effect on a channel known to have bursty error characteristics is to break the one-to-one correspondence between codewords and packets. This is achieved by encoding a block of data into several codewords, and then dispersing the symbols in each codeword among several packets. In this way a burst occurring in one packet will cause a small number of errors in each of several codewords, rather than a large number of errors in any one codeword. This approach, while theoretically interesting, has two significant disadvantages:

* the service provider normally does not have any pre-knowledge of the characteristics of any given channel at a large number of receiving sites, and therefore has no way of knowing whether this strategy is appropriate; and

* this technique will, in general, reduce bandwidth efficiency since extra padding codewords will normally need to be added to the end of a record in order to fill an integral number of packets.

3.3. Requirements of An Enhanced Error Correction System

Enhanced error correction is required for services whose fundamental information symbol contains either 7 or 8 bits. The 7-bit codes are for use in improving the reception of traditional NABTS teletext services and therefore must be compatible with the error-correction code currently defined in NABTS. The 8-bit codes are for protection of new classes of data which either inherently consist of 8-bit data symbols, or have been transformed into 8-bit data symbols by an encryption process. The 8-bit codes are not required to be compatible with existing

or proposed 7-bit codes.

The new codes shall be capable of correcting "I" independent errors or a burst of duration "J" or some combination thereof. Here "I" and "J" remain to be defined for each of the classes of service, for the following reasons:

* there are almost no statistics on error distributions in North American television channels; and

* potential service providers are unable to indicate a firm limit on the net channel error rate which they can tolerate.

However, there are indications (reference 7) that "I" should be at least 8.

The error control mechanisms must either ensure that packet erasures virtually never occur or that they can be corrected when they do occur.

3.4. Codes For 7-Bit Data Symbols

Considerable work has been done by Mortimer et al at Carleton University on the development of a family of compatible codes for protection of the 7-bit data symbols currently used in the NABTS standard. This work, which is documented in a series of unpublished reports prepared for the Department of Communications, has produced some very powerful codes which are described briefly here.

There are three levels of coding in the proposed system. The lowest level is the Product code defined in NABTS. This is a code formed by adding a single longitudinal parity byte to each 27-byte packet. Each of the 27 information bytes in a packet contains a single odd parity bit. The Product code is capable of correcting any single bit error in a packet. It can also detect all patterns of two bit errors and many patterns of multiple bit errors which occur in a packet.

The second level of coding is known as the "C" code. It is formed by reducing the packet size to 26 bytes and inserting 8 additional check-bits in the 27th byte. This additional check-byte, in combination with the longitudinal parity in the 28th byte, forms a Reed-Solomon code over the packet. This code can correct any two bit errors which occur randomly throughout the packet, or any number of errors which occur within one single byte, or any double-byte erasure.

The third level of protection is known as the Bundle code. It is obtained by periodically adding check-packets which form a Reed-Solomon code (identical to the "C" code) over the vertical code words consisting of bytes belonging to the preceding data-packets. This code can correct up to 14 byte errors provided that they occur in separate vertical code words. It can also

correct up to 28 byte erasures provided that they occur in 14 separate vertical code words. It is therefore capable of replacing one complete packet which has been lost.

One of the most important features of this group of codes is their mutual compatibility. All receivers, regardless of the level of error correction which they implement, are capable of receiving data which is coded with any of these three levels. For example, a receiver which is capable of Bundle code correction is inherently capable of "C" and Product-code correction and can therefore receive and correctly decode services which make use of only the Product or "C" code. Similarly, a receiver which only implements the Product code correction can receive a service which is encoded using the "C" or Bundle codes and simply discard the additional check-bytes which it is unable to process. This compatibility provides great flexibility to service providers who may choose to use the highest level of error correction on certain valuable data, and use a lower level of correction on other less important data because of its lower overhead. Similarly, receiver manufacturers have the flexibility of tailoring the error correction capabilities of their receivers to either the expected quality of signal or the perceived importance of the service.

One other error correction scheme for 7-bit data was presented by G. Woodsum of Zenith Corporation at ICCE '84. This is mentioned here for completeness only, since this scheme was not designed as a backward-compatible extension of NABTS. The principle of operation is as follows: a group of packets is broken up into a series of N by N matrices and a parity byte is added to each row and column of each matrix. This is a "Product code of Product codes", and bears some resemblance to the Bundle code. The principal differences are as follows:

* the horizontal code word is a fraction of a packet, rather than a full packet as in the Bundle code; and

* the vertical code word is protected by a single parity byte rather than the two bytes used by the Bundle code.

Very good performance is claimed for this code and it is internally upward and downward compatible in the sense that every record contains protocol information which indicates the size of the matrix used in calculating the code. Since the size of the matrix determines both the performance and the overhead of the code, the service provider could select a matrix size which provides the required performance for the service which he wishes to deliver. However, this error-correction scheme is not compatible with the existing NABTS standard in that it would not be possible for existing FSS decoders to receive services encoded using this method. For this reason, there is no further discussion of this coding scheme in this report.

The authors of this report are unaware of any alternative coding schemes which have been proposed for enhancing the protection of the NABTS 7-bit data bytes. For this reason the only system which can be recommended is the three-tiered Product/C/Bundle code family described previously.

There are two implementations of the Bundle code which have been studied. The first, known as the single Bundle code, is formed by protecting N data packets by 1 check packet. One value of N which has been proposed is 8. The double Bundle code consists of protecting 2N packets by 2 check packets. Studies indicate that the double Bundle code provides better performance than the single bundle code. However, before actually recommending its adoption, let us quickly examine the relative efficiencies of these two implementations.

The code rates for the three levels of code, including two versions of the Bundle code can be compared. We exclude the single parity bit in each byte from the calculation. In addition, for the Bundle codes, assume that the record size is approximately 1,000 bytes or 38 packets. On average, a record will contain an integral number of full Bundles plus one half-full Bundle. We therefore select the number of full Bundles N, such that (2N+1)/2 Bundles contain 38 packets (as nearly as possible). The following code rates result:

* the Product code has a rate of 27/28 =96.4%;

* the C code has a rate of 26/28 = 92.8%;

* for the single Bundle code with 8 data packets per Bundle, 4.5 Bundles will contain 36 data packets. Since these will require 5 check-packets, a total of 41 packets are included in the record. The code rate is therefore (36 x 26)/(41 x 28) = 81.5%; and

* the double Bundle code with 16 data packets per bundle will require 2.5 bundles for a similar record. The record will then contain 40 data packets and 6 check packets for a total of 46. The code rate is therefore (40x26)/(46x28) = 80.7%.

Thus for a typical record size of 1,000 bytes there is very little difference in efficiency between the single and double Bundle codes, and the double Bundle code is preferred because of its improved error correction capabilities. For very short records (less than one full Bundle), the efficiency of the double Bundle code would be considerably less than that of the single Bundle code. However, such small records are not typical of any known or anticipated teletext services. It could also be argued that, for such records, the Bundle code would be overkill anyway.

3.5. Codes For 8-Bit Data

3.5.1. Terminology

The reader of this section is assumed to be familiar with some of the basic concepts of error correcting codes and the underlying mathematics. As such, we will not include here a review of the algebra of finite fields. The interested reader is referred to the excellent treatment of this subject in reference 4. We will, nevertheless, define here some of the terminology which will be used in the following discussion.

* A block code of size "M" over an alphabet with "q" symbols is a set of "M"q -ary sequences of length "n" called code words.

* A code whose codewords consist of "n" symbols, "k" of which are information symbols, is known as an "(n,k) code".

* The (n-k) remaining symbols of each code word are known as parity symbols.

* The rate of a code is defined by R = k/n.

* The Hamming distance "d" between two sequences is the number of places in which they differ.

* The minimum distance of a code is the Hamming distance of the pair of codewords with smallest Hamming distance. A (n,k) block code with minimum distance d* is also known as an (n,k,d*) code.

* The number of errors "t" which can be corrected by a code is always less than or equal to (d*-1)/2.

* A finite field with "q" elements is known as a Galois field and is denoted by the label GF(q).

3.5.2. Classes of Codes

There are four general classes of codes which have received considerable study. The best known class of codes are the linear block codes, and these are the only ones which will be discussed in any depth in this report. The other three classes of codes all contain some interesting and powerful codes but, for various reasons which will be summarized here, do not merit further study in a brief summary of this nature.

There are some good non-linear codes such as the Preparata. However, there is no equivalent for non-linear codes of the vast array of powerful mathematical tools used in finding and studying linear codes. Good non-linear codes, therefore, although they may exist, are much more difficult to find. Given the variety and power of the linear codes, there is little reason to believe

that a search for an effective non-linear code for use in teletext would be a fruitful exercise.

Convolutional codes, which are a class of tree codes, are characterized by the fact that each information frame is encoded along with some number of previous information frames into a single codeword frame. This distinguishes them from most other codes where a single information frame is coded into a single codeword frame. For this reason, they are more suitable for a continuous data stream than for packet-structured data such as teletext. Again, because of the power of many of the linear block codes, it does not appear useful to adapt the convolutional codes (which, in general, do not provide any performance improvement over linear block codes) to the packet environment of teletext.

There is a class of codes based on spectral techniques, that is, on Fourier transforms over the Galois field. These receive no further specific discussion in this report for the reason that in general, while the spectral approach provides good insight into codes such as the extended Reed-Solomon, there exist equivalent time domain implementations of these codes. For this reason, and although there are some codes such as the Goppa which have been studied almost exclusively through spectral techniques, this report will not focus on this representation of codes.

3.5.3. Linear Block Codes

A linear code is a set of codewords of length n over GF(q), such that the sum of two codewords is a codeword and the product of any codeword by a field element is a codeword. The minimum distance d* of any linear (n,k) code is less than or equal to 1+n-k. This is known as the Singleton bound. Any code whose minimum distance satisfies d* = 1+n-k is called a maximum distance code.

One example of a linear code is the familiar Hamming single-error correcting code. The most familiar implementation of the Hamming code is over GF(2) where the information symbol is a bit. However, the Hamming code can also be defined over larger Galois fields i.e. for non-binary data.

Another familiar code is the Golay which is a triple error correcting code over GF(2). This code is also an example of a cyclic code, which is the subject of the next section.

While codes such as the Hamming and Golay are interesting and provide valuable properties for certain types of applications, the long packet length used in teletext will require codes having greater error correction capability.

3.5.4. Cyclic Codes

Cyclic codes are a sub-class of the class of linear codes. They have the property that the word obtained by cyclically shifting the components of any code word by one place is also a code word. This structural requirement does not inherently assure the power of these codes, however, it does simplify their analysis. For this reason the search for good error control codes has been very successful within the class of cyclic codes. One example is the Golay code mentioned previously. There are two other examples which are of particular interest here.

3.5.5. Fire Code

Fire codes are cyclic burst correcting codes and are in fact, the best known burst correcting codes of high rates. The following table gives some examples.

(n,k)	t (max burst length)
(35,27)	3
(105,94)	4
(279,265)	5

One interesting property of cyclic codes is that longer codes can be constructed from any given cyclic code by the technique of interleaving. A (jn,jk) code can be constructed from a (n,k) code by taking j code words from the original code and merging them by alternating the symbols. If the original code is capable of correcting bursts of length t, the resulting interleaved code can correct bursts of length jt. For example, interleaving 8 copies of the (35,27) Fire code yields a (280,216) Fire code which can correct bursts of length 24.

This gives us a clear indication of the power and flexibility of the Fire codes. As shown the (280,216) interleaved Fire code of rate 0.77 can correct bursts of length 24. Alternatively, the (279,265) Fire code having approximately the same value of n and rate .95 can correct bursts of length 5. Decoders for the Fire code are relatively easy to construct in hardware and one typical application for these codes has been in large disk drives.

3.5.6. Quadratic Residue Codes

Another interesting sub-class of the cyclic codes is known as the Quadratic Residue codes. These are mentioned here because they provide very high performance. The following table lists some binary Quadratic Residue codes all of which have rate of approximately 0.5. All of the codes shown in the table are the best known codes for these values of n and k.

(n,k)	d*
(23,12)	7
(41,21)	9
(47,24)	11
(79,40)	15
(89,45)	17
(103,52)	19

The first entry in this table is the (23,12,7) triple error correcting Golay code discussed previously. It is not known whether there are good Quadratic Residue codes with a block length comparable to an NABTS packet. It is, of course, possible to combine several short Quadratic Residue code words in one packet. For example, 7(41,21) code words could correct up to 7x4 bit errors. However, given the bursty nature of some teletext channels, it is unclear whether this would provide an efficient protection mechanism. For this reason, and also because the codes have low rate and are difficult to decode, their applicability to teletext is doubtful.

3.5.7. BCH Codes

The BCH codes form a large class of multiple error correcting codes and are probably the single most heavily studied class of error correction codes. They provide relatively good performance and a number of decoding techniques have been developed for them. We will not include a general discussion of BCH codes in this section, but rather will zero in on one particularly interesting sub-class known as the Reed-Solomon codes.

The Reed-Solomon codes are BCH codes in which the block length divides the multiplicative order of the symbol alphabet. A Reed-Solomon code over GF(q) has n = q-1. The Reed-Solomon codes are optimum in the sense of the Singleton bound. That is, they are maximum distance codes having a minimum distance of n-k+1. This tells us that for fixed (n,k), no code can have a larger minimum distance than a Reed-Solomon code. However, we note that the values of n are restricted for Reed-Solomon codes. Thus a Reed-Solomon code which is shortened so that it matches a particular packet size will not necessarily be optimal.

Let us examine two particular Reed-Solomon codes which could be applicable to teletext packets. Let us assume that we require the capability of correcting any single burst of 30 bits in a packet. The (255,247) RS code over GF(256), truncated to (33,25) has d* = 9 and can therefore correct up to four 8-bit symbols. This will thus allow correction of any single 32-bit burst provided that there are no other errors in the packet. Alternatively, it will allow correction of any four randomly occuring bit errors. The rate of this code is 0.76.

On the other hand, the (63,53) RS code over GF(64), truncated to (44,34) has a minimum distance of 11 and can therefore correct up to five 6-bit symbols which are in error. Thus it can correct any 30-bit burst or any five randomly occuring errors. The rate of this code is 0.77. Thus, we can see that the RS code over GF (64) yields 25% better random error protection, slightly less burst error protection and slightly higher rate than the RS code over GF(256). The price paid for this improvement is that the encoder must unpack incoming 8-bit bytes into 6-bit symbols before encoding and that the decoder must perform the reverse procedure after decoding.

Reed-Solomon codes have been applied successfully in a number of areas including the Joint Tactical Information Distribution System (reference 3), the Compact Disk audio player (reference 8) and the Bundle code proposed for 7-bit teletext (reference 6).

3.5.8. Majority-Logic-Decodable Codes

This is a class of codes which is characterized by their ability to be realized with relatively simple hardware. Yamada (reference 7) describes a (272,190) code developed by NHK for use in the Japanese teletext system. This code is optimized for correction of random bit errors and is capable of correcting any 8 errors occurring in a packet of 272 bits. When adapted to the NABTS packet size, the rate of this code would be 0.69. The reference does not indicate whether there is any burst error correction capability built into the code beyond the 8 random bit corrections.

Thus, it would appear that for any given code rate, a Reed-Solomon code should provide better burst protection and the NHK code should provide better protection against random errors. This will be discussed in greater detail in the next section.

3.6. Summary

In this section we will summarize some of the results of the discussion occurring in the previous sections. In particular, we shall re-examine the characteristics of the three following codes:

* the Fire codes, which are optimal burst correcting codes;
* the Reed-Solomon codes, which provide both good independent error and burst error protection; and
* the NHK majority logic code, which is optimized for correction of random errors.

Table 4-1 shows the characteristics of some representative members of these three classes of codes. From the table we can see that each of the three classes of codes has some interesting properties. The Fire code is the only one which can achieve any useful correction capability at very high rates. However, at the rate of .76 to .77 the RS (44,34) code is more powerful than the Fire (264,200) code. At still lower rates of .68 to .69, the RS (44,30) and NHK (264,182) codes offer somewhat different levels of performance. The NHK code is guaranteed to correct one more random bit error than is the RS code. However, the RS code is clearly superior to the NHK in terms of burst error performance if the assumption stated in note 3 is correct.

The realizations of these codes are another important consideration. All three can be realized in hardware; there presently exist integrated circuits which implement the Fire code for disk controllers, the RS code for compact disk players and the NHK code for the Japanese teletext system. All of these codes are also tractable in software, although it is unknown whether an actual software implementation of the Fire code has been demonstrated. The Reed-Solomon code has been demonstrated in software by Mortimer et al, in their work on the Bundle code. Yamada's paper (reference 7) indicates that software processing of BCH or the NHK codes is impractical, however it is unclear what assumptions about throughput underlie this statement.

The availability of integrated circuit solutions to the error correction problem may, in fact, be the determining factor in selecting a code for a particular application. In particular, one Japanese manufacturer has announced, in preliminary form, an integrated circuit which performs the NHK majority logic error correction code. Although it would appear that the RS (44,30) code will provide better performance than the NHK code, it is unlikely that any manufacturer would choose to implement it in an integrated circuit, given the availability of an integrated circuit for the NHK code. The RS code, therefore, is most likely to be implemented in applications where the instantaneous throughput requirement is not extremely high. It is unlikely that there will be a hardware implementation of this code unless one or more major service providers indicate that they require the power which it provides.

3.7. Packet Erasures

One of the stated requirements is that it must be possible to recover from the loss of complete packets. Regardless of how powerful an error correction code is applied to each packet, some packets will be lost, even if the packet prefix is included in the horizontal code word. This can occur for two reasons:

* the packet can contain errors which exceed the error correction capabilities of the code; and

* some packets will be lost due to corruption of the framing code, which cannot be protected.

The reason why the framing code cannot be corrected by a high level error correction code, is that, in OSI terms, it is a piece of protocol information passed from the physical to the link layer of the receiver. That is, it is used to obtain byte synchronization, which is required before any error correction code can be processed; it therefore clearly cannot be protected by that code itself.

Therefore, there must be provision for packet erasure protection for 8-bit data, corresponding to the Bundle code described for 7-bit data. This would be one factor in favour of the Reed-Solomon code since it should be possible to perform the same algorithm for decoding of both horizontal and vertical code words. A product code consisting of a Reed-Solomon code on each packet and occasional packets of Reed-Solomon parity bits would provide an extremely effective protection mechanism against packet erasures.

Table 3-1: Characteristics of Representatives of Classes of Codes

CODE	RANDOM ERRORS (t)	SINGLE BURST LENGTH (B)	RATE (R)
FIRE (264,250) FROM (279,265)	1	5	.95
FIRE (264,231) FROM (3X105,3X94)	1-3 [1]	12	.88
FIRE (264,200) FROM (8X35,8X27)	1-8 [1]	24	.76
RS (44,34) FROM (63,53) OVER GF(64)	>5 [2]	30	.77
RS (44,32)	>6 [2]	36	.73
RS (44,30)	>7 [2]	42	.68
RS (44,28)	>8 [2]	48	.64
NHK (264,182)	8	? [3]	.69

NOTES:

1. The (264,231) Fire code can correct 3 bursts of length 4 or less, and the (264,200) Fire code can correct 8 bursts of length 3 or less, provided that the bursts fall into separate interleaved codewords. However, the largest number of random errors which can always be corrected is 1.

2. The RS codes can correct t random errors, and many patterns of t bursts of length 6 or less.

3. The maximum burst length for the NHK (264,182) code is unknown, but is assumed to be 8.

4. ACCESS CONTROL

4.1. Introduction

As discussed in the "Applications and Requirements Report", one of the principal requirements of several new classes of teletext services is a mechanism for controlling access to the services. There are fundamentally two issues involved:

* addressing messages to specific users or specific groups of users, in other words providing a mechanism analogous to a telephone number; and

* restricting access to services to those persons who are authorized to receive them.

In general, access control is required for one of two reasons:

* because the communication is private and is intended for a specific user; or

* because the service is for sale and the service provider wishes to prevent non-paying viewers from receiving the service.

4.2. Levels of Access Control

This section will assess the degree of security required by system operators providing the data services discussed in the previous report. The reader unfamiliar with the terminology of data security is referred to Appendix A which is a glossary of some of the more important terms which will be mentioned in this section.

The four following levels of access control, although chosen arbitrarily, correspond closely to four important classes of attempts to breach the security of a data communications system. They are in order of increasing complexity and increasing resistance to unauthorized access:

* Protection from inadvertent access or deliberate but casual attempts at access (level 1). The most common method of addressing this problem is to place the data in a "location" where the user is unlikely to look for it. In a communications system this is essentially equivalent to addressing.

* Deliberate but unsophisticated eavesdropping (Level 2). This refers to an individual who actively attempts to gain access to information which he is not authorized to receive but does not possess powerful tools for "breaking" codes. This level of security is normally achieved through

encryption using the DES or a similar technique.

* Systematic eavesdropping (Level 3). This is an attempt to break the security provisions of a service despite the fact that the data is encrypted. For even modestly complex encryption schemes such as the DES this requires enormous resources and is clearly only useful for highly valuable data. Protection against this level of attack on data security is normally only required on systems carrying National Security information or highly valuable commercial data. Since this is not the type of data which is likely to be broadcast through a teletext system, protection against this class of eavesdropping is not a requirement of the access control schemes being discussed here.

* The final class is the spoofer (Level 4). This is mentioned here for completeness only. The spoofer attempts to gain information about security provisions on a data communications channel by actively altering messages being exchanged between two parties. This is clearly not an issue in teletext systems which are inherently one-way.

Clearly from this discussion only levels 1 and 2 of access control are of interest in teletext systems. The need of the service provider is to make it difficult or uneconomical for unauthorized users to eavesdrop on the service, not to make it totally impossible.

4.3. Requirements

Having now defined the applicable classes of security, this section will discuss the requirements to be met by an access control mechanism for teletext.

* It must be possible to address messages to any given receiver or to a class of receivers, i.e. a closed user group. There must be provision for a hierarchy of classes and sub-classes.

* Any user must have access to data addressed directly to him, and to any class of data for which he is authorized. Any given user may have access to more than one class of service.

* The access control mechanism must include some form of encryption. The actual technique of encryption is not an issue here since some system operators may prefer to use proprietary techniques and would feel that this is not an issue which is subject to standardization. Nevertheless, this report will discuss encryption techniques and make some recommendations.

* There must be provision for some form of key management and a mechanism for distribution of keys. The only mechanism a service provider has for cancelling the authorization of users who have stopped their subscriptions to the service is the periodic updating of the keys used in the encryption process. This mechanism would also be used by the service provider to stay "one step ahead" of unauthorized users who may be attempting to break the security provisions of the system. The frequency of updating of the keys would be determined by the service provider, based on the perceived value of the information, the actual cost in bandwidth required for the updating and the estimated time it would take a dedicated eavesdropper to actually break the security provisions and discover the key.

* The access control mechanism must include a means of distinguishing between clear text and cipher text, since both classes of data may be multiplexed into the same teletext signal and receivers must be given direction as to how to process the data.

* The encryption techniques must be compatible with the error-correction schemes which would be used on these services. An example of the type of difficulty which can be encountered here is that the DES inherently produces data of unknown parity and, without further processing, is therefore fundamentally incompatible with any error-correction scheme which assumes that each byte in a packet contains odd parity.

* An additional feature which must be considered, although it is not an attribute of the communications system as such, is the control of the use of information after it has been received by the user. This would be applicable specifically to telesoftware programs where a given user has been authorized to make use of the program, but the service provider wishes to prevent that user from distributing the program to others for use. Typically, this would imply that the software program itself somehow makes use of the security mechanisms that have been built into the receiver and is therefore only able to run on the receiver through which it was captured.

4.4. Addressing

As discussed in the preceeding sections, there is a need for addressing of messages to individual receivers or groups of receivers. In the latter case, the addressing can take one of two forms:

* The user groups could be formed in one case of all terminals which have a portion of their unique terminal identification address in common. This would be analogous to the telephone numbering system where an area code or an exchange number

serve to delimit a geographical area.

* Alternatively, a user group may consist of all terminals which are authorized to access a specific service.

For the classes of service under consideration here, only the second of these two addressing approaches is feasible. This is due to the fact that any given user may belong to more than one user group and it is clearly not possible to devise an addressing scheme which allows any given user to belong to any or all potential user groups without also having ALL users belong to all user groups.

In any case, the user group addressing is a requirement because it allows access to given services to be restricted to paying subscribers. It is therefore logical that receivers judge whether or not to attempt to access a service based on a service address rather than their own unique terminal address.

There is clearly a need then for two separate addressing schemes:

* one of these is a service address and is used to identify a class of data being carried through the network; and

* the second is a user address to allow messages to be targeted to specific users.

Fortunately, as we shall see in the Format Recommendations report, NABTS as it is currently defined provides two separate addressing mechanisms - the packet address and the record address. We shall see that these facilities can be used to implement service and user addresses without any actual change or addition to the standard. However, we may also find that for other reasons there is a requirement for some new NABTS protocol information to be defined.

The service address can be used to implement the Level 1 security referred to in Article 4.2., that is, to prevent inadvertent or deliberate but casual access to services. This could be achieved by the service provider making use of service addresses which are not normally accessible to receivers other than those purchased or leased from the service provider. For example, if the packet address in NABTS were used as a service address, the service provider could limit access to certain services by assigning to those services a packet address containing a hexadecimal digit between A and F. Such packet addresses cannot be directly accessed by NABTS decoders which conform to the FSS (Fundamental Service Specification).

4.5. Encryption

The passive security measures provided by the service address are inadequate for the protection of services whose content is of some commercial value, since that commercial value provides motivation for deliberate attempts at breaking the security provisions. The use of an undocumented address as a security mechanism is simple to defeat. There is therefore need for some form of data encryption.

Consideration will be given to the DES and simpler encryption techniques, and not to more sophisticated security techniques, because of the underlying assumption that there is little likelihood of large numbers of users applying advanced cryptanalysis techniques to breaking services. This is especially true if they could gain access through a subscription at a tiny fraction of the cost of their cryptanalysis efforts.

The Data Encryption Standard (DES), especially when combined with a suitable key management policy, is virtually impenetrable without enormous computing resources and a very large amount of time. Therefore, the DES will be the strongest form of encryption discussed here. For this same reason only the ECB (Electronic Code Book) and (OFB) Output Feedback modes of operation of the DES will be considered here.

Two other commonly used modes, (CBC) Cipher Block Chaining and (CFB) Cipher Feed Back provide greater data security than ECB but do so at the cost of more complex processing and in general lower throughput. Since the additional security which they provide is not required in this application, there is nothing to justify their additional complexity and they will therefore not be discussed further.

4.5.1. Data Encryption Standard (DES)

The Data Encryption Standard (DES) has the following general characteristics:

* extremely high data security; and

* well suited for 8-bit data because it operates on blocks of 64 bits.

In order to use the DES efficiently with 7-bit data it would be necessary to pack nine 7-bit characters into 8 bytes, run these bytes through the encryption or decryption process and unpack the resulting 64 bits into ten 7-bit bytes. For this reason the DES is fundamentally incompatible, or at the least quite unwieldy to use, in conjunction with forward error correction codes depending on a parity bit in each byte. This includes the Product code defined in NABTS, and the C and Bundle codes which have been proposed as extensions to NABTS.

The ECB mode of operation of DES is by far the simplest and is supported by all of the existing DES integrated circuits. It provides excellent security but has the unfortunate characteristic that a single bit error in a 64 bit block of cipher text will be converted into 64 bit errors in the clear text resulting from decryption. It is therefore not well suited to use over error prone communications channels. If it were used in a teletext system, it should be combined with a powerful forward error correcting code and the receiving device should perform the error correction before attempting to do decryption.

The OFB mode of operation removes the error multiplication effect of ECB but does so at the cost of greatly increased complexity and lower throughput. Existing DES integrated circuits do not support the OFB mode of operation and therefore an OFB realization will be either:

* considerably more expensive than ECB if realized in hardware; or

* considerably slower if realized in software.

One would therefore require extensive information on channel error statistics, the service provider's required net error rate and the cost sensitivity of the receiving device in order to assess whether OFB is justifiable for any given application. On the surface, however, it would appear that the additional expense of OFB would be better spent in improving forward error correction techniques or in the inclusion of a more sophisticated teletext physical layer processing circuit, such as an echo canceler or automatic equalizer.

4.5.2. Alternate Encryption Techniques

Many different operations can be carried out on blocks of data which have the effect of rendering it unrecognizable to receivers which do not possess the key.

One promising technique consists of exclusive-OR'ing the data bytes in a teletext packet with a very long key. For example, a NABTS packet consisting of 28 bytes or 224 bits can be exclusive-OR'ed with a 224 bit sequence before transmission and exclusive-OR'ed with the same sequence upon reception in order to restore the original message. Considering the key as a sequence of 28 bytes, we can see that any key byte which contains one or more bits having the value 1 will cause the corresponding data byte to be modified. The value Ø in a key byte therefore produces clear text, and the 255 other possible values of a key byte will all produce cypher text. There are therefore 255**28 (2.4x10E67) possible keys for 28 byte packets. A large majority of these keys would in fact be unusable because they would too closely resemble other keys and would therefore do a reasonable job of decrypting data which they are not intended to decrypt.

Two interesting results from error control theory can be applied to estimating the number of useful keys, and to finding them. The Singleton Bound states that the minimum distance of a code must be less than or equal n-k+1 (see Article 3.5.3.). We see then that searching for a set of keys which differ in d* positions is equivalent to designing an error-control code of minimum distance d*. We know also that a Reed-Solomon code is a maximum distance code, having d* = n-k+1.

Thus, we can use the known techniques for generating a Reed-Solomon code over GF(256) to produce an optimal set of keys. For example, taking k=6 yields 256**6 (= 2.81ØE14) keys, all of which differ from each other in at least 23 (= 28-6+1) of their 28 bytes.

Such a technique clearly provides inferior resistance to systematic cryptanalytic study than does the DES because each character is independent of other characters. It is therefore subject to study based on, for example, the frequency of occurrence of letters of the alphnabet in englisn speech or certain assumptions about the format of records and the presence of numeric data in them. However, such a technique, especially when combined with a relatively frequent replacement of the keys, would deter all but the most determined eavesdropper.

This technique has the benefit of requiring no additional hardware in the receiver and very little additional software processing and would therefore provide reasonable security with minimal cost and maximum throughput.

It also has the additional virtue that it can be made to maintain the parity of data bytes if required. This can be achieved by restricting the key bytes to those values which contain an even number of bits having the value 1 (2, 4, 6 or 8 ones). There are 127 possible key bytes under these circumstances instead of the 255 previously mentioned. This would allow this encryption technique to be used in conjunction with the Product, C or Bundle codes.

4.6. Key Management

A subscription service based on protection of the data through encryption requires a system for management of the encryption keys. This system should allow a service provider to periodically change the keys and to selectively grant or remove authorization from individual receivers for specific services.

In interactive data communication systems, a large variety of key management techniques exist. The following analogy is frequently used to describe the ideal way of transfering a piece of information from A to B. A places the information in a locked box and forwards the box to B. B places a second lock on the box and returns it to A. A removes A's lock and sends the box to B. B removes B's lock and examines the information. This allows

each user to communicate with the other without knowledge of the other's key. A number of schemes exist which are based on this type of logic. Other techniques, such as the use of public and private keys and special key encrypting keys, are also common.

Many of these techniques are unavailable in a one way system such as teletext and, as discussed previously, are probably not required for the limited level of security needed in a teletext system. There are fundamentally two ways of distributing keys in a one-way teletext system - through the actual teletext channel and through some other means.

4.6.1. In-Channel Key Distribution:

Distribution of keys through the teletext channel itself has some interesting advantages:

* The principal one is that it is possible to deny access to a specific service or group of services to an individual user very quickly by simply loading a false set of keys into that users terminal through the teletext channel.

* The second is that the service provider is considerably less dependent on other communication networks, and is less subject to associated communications costs.

* Thirdly, the user is not required to take any special action to renew his subscription to the service other than paying his bills. Certain other key distribution systems might require that the user periodically have a card updated or re-obtain authorization through a telephone line. This could have the effect of reducing the actual use of the service by some users and thereby reducing the service operator's revenues.

Even an in-channel key management scheme requires that at least some keys be distributed in another form, however. This is because keys transmitted in the channel have to be encrypted themselves. If they were transmitted in clear text, it would be possible for any eavesdropper to receive them and subsequently use them in accessing services for which he is not authorized.

There are two principal approaches to the solution of this problem:

* The first is to distribute through one of the means discussed below, a list of keys which, when combined with the keys received through the teletext channel, produce keys that can be used to decrypt the data services. In this way neither the key received through the network nor the key received through the other means is sufficient in itself to gain access to the service. Both have to be in place before the user can receive an encrypted service.

* the speed with which connections and disconnections can be made is limited by the user, the mails and/or the ease of access to the service provider's premises or host computer.

The simplest of these techniques is manual entry of user access codes. The service provider provides a list to the user of numbers to be typed into the terminal to gain access to specific services. This technique clearly has very little cost associated with it but does not offer very good security. It is suitable for distribution of the key which the subscriber uses to access information addressed specifically to him, since this is a key which the subscriber has an interest in keeping confidential. It is not suitable, however, for distributing keys which will be used in accessing services, since one authorized user could provide the keys to any number of other unauthorized users who, by simply typing them in, would be able to gain access to the services without actually paying a subscription fee. This problem can be solved by having one hard-coded key in each receiving terminal which when combined with the keys typed in by the user provide access to specific services. In this way a given set of keys is only usable with a given receiver.

This hard-coded number could typically be programmed into non-volatile memory by the service provider or his installation agent before delivering the receiving terminal to the subscriber. The use of internal switches in the receiving terminal (or any other mechanism which can be read) is not suitable since the user could read the positions of the switches and provide those to unauthorized users along with a set of keys. It should be noted that the effect of this approach is to authorize a terminal rather than an individual for access to services. In other words a business user of the service would require a different set of keys for each receiving terminal on its premises unless an arrangement could be made with the service provider whereby the same hard-key was coded into each of that customer's terminals.

Key distribution can also be carried out through the use of a magnetic card or a "smart card". A magnetic card can contain the list of keys for the services to which the user has subscribed. This is inherently quite secure since the magnetic card can not easily be duplicated, but it has the disadvantage that the only way to remove authorization from any individual terminal is to change the key being used by all terminals. Typically this would mean that the user would be required to have his card updated or replaced once per billing period and that the keys used in encrypting the services would be changed once per billing period as well. In this way users whose accounts are in good standing would always have a magnetic card which provides the correct keys for receiving the services. The obvious disadvantages of this approach are: the cost of a magnetic card reader in each receiving terminal; and the effort and cost associated with distribution and updating of the magnetic cards.

The "smart card" is a device which contains a microprocessor and some PROM's. This has the same distribution method as the magnetic card, but the cost of reading the card is built into the card itself rather than into a reader in the terminal. In general, the cost of a "smart card" is less than that of a magnetic card reader, however, there can be considerable costs associated with lost or damaged cards. The one particular feature offered by the smart card is the capability to bill the subscriber according to his use of the service. Since the user's transactions can be logged in the card's memory, the bill presented by the service provider upon receipt of the subscriber's card for renewal can be made to reflect the subscriber's use of certain services.

In interactive services, a charge for use of the service is directly related to the service provider's costs in maintaining ports on a computer, telephone lines and other such fixed equipment. There is no such per-use cost to the service provider in a teletext system, and the "pay-per-view" feature, therefore, only serves to provide some flexibility in developing a pricing structure for the service. Individual service providers will have to determine whether the cost associated with maintaining and operating a "smart card" updating facility is justified by the billing flexibility which it gives them. In this sense, the "smart card" is a billing tool and not a data security tool as such.

The last key distribution method involves the use of a secondary communication channel such as a telephone line. In this case the user periodically phones in to the service provider's host computer, transacts an electronics funds transfer to pay for his subscription and receives in return, a set of access codes which the terminal can use in decrypting the services. One advantage of this videotex-like addition to the terminal is that the service provider's billing costs can be reduced due to the use of the electronics funds transfer (where this is permitted by law). The obvious disadvantage is the cost imposed by the modem in the receiving terminal and the operation of a large interactive host computer by the service provider. This technique offers good security but as with all techniques except in-channel key management, the frequency with which authorizations can be granted or removed is limited by the user's willingness to actively renew his subscription.

Because of the advantages of in-channel key distribution and the likelihood that some service providers would choose to make use of this mechanism, the recommendations for extensions to the NABTS being put forward in the next report in this series must include a provision for delivery of keys through the teletext channel. This does not imply that this is the only valid mechanism for distributing keys, but it is the only one which requires that special provision for key management be made in the teletext protocol.

4.7. Control of the Use of Telesoftware Programs

The preceding sections have discussed some of the considerations involved in designing an access control scheme for teletext services. The purpose of this access control scheme is to select and control which receiving terminals are capable of receiving a given service. In the case of telesoftware, there may, in some cases, be a requirement for control of the use of the programs once they have been received by the terminal. For example, the service provider may be willing to deliver a telesoftware program to a paying subscriber and grant him the continued use of that program once he has paid for it; however, the service provider would not want the subscriber to be able to make copies of the software and provide them to non-subscribers.

One method of restricting the use of these programs is to build into the program a knowledge of the key used in decrypting that program for reception, and a sequence of instructions causing the computer running the program to communicate with the receiving device. More specifically, the software program would transmit a block of data to the receiving device which would encrypt it and return it for further use by the software program. The program would compare the resulting encrypted data with the value it would have expected based on its knowledge of the encrypting key, and only allow further use of the program if there is a match. In other words, the software can only be used on a computer which is connected to the teletext receiving device through which the program was received.

This feature is one which is built in to the design of the software itself and of the interface between the teletext receiver and the computer which uses the software. It is not a communications issue in that it has no impact on the teletext transmission protocol used to deliver the data. There will therefore be no further discussion of this issue in this report.

5. VIDEO "FILM STRIPS"

5.1. Introduction

In this section we examine the problem and the design alternatives associated with the provision of television picture quality video "film strip" (VFS) sequences that would be used for example in the consumer catalogue shopping application or education and training.

Strictly speaking, the problem is not in the essential category with respect to potential impact on the NABTS specification, nevertheless, the discussion presents interesting possibilities and will be reviewed here.

5.2. Assumptions

The principal assumptions that are made in our considerations include:

* to limit the discussion to the considerations of video "film strips" dissemination only - transaction return communication is assumed to be provided in some reasonable and accepted manner;

* of the many possible future alternatives only those that are reasonably practical in the near current time frame (ie. 2 to 4 years) are considered;

* that the text and graphic portion can be adequately addressed with conventional teletext delivered on cable television in full channel mode; and

* that while the range of requirements is broad, and in some cases even awesome, they may not necessarily be met in all cases, and that only a meaningful subset would be satisfied.

5.3. Requirement

We review briefly the most important requirements associated with the provision of video "film strips", henceforth referred to as VFS.

They include:

* The ability to provide the customer/user relatively immediate (several seconds) access to data sets consisting of typical teletext frames, consisting of text and graphics, and to an associated VFS, which at its fullest extent could be a full colour video sequence of several tens of seconds in duration.

* The number of pages to be accessible may range typically from several hundred to as many as 50,000.

* The number of subscibers active at any one time may be from a few to 100,000.

* Each teletext data set will be approximately 2 kbytes which is typical for conventional broadcast teletext.

* The VFS has a broad span of characteristics which includes being monochrome or full color, single frame static to full movement over several tens of seconds, and with a picture size ranging from full to partial screen (half screen in each axis).

* The ability to retain a static picture for unlimited continuous viewing, or the ability to view a moving sequence and retain the final frame as a static image. It is conceivable that the viewer may wish to re-run the animate sequence or to look a a sequence of semi-static frames from it.

5.4. Design Approach

Provision of the functionality and performance to meet the requirements may be performed in a variety of ways, each with its own trade-offs. There are combinations of functionality for the principal system components which can provide a given capability.

We shall examine the generic role of the key components, the principal problems either for each or as a whole, the major technical concept alternatives, and table the most meaningful combination. From this we derive the implications on the NABTS specification.

5.4.1. Major Components

The major components of a system to provide VFS to complement a teletext-like catalogue service include preparation/creation, distribution delivery channel and the user terminal.

Regardless of which of the multitude of possible material sources might be - paper, video, photograph, and whatever the output form - video, compact disk, videotape, videodisk or digitally codified images, a preparation/creation system will be required to collect, massage and package the VFS material for delivery and distribution to the user terminal. Depending on the final system design approach(es) taken, the preparation/creation system design could vary widely. Relative to total costs (capital and operating), and key overall system technical problems, the design considerations associated with the VFS preparation/creation system are negligible, and hence will be not be addressed in

subsequent discussion.

The next major component is the distribution/delivery channel which provides the means whereby the VFS are transferred from service distribution head-end subsystem to the user terminal. This channel may be of a variety of forms - including hard or soft medium with digital or analogue coded form.

The user terminal provides the mechanism whereby the user is able, in addition to basic teletext reception, NAPLPS decode and return transaction communication functions, to "request" and view the VFS.

Bear in mind that the nature of these generic components can vary widely depending on the approach(es) taken.

5.4.2. Major Design (Approach) Alternatives

We consider now the major design alternatives which may lead to a possible approach for implementation of a VFS system. The relatively independent candidate design considerations include;

* cyclic versus request-driven supply of VFS sequences;
* video versus coded (NAPLPS) picture information;
* cable versus other physical media delivery and distribution mechanisms;
* local storage versus per-need demand of VFS; and
* single versus time-sequence frames.

These major design considerations are relatively independent of each other with the obvious exception of the distribution and access methods. That is to say, if a physical medium such as a video disk or tape (or compact disk, for that matter) is used to store and distribute the VFS, then the user access will be to locally stored VFS data versus a demand, real or virtual, from the delivery head-end.

5.4.2.1. Cyclic And Request-Driven Delivery

There are two major approaches to the delivery of VFS data. The first is to take the complete set of VFS frames, and to cycle them on a periodic basis, very much as one does with a broadcast teletext service. At the request of the user, the terminal would then "grab" the appropriate VFS as it passed by in the cycle.

The second major approach is one similar to that used in videotex, and represents a hybrid form of videotex. This is to place a request on a return communication channel to the head-end for the delivery of a specific VFS which is recognized and

captured by the terminal.

Broadcast delivery has the obvious advantage of not requiring a return communication link which will focus the request of many subscribers to one central point, and on a continuous on-line basis. However, the entire set of VFS for a given service will have to be broadcast regardless of the demand. The volume in some cases can be relatively high, resulting in consumption of channel capacity and potentially degrading access response time at the user terminal.

There is a "hard" form of broadcast delivery (such as with tape or disk) which is probably advantageous in those situations where the data volume is relatively high compared to the communication channel capacity, and where a long update latency can be tolerated, i.e. of the order of weeks or even months.

Request-driven VFS delivery has the advantage of occupying the delivery channel only with those items that are specifically requested. This means that the potential response time in a cable delivery environment should be acceptable even with a large active request population. Of course, a fundamental disadvantage is that a return communication system is required, and that this system will be active during the user's entire viewing session (although there are techniques to partially circumvent this) instead of being concentrated to the transaction process alone.

Given that in any case a transaction return communication system is required, a combination of broadcast for the most frequently viewed VFS, together with a request for specific groups of VFS depending on the user's particular range of interest, may offer possibilities. The practicality of one or another approach is highly dependent on the profile of the service, and that of the consumer's perusal habits with the new electronic media.

5.4.2.2. Video And Coded VFS.

VFS frames must be encoded for delivery, reception and display. There are two apparent approaches. The first is to use standard NTSC video, and the second is to codify the images in a digital form such as NAPLPS (recognizing that there are other schemes that could be used depending on the nature of the image and the desired output quality).

Determination of the video encoding approach invokes consideration of several factors. The first is that of the time duration, which may be from one field or part thereof, through one frame, to a frame sequence over several seconds. The images may occupy either the full field (or frame) expanse, or several, two through eight as partial screens, may be combined into one frame-full. It would, of course, be incumbent on the decoder to correctly select and display where required the right portion of the video frame.

Digital coding of the images presents some very attractive possibilities such as precise control of resolution, spatial extent, colour depth and optimization of the image as a function of its content and intended display appearance. In addition, it allows for the possibility of coding only the true image subject. This may produce a meaningfully significant reduction of the data set size. This potential reduction is not generally possible, or at best very restrictive, with a video encoded image, in which the entire space must be codified in one way only, (ie. NTSC composite video) regardless of content structure.

For example, if a lady in a fur coat is standing in front of a background, it is conceivable that the only image detail to be retained will be the lady in the coat. Therefore coding would apply to her only with the background excluded or codified in some other form.

Both approaches have their strengths and weaknesses. A video encoded VFS occupies a bandwidth that is proportional to the number of frames and the fraction of a frame used for that image. No control exists with respect to the bandwidth used (standard broadcast NTSC video). The image cannot be improved upon with increased decoder technological capability. A distinct advantages of the video encoding approach is that the source material is easily captured in video, and requires a minimum of processing in preparation for distribution. Much of the necessary equipment already exists in video studios. In addition, because the user terminal should ideally be capable of capturing, for immediate or delayed (and perhaps repeated) viewing, only video frames offer a reasonable hope of providing this animation.

Codified VFS presents a restricted alternative although it requires significantly more processing. The advantages include the ability to accommodate a wide range of decoder sophistication, the ability to codify the image optimally with respect to its content, including colour depth, spatial resolution and extent, and be easily stored by the user terminal. The channel capacity compared to video for a like quality of image is comparable. The single distinct disadvantage is the inability to efficiently present frame sequences i.e. animate or true film strip VFS, and to deliver the data at the required rate in the context of a NABTS teletext envelope. To view video-like quality with codified VFS, decoders will require more than the 16 states of colour resolution that they now have, probably at least 256 (or 8 bit planes worth) will be necessary. This feature can be provided gracefully in time. Standard FSS decoders are not able to accommodate even static VFS adequately, both with respect to practical data acquisition and processing rate (practical picture build-up time) and colour resolution. Special decoders with video capture will be required.

5.4.2.3. Cable And Physical Media Distribution

There are two major approaches to delivering/transporting VFS frames from the service provider to the user terminal system. These include "soft" electronic forms which are delivered on cable, or "hard" electronic forms such as on video disk or tape.

The nature of teletext-based services is predicated on the availability of time-volatile information, and so, for the catalogue shopping application we do not believe that use of physical media such as video disk or tape will be viable. The latency associated with the media update and distribution is of the order of weeks or even months. Also, the user terminal system must now bear the expense of, and include a mechanism to access, this media - be it a video disk or tape playback unit together with suitable interface logic. No doubt the response time and capacity concerns are admirably addressed with this approach.

Cable distribution presents perhaps the only viable approach. It does meet the time volatility requirement, the channel capacity of cable is likely adequate - certainly with new cable systems. We assume that more than one channel is available for a service if so required.

5.4.2.4. Local Storage Versus Head-End Demand

From the discussion on channel bandwidth (see below, Article TBA), and in the context of catalogue shopping, where the number of items can be extensive (to 50,000 items), the response time latency for either a request driven or broadcast VFS delivery can be considerable. A means of providing the appearance of improving upon this problem is to provide some type of local terminal storage capability. This local storage may range from several kilobytes of internal terminal RAM, through to an attached digital storage system, such as RAM and floppy disk found on small personal computers.

The approach is one whereby, as a function of the viewer's interest, several requests can be placed for VFS frames to be captured and stored locally. Subsequent user viewing in the session provides immediate access to the video frame with an unlimited number of repeat demands. Not doing so will mean that every view and review of a VFS will require the user to wait either until that particular VFS frame appears in the broadcast cycle or until the request is acknowledged and delivered from the head end.

This is a performance-related design issue. Nevertheless it is essential to a successful implementation of the service. We believe that some local storage capability is required, beginning with a soft memory cache built into the user terminal and at least of the order of 200 kilobytes. The advent of personal computers in the home may make larger local memory sizes more

practical.

5.4.2.5. Single And Sequential Frame VFS

In the ideal circumstance, the viewer would be able to see a short, of order 5 to 15 seconds, video sequence occupying a significant if not entire television screen. Obviously this represents a considerable volume of data, and therefore, occupancy of the delivery channel with implicit degradation of the service. The amount of data contained in a single frame compared to a sequence is proportional to the duration or number of frames. For any meaning, the video frame sequence must be of the order of several seconds at least. This immediately implies that the amount of data for the typical VFS sequence will be approximately 150 to 300 times that of a single frame (i.e. 30 frames per second times 5 to 10 seconds).

We remark that if a limited number of frames such as two, three, a dozen are required, we shall consider these as a concatenation of single frames.

5.4.3. Practical Limitations and Problem Areas

"Cadillac" VFS capability can be provided with a relatively high channel capacity and decoder or user terminal system of virtually unlimited technological capability. Someday, this will be possible, but it is not practical in the near future. If one takes into account the practical statistics related to the volume of items and the number of subscribers in a realistic catalogue shopping service, then the realities of implementation must be brought to bear. These are embodied in three principal areas - namely channel bandwidth, terminal technology (and therefore cost) and compatibility with the consumer electronic publishing world.

5.4.3.1. Channel Bandwidth

For the delivery of VFS in soft electronic form on cable, a simple calculation of average expected access time, in consideration of the number of items to be made available (to 50,000) in the service, shows that there are some practical problems.

The response time will be inversely proportional to the number of cable channels available, and directly proportional to the number of picture VFS items. Again we assume that only static images are presented.

In the case for which video encoded frames are delivered, a simple calculation shows that if one uses 3 cable channels, with four video sub-pictures per video frame, and 50,000 items are cycled, then the average access time to a specific item will be

of the order of about 70 seconds. This is probably acceptable. Time of the order of few tens of seconds is preferable. This approach implies that a request driven service for video would be necessary and that the terminal would have to be capable of grabbing a specific frame and extracting from it a portion of the video field i.e. the sub-picture to be viewed.

Now, let us assume that all VFS data is delivered in some codified form such as NAPLPS, and that it is delivered on a full channel using the NABTS teletext delivery structure. Then the average access time will be inversely proportional to the number of channels, the number of available lines per channel (typically about 500), and directly proportional to the data set size for each frame (expected to be of the order of 60 kilobytes) and the number of items in the cycle (possibly up to 50,000). In the case that its delivery is cyclic, the response time is expected to be approximately 3 times as for video encoded material. However possibly with limited terminal storage, the access time may have the appearance of being somewhat reduced.

In general, channel capacity usage is optimal for the composite video form of the data. The only practical way of improving the apparent response time will be a combination of using all available cable channels, (several in number are probably reasonable - certainly in new cable systems), and providing local terminal storage. Also, many users will, at any instant, be engaged in traditional teletext. Only when they find a subject of interest will they request the VFS, and the number of active VFS users may well be significantly less than the total number of users, relieving strain in a demand driven system.

5.4.3.2. Terminal Technology

While the proposed electronic catalogue shopping applications is in essence a large-scale classical teletext service, it does require the ability to provide VFS. It is the size of the potential services and quality of the VFS image, which put demands on the channel delivery capacity, and for which the achievement of acceptable performance must be addressed by an augmentation of the terminal technology in order to be viable. The unstated but obvious implication is that the user terminal cost will be higher. The task therefore is to find the point of optimal balance between cost, functionality, and performance while retaining the viability of the service.

There are two major considerations involved in terminal technology. The first relates to functionality, i.e. the spatial and colour resolution of the terminal such that it can present an image that is of normal TV quality on a portion or entire screen. The second relates to performance, that is the access time to a specific VFS for services that have of the order of 50,000 items each with its own VFS.

The standard FSS teletext terminals have a spatial resolution of 200Y x 256X picture elements, each with a storage capability of 16 colour and/or transparency states. The 16 states may be assigned to a minimum of 512 possible colours and/or blink sequence combinations. For all but specialized display situations this will be insufficient to meet the service requirements. The terminal will require additional display memory of at least 8 bits per picture element occupying at least 25 to 40 per cent of the total screen surface. Obviously coverage of the entire video screen would be advantageous. The allocation of the 8 bits would have to be some combination to optimally present the colour information. This might involve the specialized colour models (HLS or CYN) or some other appropriate scheme. Indeed we also should consider the possibility of allocating the memory space at different resolutions depending on the type of colour information which is contained to achieve the optimal visual effect with minimal memory. For example, luminance information must be displayed at higher resolution while chrominance information can be meaningfully displayed at one half or even one quarter the rate of luminance. This approach is the basis of the NTSC composite video colour coding scheme.

For the purpose of reducing the apparent access time to any randomly selected VFS item, a local storage scheme would be desirable. This ranges from adding local internal RAM, (of order several hundred kilobytes), to providing an interface to an external storage mechanism such as a personal computer mass store device. The disadvantage of the latter is the incremental cost and low likelihood of the average consumer possessing this item, and therefore we discount it from further consideration.

In order to take advantage of the additional channel capacity available with several video channels to distribute a given service (i.e. increasing the band width), the capability of a frequency agile tuner will have to be provided. Today's electronic tuners with digital electronic frequency synthesizers make this practical at little expense. Appropriate control will have to be exercised by the delivery end to insure that the basic teletext database is correctly linked and organized with the VFS data sets.

One final aspect of terminal technology relates to speed of decoding. Obviously, with a video image of any sort, the rate of display must be of the order of a second. This will probably mean that in addition to high speed access and perhaps some type of local storage, the writing rate into the display memory will have to be optimal, requiring a more powerful processing and display generator capability.

In summary, the current VFS decoder technology is probably only marginally acceptable for services such as those described. As terminal costs permit, additional buffer and video memory, and display generator performance will have to be introduced. The resulting terminal will be a fully compatible super set of the

current basic FSS terminal.

5.4.3.3. Compatibility

Services which require the addition of VFS capability must be compatible with the existing VFS technology for several important reasons:

* the initial decoder population will be automatically higher by virtue of the existence of the traditional broadcast services;

* the equipments available for the teletext delivery will already exist and be available, as will be picture creation systems;

* and the service may be gracefully upgraded in a strict super-set compatible manner above that which is available in a basic VFS, ensuring the greater probability of success.

5.4.4. Synthesis

We have examined components of the major approach alternatives, their practical limitations and associated problems, and tried to ensure that all of the considerations were independent of each other. One must therefore conclude that a practical system is made up of some meaningful combination of these approach components. Of the list presented in Article TBA, two in our opinion leave no alternative, namely:

* NAPLPS coded images require about three times the channel capacity as that of composite images, and in addition are not viable whatsoever to be used for the presentation of animated sequences, and will therefore be discounted from all further consideration; and

* cable (soft) distribution of electronic data is the only approach to meeting the important requirement of preserving volatile information, and therefore we discount the use of the VFS delivery in "hard" electronic form on such as video disks or tapes.

This leaves us with three categories of choices:

* cyclic versus request-driven broadcast VFS delivery; and

* local storage versus demand access; and

* single versus time-sequence frames.

5.4.5. A Meaningful Approach

We believe that the most meaningful combination of these is one in which the service with the VFS is delivered with:

* a mixture of cyclic VFS broadcast, for the frequently accessed VFS together with the broadcast of specially and less frequently requested items;

* that while demand to broadcast access will prevail, some limited form (at the manufacturer's discretion) of local static frame storage is tenable; and

* that while single frame VFS will prevail (in the interest of optimal response within the constraints of total channel capacity) and can be retained for continuous viewing, animated sequences will be available (probably by request) with the terminal capable of "freezing" and retaining a frame for continuous viewing.

5.4.6. Implications on Terminal Design

With this approach there are design implications on the user terminal which may be summarized as follows:

* at a minimum, a capability of performing all of the functions required to accomodate NABTS to the FSS level;

* a return communication capability, be it telco (most likely) or cable based;

* a frequency agile CATV tuner;

* a digital NTSC composite video continuous video display and single frame capture (or "freeze") capability - both functions with the ability to transpose a portion of the incoming VFS video into any region of the screen as seen by the viewer; and

* a capability to real-time process a NABTS compatible identification "label", delivered in the VBI simultaneously with a VFS frame or sequence, so that the digitizer / transposer hardware will know when to commence processing and open a "window" onto the required VFS for the viewer to see.

5.4.7. Implications on NABTS

In order to aid the real-time identification of VFS frames or sequences, special teletext digital front-end hardware will have to be added. To maintain simplicity of processing, the following features will have to be provided in the NABTS specification:

* VFS frame identification and attributes should be assigned to as low a level as possible;

* VFS frame identification might be made with a different Data Group Type label;

* a frame count number for the number of video frames belonging to a specific VFS animate sequence; and

* attributes identifying the structure and allocation of the video frame in those cases where it is divided into smaller subpictures.

6. ASIATIC TEXT FONTS

6.1. Introduction

Strictly speaking, the transfer of North American teletext and videotex technology, as embodied in the NABTS and NAPLPS standards, to Asiatic countries is not directly related to the problems associated with the examination of new teletext services in the North American environment, but nevertheless we address the problem for purposes of interest and completeness.

Technical consideration reveals that the essence of the problem associated with transfer of the North American technology to the Asiatic markets focuses on the suitable presentation of text fonts reflecting the indigenous writing methods for text in those regions.

The areas that we address are primarily Japanese and Arabic, with lesser emphasis on Korean and Chinese.

6.2. Assumptions

In approaching the problem we make the following basic assumptions:

* that the proposed approaches must be compatible supersets of NAPLPS and NABTS;

* that the problem of coding at the presentation level, and of decoding and display is significantly different and perhaps more fundamental than the process of creation, and therefore we ignore the creation process for this discussion;

* we do not address the Indian related languages (due to lack of time and relevant information, and because of the little likelihood of these technologies penetrating this region in the near foreseeable future) - perhaps the considerations used in addressing the other scripts will, in principle, be applicable to that of the Indian language.

6.3. Requirement

The principal requirement is to provide a consumer oriented teletext (and videotex) services, analogous to those in the North American environment, which can operate in the far and near Eastern Asiatic regions.

The general intent is to provide, in addition to the existing graphic and photographic capabilities of videotex and teletext decoders, a suitable rendition with sufficiently efficient coding of Asiatic text fonts. The net objective of this examination

will be to find the approach that requires the least development, both at a conceptual and at a product developmental level, such that the system requirements based on the North American products can be easily, inexpensively and quickly realized.

6.4. Text Characteristics

The Language Text that we will address divide into two groups, the first of which contains Chinese, Japanese, Korean, and the second which is Arabic.

The basic information used is based primarily on a series of articles published in the recent issue of IEEE Computer magazine, which addresses the problems of input of Chinese/Kanji, with the addition of problems associated with Chinese/Kanji text and data processing (reference 9).

We summarize the essential characteristics of the written component of the languages as they relate to their handling and display on digital computer system displays.

While there are different dialects of Arabic languages spoken in various Arabic countries, they all have essentially the same script characteristics. Our information for this analysis is based on a series of conversations with organizations working on similar problems for computer display products.

6.4.1. Chinese, Japanese and Korean

Chinese written characters originated as pictographs some 4,000 years ago, and have been used in their present form for more than 2,000 years. In Chinese they are called "hanzi", a term meaning "characters". This term is pronounced "kanji" in Japanese and "haja" in the Korean language.

Each hanzi represents a single spoken Chinese syllable and usually a single concept, but no reliable relationship remains between the graphical form of a character and either its sound or meaning. Most hanzi are built up out of component graphical structures in an orderly way but many of the most common characters have unique, indivisible forms.

About 1500 years ago, Japanese and Koreans borrowed the Chinese hanzi to write their own languages, even though these languages were totally unrelated to Chinese. Unlike Chinese, Japanese and Korean have grammatical word endings, (something like "-ed" or "-ing" endings in English. Since Chinese hanzi provided no way to write grammatical endings, the Japanese and Koreans devised phonetic alphabets for their own language. The Japanese "kana" phonetics alphabet exists as "hiragana" and "katakana"; the Korean phonetic alphabet is called "hangul".

The invention of these phonetic alphabets centuries ago made Chinese hanzi unnecessary for writing Japanese and Korean, yet they have been retained because of strong attachment to Chinese culture and beauty of the hanzi. As a compromise Japanese and Korean are written in an amalgam of Chinese hanzi ideographs plus native phonetic letters. The hanzi are gradually being phased out, noteably in Korea, but hanzi appear likely to remain an optional part of the Japanese and Korean scripts.

Vertical columns were the original mode of hanzi writing, and are still used in many contexts such as newspapers and novels, but in the modern context of business and technical writing the left to right horizontal format is a well established alternative.

Note also that there are many schemes for writing each of these languages entirely in the roman alphabet. Nowadays romanizations coexist with the native scripts, but are by no means expected to supplement them.

The question of how many distinct hanzi are in use is important for the design of digital video decoding and display systems. Practical modern dictionaries and computer encoder proposals offer only about 7,000 different hanzi. The number that an office typist might actually use is lower still - on the order of 4,000 for Chinese and 2,000 for Japanese and Korean. Moreover, Korean is now commonly written in pure hangul with no hanzi at all.

Specifically for the Japanese environment, the Radio Technical Council of Japan is making a recommendation for four character sets that are recommended to be used simultaneously:

* The hanzi character set which defines a total of 3,725 characters including hanzi characters, the katakana and hiragana signs, roman characters, numerics and additional characters (2 byte codes are used for the hanzi character set). These characters can be displayed in normal, double width, double height and double size.

* The primary characters set which defines the 52 roman characters, 10 numerics, and 32 marks and for which a primary character is represented by a single byte code.

* The katakana character set which defines 94 phonetic signs - katakana characters represented by a single byte code and any character size is valid.

* The hiragana character set which defines another set of phonetics having 91 signs, and is represented by a single byte code, any character size is valid.

6.4.2. Arabic

Written Arabic text is the same for most of the near east Arabic countries and dialects. The characteristics of the written text may be summarized as:

* having about 40 different logical characters, each with its own shape;
* text is written from right to left;
* the shape of most characters is dependent on position, ie. whether it is alone, at the beginning, middle or end of a word, resulting in a total of about 140 to 160 different character shapes.
* the appearance of the characters in a word is such that they touch very much in the sense that hand written roman letters touch;

6.5. Design Approach

6.5.1. NAPLPS Features

With the premise that we would like to accommodate the various Asian text with as little deviation from NAPLPS as possible, we now turn to review briefly the fundamental features of NAPLPS to generate the basis for possible approaches for accommodation of these Asiatic text requirements.

The relevant NAPLPS features that might be applicable include:

* the basic G-set structure which provides for less than 128 access codes;
* the concept of MACRO and DRCS definitions and invocations within the constraints of a 3 kilobyte total storage buffer;
* the ability to define an arbitrary (along axes) character path (up, down, left, right); and
* the capability of having 2 byte character reference sequence accessible under ISO 2022 , which is beyond the scope of NAPLPS but nevertheless compatible with it.

6.5.2. Major Approach Alternatives

The major approach alternatives to providing the standard text capabilities include:

* to provide a "soft" definition of the extended text set using concepts based on an extension to the MACRO and/or DRCS concepts to accomodate soft downloaded definitions of

the characters;

* to provide hard definition of the extended text set using local ROM based definition of the characters;

* to generate a graphical rendition of a text font using basic NAPLPS geometric drawing primitives; or

* some combination of the above.

If one considers as a first priority, the requirement to have optimal use of communication and storage space, then the use of direct letter for letter, line for line graphical definition of a text fonts becomes prohibitive in communication overhead and response, and is excluded all from further consideration.

That leaves us with the possibility of having either a soft or hard definition of an expanded G-set. Here again, the communication overhead associated with downloading the font definitions would be prohibitive.

For example, in the case of Japanese hanzi there are over 3,000 symbols which simply cannot be reloaded for every session.

If all of the necessary character shapes can be compressed into less than 127 symbols, then perhaps a downloading of the definitions either into an extended DRCS or MACRO definition may be applicable. A little calculation shows that assuming a 15 x 15 dot array character definition, the order of magnitude for the data set size to be downloaded would be approximately 4 kilobytes.

This leaves us with the conclusion that for hanzi/kanji character sets only a local terminal ROM definition is practical, and that for Arabic and the phonetic Japanese and Korean sets that downloading a soft definition into an extended DRCS or MACRO definition is also viable.

6.6. Application to Specific Languages

6.6.1. Japanese

Based on the recommendation based on the Radio Technical Council of Japan for the use of Japanese language for teletext, in a 525 line system we make our suggestions.

There are two basic types of character paths for the writing of Japanese language - they are vertical and horizontal, and they can be used in mixed mode. The minimum recommended density for text is such that it will require 248 horizontal by 204 vertical picture elements on the screen. For horizontally written text, a normal character field of 16 horizontal by 24 vertical dots are recommended. For vertical writing the normal character field size is 24 horizontal by 20 vertical dots is recommended.

Both character field sizes include the spaces between 2 adjacent characters in adjacent rows. A normal size character font composition of 15 horizontal by 18 vertical dots is suggested. The font can accommodate the hanzi characters as well as the Latin alphabet.

Here again the character path capability of NAPLPS provides a natural mechanism for writing in the vertical direction.

6.6.2. Arabic

In order to take the typical four variants of the 40 basic characters of Arabic and to create a set that would require less than 128 key codes, and such that they would look satisfactory on a low cost decoder, we make the following recommendation:

* That the character shape be the same in the beginning and middle of a word and that the shape of the letter at the end of a word and a single letter be the same. This is acceptable most of the time, in fact, it is a technique that is being used in the design of ASCII terminals targeted for the middle east market, and results in a 94 character set;

* All of the characters can and should be embedded in an 8 horizontal by 14 vertical cell definition, which results in a character array of 32 horizontal by 14 vertical in a 256X by 200Y system or 40 characters horizontal by 17 vertical in a 320 by 240 system. The latter may very well be required given that the 625 line PAL video system standard is used in the majority of middle east countries, and would require this kind of decoder resolution to maintain correct pictorial aspect ratio.

Perhaps by limiting the vertical cell resolution to 8 by 12, up to 20 horizontal lines of text can be provided. This appears to be sufficient for most normal applications.

A. DATA SECURITY: (A Mini-Course)

This short list of terms and their explanation is not ordered alphabetically, but is structured as a primer on the basic data security concepts required for this discussion.

ENCRYPTION: The process of altering a stream or block of data by the application of a process known to both the sender and the intended receiver, for the purpose of rendering the data unintelligible to eavesdroppers who do not know the details of the process.

CLEAR TEXT: Data which is not encrypted. Also known as "plain text".

CIPHER TEXT: Data which has been encrypted.

DES: Data Encryption Standard. An encryption algorithm approved by the U.S. National Bureau of Standards in 1977. It modifies each block of 64 data bits by means of a series of permutations and substitutions based on a 56-bit "key". The resulting block has the property that each of its 64 bits depends on each of the 64 input bits and each of the 56 bits of the key.

KEY: A number used in encrypting or decrypting data. In general, the same key is used by both sender and receiver. "Public Key" systems, which use two different but mathematically related keys, are the exception to this rule.

EAVESDROPPER: Also known as "passive tapper". An individual who attempts to gain access to encrypted data by observing and analyzing the data.

SPOOFER: Also known as "active tapper". An individual who inserts himself between the two participants in a data communication, and intercepts and transforms messages for the purpose of defeating the security provisions in use.

ECB: Electronic Code Book. The basic mode of operation of the DES, in which the same block of clear text always produces the same cipher text, for a given key. A single bit error in cipher text propagates into 64 bit errors after decryption.

CBC: Cipher Block Chaining. A DES technique in which each block of clear text is exclusive - or'ed with the previous cipher text block before encryption. This offers increased security relative to ECB, since each block is dependent on previous history and any attempt to break the code by studying a single block is doomed to failure. A single bit error in the cipher text causes 128 bits to be in error after decryption.

CFB: Cipher Feedback. A DES technique in which K bits of clear text are combined with K bits of cipher text and fed back to the

input of the encryption process. This increases security in the same way as CBC, but a single bit error in cipher text affects only 64 bits after decryption. The penalty paid is a more complex hardware realization which must perform encryption and decryption at (64/K) times the data rate.

OFB: Output Feedback. This DES technique is similar to CFB, but has the desirable property that a bit error in cipher text produces only a single error after decryption. However, this also makes it easier for a spoofer to make undetected modifications.

B. REFERENCES

1. "Data Encryption Equipment", C.R. Abbruscato, IEEE Communications Magazine, Sept. 1984.

2. "Special issue on Computer Security", IEEE Computer Magazine, July 1983.

3. "The Technology of Error-Correcting Codes", E.R. Beriekamp, Proc. IEEE, May 1980.

4. "Theory and Practice of Error Control Codes", R.E. Blanut, Addison - Wesley, 1983.

5. "Determination of Throughputs, Efficiencies and Optimal Block Lengths for an Error-Correction Scheme for the Canadian Broadcast Telidon System", M. Sablatash and J.R. Storey, Can. Elec. Eng. J., Vol. 5, No. 4, 1980.

6. Various unpublished papers by B. Mortimer et.al., Carleton University, Ottawa, Canada; on "Carleton", "C" and "Bundle" Codes.

7. "Development of Error-Correcting Methods For Broadcast Data Packets Multiplexed on NTSC TV Signals", O. Yamada, NHK Science And Technical Research Laboratories.

8. "Communications Aspects of the Compact Disk Digital Audio System", J.B.H. Peek, IEEE Communications Magazine, Feb.1985.

9. "Typing Chinese, Japanese and Korean", J.D.Becker, Xerox Corporation, IEEE Computer Magazine, January 1985

****

Customer

Model

Serial Number

10 Hearst Way, Kanata, Ontario K2L 2P4 613-592-4164 Telex 053-4174